Wireless medium busy - do not send

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_CCABUSY			Wireless medium busy - do not send
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.43 EVENTS_CCASTOPPED

Address offset: 0x14C

The CCA has stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_CCASTOPPED			The CCA has stopped
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.44 EVENTS_RATEBOOST

Address offset: 0x150

Ble_LR CI field received, receive mode is changed from Ble_LR125Kbit to Ble_LR500Kbit.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RATEBOOST			Ble_LR CI field received, receive mode is changed from Ble_LR125Kbit to Ble_LR500Kbit.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.45 EVENTS_TXREADY

Address offset: 0x154

RADIO has ramped up and is ready to be started TX path

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_TXREADY			RADIO has ramped up and is ready to be started TX path
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.46 EVENTS_RXREADY

Address offset: 0x158

RADIO has ramped up and is ready to be started RX path

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RXREADY			RADIO has ramped up and is ready to be started RX path
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.47 EVENTS_MHRMATCH

Address offset: 0x15C

MAC header match found

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_MHRMATCH			MAC header match found
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.48 EVENTS_SYNC

Address offset: 0x168

Preamble indicator

A possible preamble has been received in Ble_LR125Kbit, Ble_LR500Kbit, or Ieee802154_250Kbit modes during an RX transaction. False triggering of the event is possible.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_SYNC			Preamble indicator A possible preamble has been received in Ble_LR125Kbit, Ble_LR500Kbit, or Ieee802154_250Kbit modes during an RX transaction. False triggering of the event is possible.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.49 EVENTS_PHYEND

Address offset: 0x16C

Generated when last bit is sent on air, or received from air

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_PHYEND			Generated when last bit is sent on air, or received from air
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.50 EVENTS_CTEPRESENT

Address offset: 0x170

CTE is present (early warning right after receiving CTEInfo byte)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_CTEPRESENT			CTE is present (early warning right after receiving CTEInfo byte)
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.26.15.51 PUBLISH_READY

Address offset: 0x180

Publish configuration for event READY

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event READY will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.52 PUBLISH_ADDRESS

Address offset: 0x184

Publish configuration for event ADDRESS

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ADDRESS will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.53 PUBLISH_PAYLOAD

Address offset: 0x188

Publish configuration for event PAYLOAD

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event PAYLOAD will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.54 PUBLISH_END

Address offset: 0x18C

Publish configuration for event END

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event END will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.55 PUBLISH_DISABLED

Address offset: 0x190

Publish configuration for event DISABLED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event DISABLED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.56 PUBLISH_DEVMATCH

Address offset: 0x194

Publish configuration for event DEVMATCH

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event DEVMATCH will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.57 PUBLISH_DEVMISS

Address offset: 0x198

Publish configuration for event DEVMISS

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event DEVMISS will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.58 PUBLISH_RSSIEND

Address offset: 0x19C

Publish configuration for event RSSIEND

A new RSSI sample is ready for readout from the RADIO.RSSISAMPLE register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RSSIEND will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.59 PUBLISH_BCMATCH

Address offset: 0x1A8

Publish configuration for event BCMATCH

Bit counter value is specified in the RADIO.BCC register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event BCMATCH will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.60 PUBLISH_CRCOK

Address offset: 0x1B0

Publish configuration for event CRCOK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CRCOK will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.61 PUBLISH_CRCERROR

Address offset: 0x1B4

Publish configuration for event CRCERROR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CRCERROR will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.62 PUBLISH_FRAMESTART

Address offset: 0x1B8

Publish configuration for event FRAMESTART

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event FRAMESTART will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.63 PUBLISH_EDEND

Address offset: 0x1BC

Publish configuration for event EDEND

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event EDEND will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.64 PUBLISH_EDSTOPPED

Address offset: 0x1C0

Publish configuration for event EDSTOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event EDSTOPPED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.65 PUBLISH_CCAIDLE

Address offset: 0x1C4

Publish configuration for event CCAIDLE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CCAIDLE will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.66 PUBLISH_CCABUSY

Address offset: 0x1C8

Publish configuration for event CCABUSY

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CCABUSY will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.67 PUBLISH_CCASTOPPED

Address offset: 0x1CC

Publish configuration for event CCASTOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CCASTOPPED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.68 PUBLISH_RATEBOOST

Address offset: 0x1D0

Publish configuration for event RATEBOOST

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RATEBOOST will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.69 PUBLISH_TXREADY

Address offset: 0x1D4

Publish configuration for event TXREADY

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event TXREADY will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.70 PUBLISH_RXREADY

Address offset: 0x1D8

Publish configuration for event RXREADY

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RXREADY will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.71 PUBLISH_MHRMATCH

Address offset: 0x1DC

Publish configuration for event MHRMATCH

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event MHRMATCH will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.72 PUBLISH_SYNC

Address offset: 0x1E8

Publish configuration for event SYNC

A possible preamble has been received in Ble_LR125Kbit, Ble_LR500Kbit, or Ieee802154_250Kbit modes during an RX transaction. False triggering of the event is possible.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event SYNC will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.73 PUBLISH_PHYEND

Address offset: 0x1EC

Publish configuration for event PHYEND

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event PHYEND will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.74 PUBLISH_CTEPRESENT

Address offset: 0x1F0

Publish configuration for event CTEPRESENT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CTEPRESENT will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.26.15.75 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				U T S R Q P O N M L K H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	READY_START			Shortcut between event READY and task START
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
B	RW	END_DISABLE			Shortcut between event END and task DISABLE
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
C	RW	DISABLED_TXEN			Shortcut between event DISABLED and task TXEN
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
D	RW	DISABLED_RXEN			Shortcut between event DISABLED and task RXEN
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
E	RW	ADDRESS_RSSISTART			Shortcut between event ADDRESS and task RSSISTART
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
F	RW	END_START			Shortcut between event END and task START
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
G	RW	ADDRESS_BCSTART			Shortcut between event ADDRESS and task BCSTART
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				U T S R Q P O N M L K H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
H	RW	DISABLED_RSSISTOP			Shortcut between event DISABLED and task RSSISTOP
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
K	RW	RXREADY_CCASTART			Shortcut between event RXREADY and task CCASTART
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
L	RW	CCAIDLE_TXEN			Shortcut between event CCAIDLE and task TXEN
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
M	RW	CCABUSY_DISABLE			Shortcut between event CCABUSY and task DISABLE
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
N	RW	FRAMESTART_BCSTART			Shortcut between event FRAMESTART and task BCSTART
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
O	RW	READY_EDSTART			Shortcut between event READY and task EDSTART
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
P	RW	EDEND_DISABLE			Shortcut between event EDEND and task DISABLE
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
Q	RW	CCAIDLE_STOP			Shortcut between event CCAIDLE and task STOP
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
R	RW	TXREADY_START			Shortcut between event TXREADY and task START
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
S	RW	RXREADY_START			Shortcut between event RXREADY and task START
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
T	RW	PHYEND_DISABLE			Shortcut between event PHYEND and task DISABLE
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
U	RW	PHYEND_START			Shortcut between event PHYEND and task START
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.26.15.76 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				a Z Y V U T S R Q P O N M L K I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	READY			Write '1' to enable interrupt for event READY
			Set	1	Enable
			Disabled	0	Read: Disabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				a Z Y V U T S R Q P O N M L K I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Enabled	1	Read: Enabled
B	RW	ADDRESS			Write '1' to enable interrupt for event ADDRESS
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	PAYLOAD			Write '1' to enable interrupt for event PAYLOAD
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	END			Write '1' to enable interrupt for event END
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	DISABLED			Write '1' to enable interrupt for event DISABLED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	DEVMATCH			Write '1' to enable interrupt for event DEVMATCH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	DEVMISS			Write '1' to enable interrupt for event DEVMISS
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	RSSIEND			Write '1' to enable interrupt for event RSSIEND A new RSSI sample is ready for readout from the RADIO.RSSISAMPLE register
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
I	RW	BCMATCH			Write '1' to enable interrupt for event BCMATCH Bit counter value is specified in the RADIO.BCC register
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
K	RW	CRCOK			Write '1' to enable interrupt for event CRCOK
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
L	RW	CRCERROR			Write '1' to enable interrupt for event CRCERROR
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
M	RW	FRAMESTART			Write '1' to enable interrupt for event FRAMESTART
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
N	RW	EDEND			Write '1' to enable interrupt for event EDEND
			Set	1	Enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				a Z Y V U T S R Q P O N M L K I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
O	RW	EDSTOPPED			Write '1' to enable interrupt for event EDSTOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
P	RW	CCAIDLE			Write '1' to enable interrupt for event CCAIDLE
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Q	RW	CCABUSY			Write '1' to enable interrupt for event CCABUSY
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
R	RW	CCASTOPPED			Write '1' to enable interrupt for event CCASTOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
S	RW	RATEBOOST			Write '1' to enable interrupt for event RATEBOOST
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
T	RW	TXREADY			Write '1' to enable interrupt for event TXREADY
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
U	RW	RXREADY			Write '1' to enable interrupt for event RXREADY
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
V	RW	MHRMATCH			Write '1' to enable interrupt for event MHRMATCH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Y	RW	SYNC			Write '1' to enable interrupt for event SYNC A possible preamble has been received in Ble_LR125Kbit, Ble_LR500Kbit, or Ieee802154_250Kbit modes during an RX transaction. False triggering of the event is possible.
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Z	RW	PHYEND			Write '1' to enable interrupt for event PHYEND
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
a	RW	CTEPRESENT			Write '1' to enable interrupt for event CTEPRESENT
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.26.15.77 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				a	Z	Y			V	U	T	S	R	Q	P	O	N	M	L	K		I			H	G	F	E	D	C	B	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	RW	READY			Write '1' to disable interrupt for event READY
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	ADDRESS			Write '1' to disable interrupt for event ADDRESS
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	PAYLOAD			Write '1' to disable interrupt for event PAYLOAD
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	END			Write '1' to disable interrupt for event END
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	DISABLED			Write '1' to disable interrupt for event DISABLED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	DEVMATCH			Write '1' to disable interrupt for event DEVMATCH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	DEVMISS			Write '1' to disable interrupt for event DEVMISS
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	RSSIEND			Write '1' to disable interrupt for event RSSIEND A new RSSI sample is ready for readout from the RADIO.RSSISAMPLE register
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
I	RW	BCMATCH			Write '1' to disable interrupt for event BCMATCH Bit counter value is specified in the RADIO.BCC register
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
K	RW	CRCOK			Write '1' to disable interrupt for event CRCOK
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
L	RW	CRCERROR			Write '1' to disable interrupt for event CRCERROR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				a Z Y V U T S R Q P O N M L K I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
M	RW	FRAMESTART			Write '1' to disable interrupt for event FRAMESTART
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
N	RW	EDEND			Write '1' to disable interrupt for event EDEND
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
O	RW	EDSTOPPED			Write '1' to disable interrupt for event EDSTOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
P	RW	CCAIDLE			Write '1' to disable interrupt for event CCAIDLE
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Q	RW	CCABUSY			Write '1' to disable interrupt for event CCABUSY
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
R	RW	CCASTOPPED			Write '1' to disable interrupt for event CCASTOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
S	RW	RATEBOOST			Write '1' to disable interrupt for event RATEBOOST
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
T	RW	TXREADY			Write '1' to disable interrupt for event TXREADY
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
U	RW	RXREADY			Write '1' to disable interrupt for event RXREADY
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
V	RW	MHRMATCH			Write '1' to disable interrupt for event MHRMATCH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Y	RW	SYNC			Write '1' to disable interrupt for event SYNC A possible preamble has been received in Ble_LR125Kbit, Ble_LR500Kbit, or Ieee802154_250Kbit modes during an RX transaction. False triggering of the event is possible.
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				a Z Y V U T S R Q P O N M L K I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
Z	RW	PHYEND			Write '1' to disable interrupt for event PHYEND
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
a	RW	CTEPRESENT			Write '1' to disable interrupt for event CTEPRESENT
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.26.15.78 CRCSTATUS

Address offset: 0x400

CRC status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	CRCSTATUS			CRC status of packet received
			CRCError	0	Packet received with CRC error
			CRCOk	1	Packet received with CRC ok

7.26.15.79 RXMATCH

Address offset: 0x408

Received address

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	RXMATCH			Received address
					Logical address of which previous packet was received

7.26.15.80 RXCRC

Address offset: 0x40C

CRC field of previously received packet

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	RXCRC			CRC field of previously received packet
					CRC field of previously received packet

7.26.15.81 DAI

Address offset: 0x410

Device address match index

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	DAI			Device address match index Index (n) of device address, see DAB[n] and DAP[n], that got an address match

7.26.15.82 PDUSTAT

Address offset: 0x414

Payload status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	PDUSTAT			Status on payload length vs. PCNF1.MAXLEN
			LessThan	0	Payload less than PCNF1.MAXLEN
			GreaterThan	1	Payload greater than PCNF1.MAXLEN
B	R	CISTAT			Status on what rate packet is received with in Long Range
			LR125kbit	0	Frame is received at 125 kbps
			LR500kbit	1	Frame is received at 500 kbps

7.26.15.83 CTESTATUS

Address offset: 0x44C

CTEInfo parsed from received packet

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C C B A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	CTETIME			CTETime parsed from packet
B	R	RFU			RFU parsed from packet
C	R	CTETYPE			CTEType parsed from packet

7.26.15.84 DFESTATUS

Address offset: 0x458

DFE status information

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	SWITCHINGSTATE			Internal state of switching state machine
			Idle	0	Switching state Idle
			Offset	1	Switching state Offset
			Guard	2	Switching state Guard
			Ref	3	Switching state Ref
			Switching	4	Switching state Switching
			Ending	5	Switching state Ending
B	R	SAMPLINGSTATE			Internal state of sampling state machine
			Idle	0	Sampling state Idle
			Sampling	1	Sampling state Sampling

7.26.15.85 PACKETPTR

Address offset: 0x504

Packet pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x01000000				**0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PACKETPTR			Packet pointer Packet address to be used for the next transmission or reception. When transmitting, the packet pointed to by this address will be transmitted and when receiving, the received packet will be written to this address. This address is a byte aligned RAM address. See the memory chapter for details about which memories are avilable for EasyDMA.

7.26.15.86 FREQUENCY

Address offset: 0x508

Frequency

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000002				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	FREQUENCY		[0..100]	Radio channel frequency Frequency = 2400 + FREQUENCY (MHz)
B	RW	MAP			Channel map selection
			Default	0	Channel map between 2400 MHz and 2500 MHz Frequency = 2400 + FREQUENCY (MHz)
			Low	1	Channel map between 2360 MHz and 2460 MHz Frequency = 2360 + FREQUENCY (MHz)

7.26.15.87 TXPOWER

Address offset: 0x50C

Output power

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description	
A	RW	TXPOWER			RADIO output power Output power in number of dBm, i.e. if the value -20 is specified the output power will be set to -20 dBm. When the radio is operated on high voltage (see VREQCTRL — Voltage request control on page 60 for how to control voltage), the output power is increased by 3 dBm. I.e. if the TXPOWER value is set to 0 dBm and high voltage is requested using VREQCTRL, the output power will be +3 dBm.	
			0dBm	0x0	0 dBm	
			Neg1dBm	0xFF	-1 dBm	
			Neg2dBm	0xFE	-2 dBm	
			Neg3dBm	0xFD	-3 dBm	
			Neg4dBm	0xFC	-4 dBm	
			Neg5dBm	0xFB	-5 dBm	
			Neg6dBm	0xFA	-6 dBm	
			Neg7dBm	0xF9	-7 dBm	
			Neg8dBm	0xF8	-8 dBm	
			Neg12dBm	0xF4	-12 dBm	
			Neg16dBm	0xF0	-16 dBm	
			Neg20dBm	0xEC	-20 dBm	
			Neg30dBm	0xE2	-40 dBm	Deprecated
			Neg40dBm	0xD8	-40 dBm	

7.26.15.88 MODE

Address offset: 0x510

Data rate and modulation

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MODE			Radio data rate and modulation setting. The radio supports frequency-shift keying (FSK) modulation.
			Nrf_1Mbit	0	1 Mbps Nordic proprietary radio mode
			Nrf_2Mbit	1	2 Mbps Nordic proprietary radio mode
			Ble_1Mbit	3	1 Mbps BLE
			Ble_2Mbit	4	2 Mbps BLE
			Ble_LR125Kbit	5	Long Range 125 kbps TX, 125 kbps and 500 kbps RX
			Ble_LR500Kbit	6	Long Range 500 kbps TX, 125 kbps and 500 kbps RX
			Ieee802154_250Kbit	15	IEEE 802.15.4-2006 250 kbps

7.26.15.89 PCNF0

Address offset: 0x514

Packet configuration register 0

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID		J	J			I	H	H	G	G		F	E	E	E	E								C					A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LFLEN			Length on air of LENGTH field in number of bits
C	RW	S0LEN			Length on air of S0 field in number of bytes
E	RW	S1LEN			Length on air of S1 field in number of bits
F	RW	S1INCL			Include or exclude S1 field in RAM
			Automatic	0	Include S1 field in RAM only if S1LEN > 0
			Include	1	Always include S1 field in RAM independent of S1LEN
G	RW	CILEN			Length of code indicator - Long Range
H	RW	PLEN			Length of preamble on air. Decision point: TASKS_START task
			8bit	0	8-bit preamble
			16bit	1	16-bit preamble
			32bitZero	2	32-bit zero preamble - used for IEEE 802.15.4
			LongRange	3	Preamble - used for Bluetooth LE Long Range
I	RW	CRCINC			Indicates if LENGTH field contains CRC or not
			Exclude	0	LENGTH does not contain CRC
			Include	1	LENGTH includes CRC
J	RW	TERMLEN			Length of TERM field in Long Range operation

7.26.15.90 PCNF1

Address offset: 0x518

Packet configuration register 1

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C C C B B B B B B B B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MAXLEN		[0..255]	Maximum length of packet payload. If the packet payload is larger than MAXLEN, the radio will truncate the payload to MAXLEN.
B	RW	STATLEN		[0..255]	Static length in number of bytes The static length parameter is added to the total length of the payload when sending and receiving packets, e.g. if the static length is set to N the radio will receive or send N bytes more than what is defined in the LENGTH field of the packet.
C	RW	BALEN		[2..4]	Base address length in number of bytes The address field is composed of the base address and the one byte long address prefix, e.g. set BALEN=2 to get a total address of 3 bytes.
D	RW	ENDIAN			On-air endianness of packet, this applies to the S0, LENGTH, S1, and the PAYLOAD fields.
			Little	0	Least significant bit on air first
			Big	1	Most significant bit on air first
E	RW	WHITEEN			Enable or disable packet whitening Including the address field to CRC check is not supported for whitened packets.
			Disabled	0	Disable
			Enabled	1	Enable

7.26.15.91 BASE0

Address offset: 0x51C

Base address 0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	BASE0			Base address 0

7.26.15.92 BASE1

Address offset: 0x520

Base address 1

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	BASE1			Base address 1

7.26.15.93 PREFIX0

Address offset: 0x524

Prefixes bytes for logical addresses 0-3

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				D D D D D D D D C C C C C C C C B B B B B B B B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-D	RW	AP[i] (i=0..3)			Address prefix i.

7.26.15.94 PREFIX1

Address offset: 0x528

Prefixes bytes for logical addresses 4-7

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				D D D D D D D D C C C C C C C C B B B B B B B B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-D	RW	AP[i] (i=4..7)			Address prefix i.

7.26.15.95 TXADDRESS

Address offset: 0x52C

Transmit address select

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TXADDRESS			Transmit address select Logical address to be used when transmitting a packet

7.26.15.96 RXADDRESSES

Address offset: 0x530

Receive address select

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-H	RW	ADDR[i] (i=0..7)			Enable or disable reception on logical address i.
			Disabled	0	Disable
			Enabled	1	Enable

7.26.15.97 CRCCNF

Address offset: 0x534

CRC configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LEN		[1..3]	CRC length in number of bytes For MODE Ble_LR125Kbit and Ble_LR500Kbit, only LEN set to 3 is supported
			Disabled	0	CRC length is zero and CRC calculation is disabled
			One	1	CRC length is one byte and CRC calculation is enabled
			Two	2	CRC length is two bytes and CRC calculation is enabled
			Three	3	CRC length is three bytes and CRC calculation is enabled
B	RW	SKIPADDR			Include or exclude packet address field out of CRC calculation.
			Include	0	CRC calculation includes address field
			Skip	1	CRC calculation does not include address field. The CRC calculation will start at the first byte after the address.
			Ieee802154	2	CRC calculation as per 802.15.4 standard. Starting at first byte after length field.

7.26.15.98 CRCPOLY

Address offset: 0x538

CRC polynomial

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CRCPOLY			CRC polynomial Each term in the CRC polynomial is mapped to a bit in this register which index corresponds to the term's exponent. The least significant term/bit is hardwired internally to 1, and bit number 0 of the register content is ignored by the hardware. The following example is for an 8 bit CRC polynomial: x8 + x7 + x3 + x2 + 1 = 1 1000 1101 .

7.26.15.99 CRCINIT

Address offset: 0x53C

CRC initial value

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CRCINIT			CRC initial value Initial value for CRC calculation

7.26.15.100 TIFS

Address offset: 0x544

Interframe spacing in µs

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TIFS			Interframe spacing in µs. Interframe space is the time interval between two consecutive packets. It is defined as the time, in microseconds, from the end of the last bit of the previous packet to the start of the first bit of the subsequent packet.

7.26.15.101 RSSISAMPLE

Address offset: 0x548

RSSI sample

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	RSSISAMPLE		[0..127]	RSSI sample. RSSI sample result. The value of this register is read as a positive value while the actual received signal strength is a negative value. Actual received signal strength is therefore as follows: received signal strength = -A dBm.

7.26.15.102 STATE

Address offset: 0x550

Current radio state

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	STATE			Current radio state
			Disabled	0	RADIO is in the Disabled state
			RxRu	1	RADIO is in the RXRU state
			RxIdle	2	RADIO is in the RXIDLE state
			Rx	3	RADIO is in the RX state
			RxDisable	4	RADIO is in the RXDISABLED state
			TxRu	9	RADIO is in the TXRU state
			TxIdle	10	RADIO is in the TXIDLE state
			Tx	11	RADIO is in the TX state
			TxDisable	12	RADIO is in the TXDISABLED state

7.26.15.103 DATAWHITEIV

Address offset: 0x554

Data whitening initial value

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A
Reset 0x00000040				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	DATAWHITEIV			Data whitening initial value. Bit 6 is hardwired to '1', writing '0' to it has no effect, and it will always be read back and used by the device as '1'. Bit 0 corresponds to Position 6 of the LSFR, Bit 1 to Position 5, etc.

7.26.15.104 BCC

Address offset: 0x560

Bit counter compare

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	BCC			Bit counter compare Bit counter compare register

7.26.15.105 DAB[n] (n=0..7)

Address offset: 0x600 + (n × 0x4)

Device address base segment n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	DAB			Device address base segment n

7.26.15.106 DAP[n] (n=0..7)

Address offset: 0x620 + (n × 0x4)

Device address prefix n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	DAP			Device address prefix n

7.26.15.107 DACNF

Address offset: 0x640

Device address match configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-H	RW	ENA[i] (i=0..7)			Enable or disable device address matching using device address i
			Disabled	0	Disabled
			Enabled	1	Enabled
I-P	RW	TXADD[i] (i=0..7)			TxAdd for device address i

7.26.15.108 MHRMATCHCONF

Address offset: 0x644

Search pattern configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MHRMATCHCONF			Search pattern configuration

7.26.15.109 MHRMATCHMAS

Address offset: 0x648

Pattern mask

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MHRMATCHMAS			Pattern mask

7.26.15.110 MODECNF0

Address offset: 0x650

Radio mode configuration register 0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C C A
Reset 0x00000200				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	RU			Radio ramp-up time
			Default	0	Default ramp-up time (tRXEN and tTXEN), compatible with firmware written for nRF51
			Fast	1	Fast ramp-up (tRXEN,FAST and tTXEN,FAST), see electrical specifications for more information When enabled, TIFS is not enforced by hardware and software needs to control when to turn on the Radio

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C C A
Reset 0x00000200				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
C	RW	DTX			Default TX value Specifies what the RADIO will transmit when it is not started, i.e. between: RADIO.EVENTS_READY and RADIO.TASKS_START RADIO.EVENTS_END and RADIO.TASKS_START RADIO.EVENTS_END and RADIO.EVENTS_DISABLED For IEEE 802.15.4 250 kbps mode only Center is a valid setting For Bluetooth Low Energy Long Range mode only Center is a valid setting
			B1	0	Transmit '1'
			B0	1	Transmit '0'
			Center	2	Transmit center frequency When tuning the crystal for center frequency, the RADIO must be set in DTX = Center mode to be able to achieve the expected accuracy

7.26.15.111 SFD

Address offset: 0x660

IEEE 802.15.4 start of frame delimiter

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x000000A7				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	SFD			IEEE 802.15.4 start of frame delimiter

7.26.15.112 EDCNT

Address offset: 0x664

IEEE 802.15.4 energy detect loop count

Number of iterations to perform an ED scan. If set to 0 one scan is performed, otherwise the specified number + 1 of ED scans will be performed and the max ED value tracked in EDSAMPLE.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EDCNT			IEEE 802.15.4 energy detect loop count

7.26.15.113 EDSAMPLE

Address offset: 0x668

IEEE 802.15.4 energy detect level

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	EDLVL		[0..127]	IEEE 802.15.4 energy detect level Register value must be converted to IEEE 802.15.4 range by an 8-bit saturating multiplication by factor ED_RSSISCALE, as shown in the code example for ED sampling

7.26.15.114 CCACTRL

Address offset: 0x66C

IEEE 802.15.4 clear channel assessment control

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				D D D D D D D D C C C C C C C C B B B B B B B B A A A
Reset 0x052D0000				**0 0 0 0 0 1 0 1 0 0 1 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CCAMODE			CCA mode of operation
			EdMode	0	Energy above threshold Will report busy whenever energy is detected above CCAEDTHRES
			CarrierMode	1	Carrier seen Will report busy whenever compliant IEEE 802.15.4 signal is seen
			CarrierAndEdMode	2	Energy above threshold AND carrier seen
			CarrierOrEdMode	3	Energy above threshold OR carrier seen
			EdModeTest1	4	Energy above threshold test mode that will abort when first ED measurement over threshold is seen. No averaging.
B	RW	CCAEDTHRES			CCA energy busy threshold. Used in all the CCA modes except CarrierMode. Must be converted from IEEE 802.15.4 range by dividing by factor ED_RSSISCALE - similar to EDSAMPLE register
C	RW	CCACORRTHRES			CCA correlator busy threshold. Only relevant to CarrierMode, CarrierAndEdMode, and CarrierOrEdMode.
D	RW	CCACORRCNT			Limit for occurances above CCACORRTHRES. When not equal to zero the corrolator based signal detect is enabled.

7.26.15.115 DFEMODE

Address offset: 0x900

Whether to use Angle-of-Arrival (AOA) or Angle-of-Departure (AOD)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	DFEOPMODE			Direction finding operation mode
			Disabled	0	Direction finding mode disabled
			AoD	2	Direction finding mode set to AoD
			AoA	3	Direction finding mode set to AoA

7.26.15.116 CTEINLINECONF

Address offset: 0x904

Configuration for CTE inline mode

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					I	I	I	I	I	I	I	I	H	H	H	H	H	H	H	H	G	G	G	F	F	F			E	E		C	B			A
Reset 0x00002800					**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CTEINLINECTRLEN			Enable parsing of CTEInfo from received packet in BLE modes
			Enabled	1	Parsing of CTEInfo is enabled
			Disabled	0	Parsing of CTEInfo is disabled
B	RW	CTEINFOINS1			CTEInfo is S1 byte or not
			InS1	1	CTEInfo is in S1 byte (data PDU)
			NotInS1	0	CTEInfo is NOT in S1 byte (advertising PDU)
C	RW	CTEERRORHANDLING			Sampling/switching if CRC is not OK
			Yes	1	Sampling and antenna switching also when CRC is not OK
			No	0	No sampling and antenna switching when CRC is not OK
E	RW	CTETIMEVALIDRANGE			Max range of CTETime Valid range is 2-20 in the Bluetooth Core Specification. If larger than 20, it can be an indication of an error in the received packet.
			20	0	20 in 8 µs unit (default) Set to 20 if parsed CTETime is larger than 20
			31	1	31 in 8 µs unit
			63	2	63 in 8 µs unit
F	RW	CTEINLINERXMODE1US			Spacing between samples for the samples in the SWITCHING period when CTEINLINEMODE is set. AoD 1 µs
			4us	1	4 µs
			2us	2	2 µs
			1us	3	1 µs
			500ns	4	0.5 µs
			250ns	5	0.25 µs
			125ns	6	0.125 µs
G	RW	CTEINLINERXMODE2US			Spacing between samples for the samples in the SWITCHING period when CTEINLINEMODE is set. When the device is in AoD mode, this is used when the received CTEType is "AoD 2 µs". When in AoA mode, this is used when TSWITCHSPACING is 4 µs.
			4us	1	4 µs
			2us	2	2 µs
			1us	3	1 µs
			500ns	4	0.5 µs
			250ns	5	0.25 µs
			125ns	6	0.125 µs
H	RW	S0CONF			S0 bit pattern to match The least significant bit always corresponds to the first bit of S0 received.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				I I I I I I I I H H H H H H H H G G G F F F E E C B A
Reset 0x00002800				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
I	RW	S0MASK			S0 bit mask to set which bit to match The least significant bit always corresponds to the first bit of S0 received.

7.26.15.117 DFECTRL1

Address offset: 0x910

Various configuration for Direction finding

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				I I I I H H H H G G G F E E E C C C B A A A A A A
Reset 0x00023282				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 1 0 1 0 0 0 0 0 1 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	NUMBEROF8US			Length of the AoA/AoD procedure in number of 8 µs units Always used in TX mode, but in RX mode only when CTEINLINECTRLEN is 0
B	RW	DFEINEXTENSION			Add CTE extension and do antenna switching/sampling in this extension
			CRC	1	AoA/AoD procedure triggered at end of CRC
			Payload	0	Antenna switching/sampling is done in the packet payload
C	RW	TSWITCHSPACING			Interval between every time the antenna is changed in the SWITCHING state
			4us	1	4 µs
			2us	2	2 µs
			1us	3	1 µs
E	RW	TSAMPLESPACINGREF			Interval between samples in the REFERENCE period
			4us	1	4 µs
			2us	2	2 µs
			1us	3	1 µs
			500ns	4	0.5 µs
			250ns	5	0.25 µs
			125ns	6	0.125 µs
F	RW	SAMPLETYPE			Whether to sample I/Q or magnitude/phase
			IQ	0	Complex samples in I and Q
			MagPhase	1	Complex samples as magnitude and phase
G	RW	TSAMPLESPACING			Interval between samples in the SWITCHING period when CTEINLINECTRLEN is 0 When CTEINLINECTRLEN is 1, CTEINLINERXMODE1US or CTEINLINERXMODE2US is used instead of TSAMPLESPACING.
			4us	1	4 µs
			2us	2	2 µs
			1us	3	1 µs
			500ns	4	0.5 µs
			250ns	5	0.25 µs
			125ns	6	0.125 µs
H	RW	REPEATPATTERN			Repeat each individual antenna pattern N times sequentially, i.e. P0, P0, P1, P1, P2, P2, P3, P3, etc.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				I I I I H H H H G G G F E E E C C C B A A A A A A
Reset 0x00023282				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 1 0 1 0 0 0 0 0 1 0**
ID	R/W	Field	Value ID	Value	Description
			NoRepeat	0	Do not repeat (1 time in total)
I	RW	AGCBACKOFFGAIN			Gain will be lowered by the specified number of gain steps at the start of CTE At the start of receiving the CTE, the gain is lowered by the specified number of gain steps (approximately 3 dB ± 1 dB per step). The initial gain reduction is implemented in the analog front-end, thus the gain back-off can be used to prevent sample saturation.

7.26.15.118 DFECTRL2

Address offset: 0x914

Start offset for Direction finding

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B B B B B B B B B B A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TSWITCHOFFSET			Signed value offset after the end of the CRC before starting switching in number of 16 MHz clock cycles Decreasing TSWITCHOFFSET beyond the trigger of the AoA/AoD procedure will have no effect
B	RW	TSAMPLEOFFSET			Signed value offset in number of 16 MHz clock cycles for fine tuning of the sampling instant for all IQ samples. With TSAMPLEOFFSET=0 the first sample is taken immediately at the start of the reference period Decreasing TSAMPLEOFFSET beyond the trigger of the AoA/AoD procedure will have no effect

7.26.15.119 SWITCHPATTERN

Address offset: 0x928

GPIO patterns to be used for each antenna

Maximum 8 GPIOs can be controlled. To secure correct signal levels on the pins, the pins must be configured in the GPIO peripheral as described in Pin configuration.

If, during switching, the total number of antenna slots is bigger than the number of written patterns, the RADIO loops back to the pattern used after the reference pattern.

A minimum number of three patterns must be written.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	SWITCHPATTERN			Fill array of GPIO patterns for antenna control. The GPIO pattern array size is 40 entries. When written, bit n corresponds to the GPIO configured in PSEL.DFEGPIO[n]. When read, returns the number of GPIO patterns written since the last time the array was cleared. Use CLEARPATTERN to clear the array.

7.26.15.120 CLEARPATTERN

Address offset: 0x92C

Clear the GPIO pattern array for antenna control

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CLEARPATTERN			Clears GPIO pattern array for antenna control
			Clear	1	Clear the GPIO pattern

7.26.15.121 PSEL.DFEGPIO[n] (n=0..7)

Address offset: 0x930 + (n × 0x4)

Pin select for DFE pin n

Must be set before enabling the radio

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.26.15.122 DFEPACKET.PTR

Address offset: 0x950

Data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x01000000	**0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer
					See the memory chapter for details about which memories are available for EasyDMA.

7.26.15.123 DFEPACKET.MAXCNT

Address offset: 0x954

Maximum number of buffer words to transfer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A
Reset 0x00001000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT			Maximum number of buffer words to transfer

7.26.15.124 DFEPACKET.AMOUNT

Address offset: 0x958

Number of samples transferred in the last transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT			Number of samples transferred in the last transaction

7.26.15.125 POWER

Address offset: 0xFFC

Peripheral power control

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A
Reset 0x00000001	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**

ID	R/W	Field	Value ID	Value	Description
A	RW	POWER			Peripheral power control. The peripheral and its registers will be reset to its initial state by switching the peripheral off and then back on again.
			Disabled	0	Peripheral is powered off
			Enabled	1	Peripheral is powered on

7.26.16 Electrical specification

7.26.16.1 General radio characteristics

Symbol	Description	Min.	Typ.	Max.	Units
f_{OP}	Operating frequencies	2360		2500	MHz
$f_{PLL,CH,SP}$	PLL channel spacing		1.0		MHz
$f_{DELTA,1M}$	Frequency deviation @ 1 Mbps		±170		kHz
$f_{DELTA,BLE,1M}$	Frequency deviation @ Bluetooth LE 1 Mbps		±250		kHz
$f_{DELTA,2M}$	Frequency deviation @ 2 Mbps		±320		kHz
$f_{DELTA,BLE,2M}$	Frequency deviation @ Bluetooth LE 2 Mbps		±500		kHz
fsk_{BPS}	On-the-air data rate	125		2000	kbps
$f_{chip,\ IEEE\ 802.15.4}$	Chip rate in IEEE 802.15.4 mode		2000		kchip/s

7.26.16.2 Radio current consumption (transmitter)

Symbol	Description	Min.	Typ.	Max.	Units
$I_{TX,PLUS3dBM,DCDC}$	TX only run current DC/DC, 3 V, P_{RF} = +3 dBm		5.1		mA
$I_{TX,PLUS3dBM}$	TX only run current P_{RF} = +3 dBm		11.3		mA
$I_{TX,0dBM,DCDC}$	TX only run current DC/DC, 3 V, P_{RF} = 0 dBm		3.4		mA
$I_{TX,0dBM}$	TX only run current P_{RF} = 0 dBm		9.1		mA
$I_{TX,MINUS4dBM,DCDC}$	TX only run current DC/DC, 3 V, P_{RF} = -4 dBm		2.7		mA
$I_{TX,MINUS4dBM}$	TX only run current P_{RF} = -4 dBm		7.2		mA
$I_{TX,MINUS8dBM,DCDC}$	TX only run current DC/DC, 3 V, P_{RF} = -8 dBm		2.2		mA
$I_{TX,MINUS8dBM}$	TX only run current P_{RF} = -8 dBm		5.8		mA
$I_{TX,MINUS12dBM,DCDC}$	TX only run current DC/DC, 3 V, P_{RF} = -12 dBm		2.0		mA
$I_{TX,MINUS12dBM}$	TX only run current P_{RF} = -12 dBm		5.0		mA
$I_{TX,MINUS16dBM,DCDC}$	TX only run current DC/DC, 3 V, P_{RF} = -16 dBm		1.8		mA
$I_{TX,MINUS16dBM}$	TX only run current P_{RF} = -16 dBm		4.5		mA
$I_{TX,MINUS20dBM,DCDC}$	TX only run current DC/DC, 3 V, P_{RF} = -20 dBm		1.7		mA
$I_{TX,MINUS20dBM}$	TX only run current P_{RF} = -20 dBm		4.2		mA
$I_{TX,MINUS40dBM,DCDC}$	TX only run current DC/DC, 3 V, P_{RF} = -40 dBm		1.5		mA
$I_{TX,MINUS40dBM}$	TX only run current P_{RF} = -40 dBm		3.8		mA
$I_{START,TX,DCDC}$	TX start-up current DC/DC, 3 V, P_{RF} = 4 dBm		2.4		mA
$I_{START,TX}$	TX start-up current, P_{RF} = 4 dBm		5.4		mA

7.26.16.3 Radio current consumption (Receiver)

Symbol	Description	Min.	Typ.	Max.	Units
$I_{RX,1M,DCDC}$	RX only run current DC/DC, 3 V, 1 Mbps/1 Mbps Bluetooth LE mode		2.7		mA
$I_{RX,1M}$	RX only run current LDO, 3 V, 1 Mbps/1 Mbps Bluetooth LE mode		6.7		mA
$I_{RX,2M,DCDC}$	RX only run current DC/DC, 3 V, 2 Mbps/2 Mbps Bluetooth LE mode		3.1		mA
$I_{RX,2M}$	RX only run current LDO, 3 V, 2 Mbps/2 Mbps Bluetooth LE mode		7.9		mA
$I_{START,RX,1M,DCDC}$	RX start-up current DC/DC, 3 V, 1 Mbps/1 Mbps Bluetooth LE mode	..	..	..	mA
$I_{START,RX,1M}$	RX start-up current 1 Mbps/1 Mbps Bluetooth LE mode	..	..	..	mA

7.26.16.4 Transmitter specification

Symbol	Description	Min.	Typ.	Max.	Units
P_{RF}	Maximum output power		3.0		dBm
P_{RFC}	RF power control range		23.0		dB
P_{RFCR}	RF power accuracy		±2		dB
$P_{RF1,1}$	1st Adjacent Channel Transmit Power 1 MHz (1 Mbps)		-24		dBc
$P_{RF2,1}$	2nd Adjacent Channel Transmit Power 2 MHz (1 Mbps)		-52		dBc
$P_{RF1,2}$	1st Adjacent Channel Transmit Power 2 MHz (2 Mbps)		-25		dBc
$P_{RF2,2}$	2nd Adjacent Channel Transmit Power 4 MHz (2 Mbps)		-50		dBc
E_{vm}	Error vector magnitude in IEEE 802.15.4 mode	..	..	..	%rms
$P_{harm2nd, IEEE 802.15.4}$	2nd harmonics in IEEE 802.15.4 mode		-51		dBm
$P_{harm3rd, IEEE 802.15.4}$	3rd harmonics in IEEE 802.15.4 mode		-51		dBm

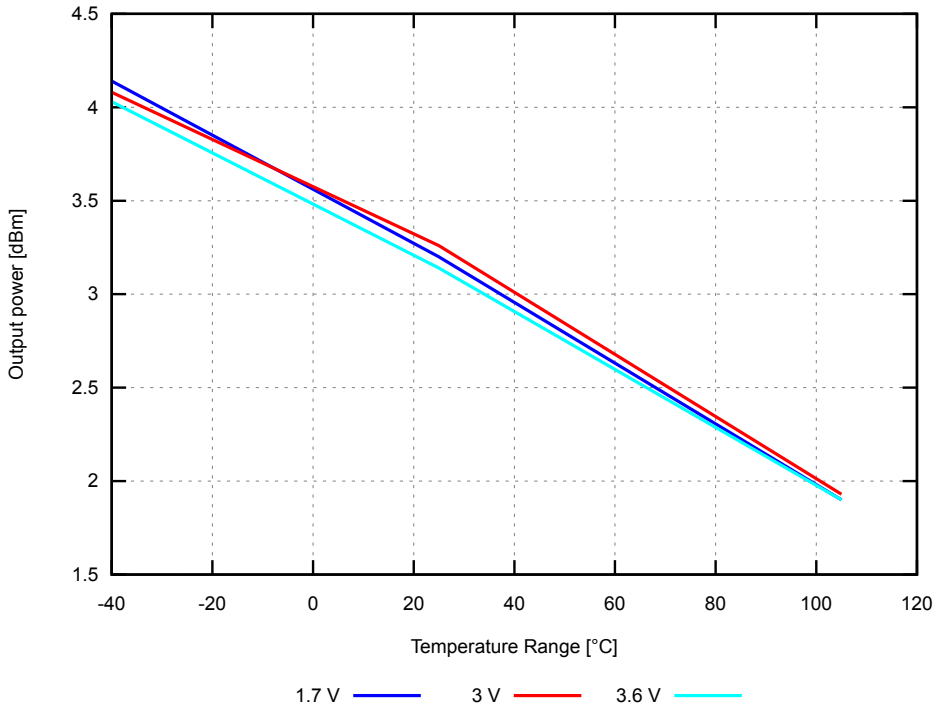


Figure 170: Output power, 1 Mbps Bluetooth low energy mode, at maximum TXPOWER setting (typical values)

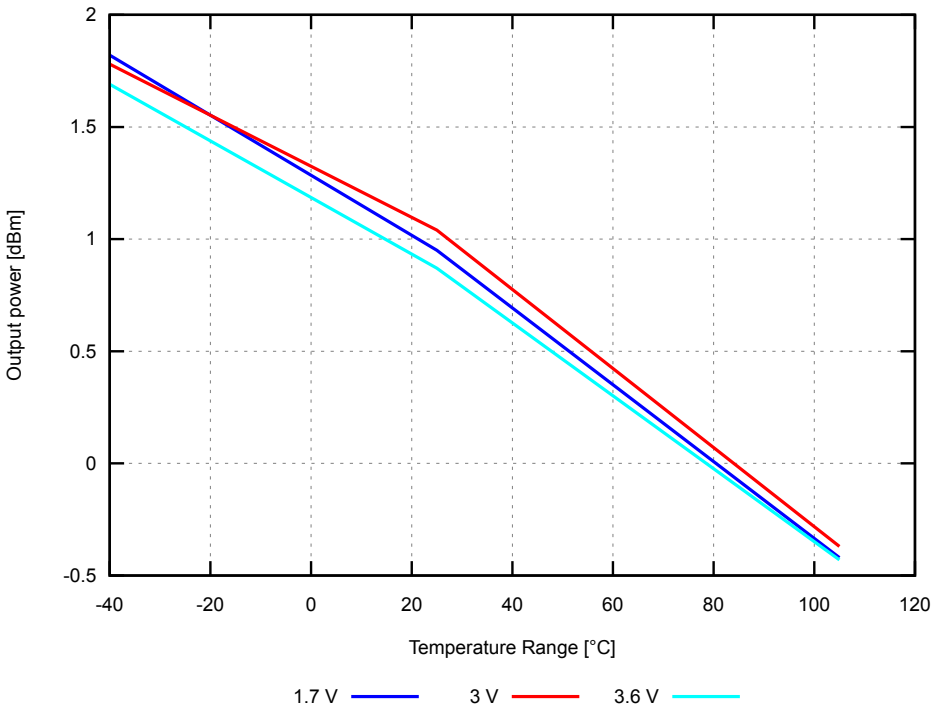


Figure 171: Output power, 1 Mbps Bluetooth low energy mode, at 0 dBm TXPOWER setting (typical values)

7.26.16.5 Receiver operation

Symbol	Description	Min.	Typ.	Max.	Units
$P_{RX,MAX}$	Maximum received signal strength at < 0.1% PER		0		dBm
$P_{SENS,IT,1M}$	Sensitivity, 1 Mbps nRF mode ideal transmitter [12]		-95		dBm
$P_{SENS,IT,2M}$	Sensitivity, 2 Mbps nRF mode ideal transmitter [12]		-92		dBm
$P_{SENS,IT,SP,1M,BLE}$	Sensitivity, 1 Mbps Bluetooth LE ideal transmitter, packet length ≤ 37 bytes BER = 1E-3[13]		-98		dBm
$P_{SENS,IT,LP,1M,BLE}$	Sensitivity, 1 Mbps Bluetooth LE ideal transmitter, packet length ≥ 128 bytes BER = 1E-4 [14]		-97		dBm
$P_{SENS,IT,SP,2M,BLE}$	Sensitivity, 2 Mbps Bluetooth LE ideal transmitter, packet length ≤ 37 bytes		-95		dBm
$P_{SENS,IT,BLE\ LE125k}$	Sensitivity, 125 kbps Bluetooth LE mode		-104		dBm
$P_{SENS,IT,BLE\ LE500k}$	Sensitivity, 500 kbps Bluetooth LE mode		-100		dBm
$P_{SENS,IEEE\ 802.15.4}$	Sensitivity in IEEE 802.15.4 mode		-101		dBm

12 Typical sensitivity applies when ADDR0 is used for receiver address correlation. When ADDR[1...7] are used for receiver address correlation, the typical sensitivity for this mode is degraded by 3 dB.

13 As defined in the *Bluetooth Core Specification v4.0 Volume 6: Core System Package (Low Energy Controller Volume)*.

14 Equivalent BER limit < 10E-04.

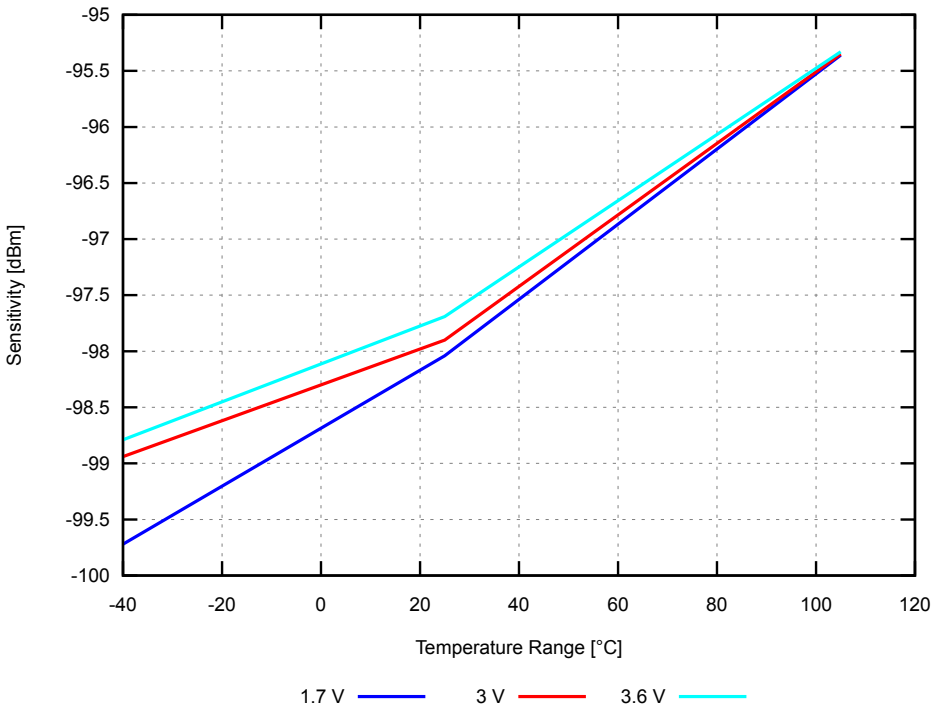


Figure 172: Sensitivity, 1 Mbps Bluetooth low energy mode, Regulator = DCDC (typical values)

7.26.16.6 RX selectivity

RX selectivity with equal modulation on interfering signal[15]

Symbol	Description	Min.	Typ.	Max.	Units
$C/I_{1M,co\text{-}channel}$	1Mbps mode, co-channel interference	..	..	..	dB
$C/I_{1M,-1MHz}$	1 Mbps mode, Adjacent (-1 MHz) interference	..	..	..	dB
$C/I_{1M,+1MHz}$	1 Mbps mode, Adjacent (+1 MHz) interference	..	..	..	dB
$C/I_{1M,-2MHz}$	1 Mbps mode, Adjacent (-2 MHz) interference	..	..	..	dB
$C/I_{1M,+2MHz}$	1 Mbps mode, Adjacent (+2 MHz) interference	..	..	..	dB
$C/I_{1M,-3MHz}$	1 Mbps mode, Adjacent (-3 MHz) interference	..	..	..	dB
$C/I_{1M,+3MHz}$	1 Mbps mode, Adjacent (+3 MHz) interference	..	..	..	dB
$C/I_{1M,\pm6MHz}$	1 Mbps mode, Adjacent (≥6 MHz) interference	..	..	..	dB
$C/I_{1MBLE,co\text{-}channel}$	1 Mbps Bluetooth LE mode, co-channel interference		6.9		dB
$C/I_{1MBLE,-1MHz}$	1 Mbps Bluetooth LE mode, Adjacent (-1 MHz) interference		-2.6		dB
$C/I_{1MBLE,+1MHz}$	1 Mbps Bluetooth LE mode, Adjacent (+1 MHz) interference		-8.5		dB
$C/I_{1MBLE,-2MHz}$	1 Mbps Bluetooth LE mode, Adjacent (-2 MHz) interference		-27		dB
$C/I_{1MBLE,+2MHz}$	1 Mbps Bluetooth LE mode, Adjacent (+2 MHz) interference		-45		dB
$C/I_{1MBLE,>3MHz}$	1 Mbps Bluetooth LE mode, Adjacent (≥3 MHz) interference		-50		dB
$C/I_{1MBLE,image}$	Image frequency interference		-27		dB
$C/I_{1MBLE,image,1MHz}$	Adjacent (1 MHz) interference to in-band image frequency		-41		dB
$C/I_{2M,co\text{-}channel}$	2 Mbps mode, co-channel interference	..	..	..	dB
$C/I_{2M,-2MHz}$	2 Mbps mode, Adjacent (-2 MHz) interference	..	..	..	dB
$C/I_{2M,+2MHz}$	2 Mbps mode, Adjacent (+2 MHz) interference	..	..	..	dB
$C/I_{2M,-4MHz}$	2 Mbps mode, Adjacent (-4 MHz) interference	..	..	..	dB
$C/I_{2M,+4MHz}$	2 Mbps mode, Adjacent (+4 MHz) interference	..	..	..	dB

[15] Desired signal level at P_{IN} = -67 dBm. One interferer is used, having equal modulation as the desired signal. The input power of the interferer where the sensitivity equals BER = 1E-4 is presented.

Symbol	Description	Min.	Typ.	Max.	Units
$C/I_{2M,-6MHz}$	2 Mbps mode, Adjacent (-6 MHz) interference	..	..	..	dB
$C/I_{2M,+6MHz}$	2 Mbps mode, Adjacent (+6 MHz) interference	..	..	..	dB
$C/I_{2M,\geq 12MHz}$	2 Mbps mode, Adjacent (≥12 MHz) interference	..	..	..	dB
$C/I_{2MBLE,co\text{-}channel}$	2 Mbps Bluetooth LE mode, co-channel interference		7.1		dB
$C/I_{2MBLE,-2MHz}$	2 Mbps Bluetooth LE mode, Adjacent (-2 MHz) interference		-2		dB
$C/I_{2MBLE,+2MHz}$	2 Mbps Bluetooth LE mode, Adjacent (+2 MHz) interference		-11		dB
$C/I_{2MBLE,-4MHz}$	2 Mbps Bluetooth LE mode, Adjacent (-4 MHz) interference		-22		dB
$C/I_{2MBLE,+4MHz}$	2 Mbps Bluetooth LE mode, Adjacent (+4 MHz) interference		-47		dB
$C/I_{2MBLE,\geq 6MHz}$	2 Mbps Bluetooth LE mode, Adjacent (≥6 MHz) interference		-54		dB
$C/I_{2MBLE,image}$	Image frequency interference		-22		dB
$C/I_{2MBLE,image,\ 2MHz}$	Adjacent (2 MHz) interference to in-band image frequency		-42		dB
$C/I_{125k\ BLE\ LR,co\text{-}channel}$	125 kbps Bluetooth LE LR mode, co-channel interference	..	..	..	dB
$C/I_{125k\ BLE\ LR,-1MHz}$	125 kbps Bluetooth LE LR mode, Adjacent (-1 MHz) interference	..	..	..	dB
$C/I_{125k\ BLE\ LR,+1MHz}$	125 kbps Bluetooth LE LR mode, Adjacent (+1 MHz) interference	..	..	..	dB
$C/I_{125k\ BLE\ LR,-2MHz}$	125 kbps Bluetooth LE LR mode, Adjacent (-2 MHz) interference	..	..	..	dB
$C/I_{125k\ BLE\ LR,+2MHz}$	125 kbps Bluetooth LE LR mode, Adjacent (+2 MHz) interference	..	..	..	dB
$C/I_{125k\ BLE\ LR,>3MHz}$	125 kbps Bluetooth LE LR mode, Adjacent (≥3 MHz) interference	..	..	..	dB
$C/I_{125k\ BLE\ LR,image}$	Image frequency interference	..	..	..	dB
$C/I_{IEEE\ 802.15.4,-5MHz}$	IEEE 802.15.4 mode, Adjacent (-5 MHz) rejection		-33		dB
$C/I_{IEEE\ 802.15.4,+5MHz}$	IEEE 802.15.4 mode, Adjacent (+5 MHz) rejection		-38		dB
$C/I_{IEEE\ 802.15.4,\ \pm 10MHz}$	IEEE 802.15.4 mode, Alternate (±10 MHz) rejection		-50		dB

7.26.16.7 RX intermodulation

RX intermodulation. Desired signal level at P_{IN} = -64 dBm. Two interferers with equal input power are used. The interferer closest in frequency is not modulated, the other interferer is modulated equal with the desired signal. The input power of the interferers where the sensitivity equals BER = 1E-3 is presented.

Symbol	Description	Min.	Typ.	Max.	Units
$P_{IMD,5TH,1M}$	IMD performance, 1 Mbps, 5th offset channel, packet length ≤ 37 bytes	..	..	..	dBm
$P_{IMD,5TH,1M,BLE}$	IMD performance, Bluetooth LE 1 Mbps, 5th offset channel, packet length ≤ 37 bytes		-26		dBm
$P_{IMD,5TH,2M}$	IMD performance, 2 Mbps, 5th offset channel, packet length ≤ 37 bytes	..	..	..	dBm
$P_{IMD,5TH,2M,BLE}$	IMD performance, Bluetooth LE 2 Mbps, 5th offset channel, packet length ≤ 37 bytes		-22		dBm

7.26.16.8 Radio timing

Symbol	Description	Min.	Typ.	Max.	Units
$t_{TXEN,BLE,1M}$	Time between TXEN task and READY event after channel FREQUENCY configured (1 Mbps Bluetooth LE and 150 µs TIFS)		140		µs

Symbol	Description	Min.	Typ.	Max.	Units
$t_{TXEN,FAST,BLE,1M}$	Time between TXEN task and READY event after channel FREQUENCY configured (1 Mbps Bluetooth LE with fast ramp-up and 150 µs TIFS)		40		µs
$t_{TXDIS,BLE,1M}$	When in TX, delay between DISABLE task and DISABLED event for MODE = Nrf_1Mbit and MODE = Ble_1Mbit		6		µs
$t_{RXEN,BLE,1M}$	Time between the RXEN task and READY event after channel FREQUENCY configured (1 Mbps Bluetooth LE)		140		µs
$t_{RXEN,FAST,BLE,1M}$	Time between the RXEN task and READY event after channel FREQUENCY configured (1 Mbps Bluetooth LE with fast ramp-up)		40		µs
$t_{RXDIS,BLE,1M}$	When in RX, delay between DISABLE task and DISABLED event for MODE = Nrf_1Mbit and MODE = Ble_1Mbit		0		µs
$t_{TXDIS,BLE,2M}$	When in TX, delay between DISABLE task and DISABLED event for MODE = Nrf_2Mbit and MODE = Ble_2Mbit		4		µs
$t_{RXDIS,BLE,2M}$	When in RX, delay between DISABLE task and DISABLED event for MODE = Nrf_2Mbit and MODE = Ble_2Mbit		0		µs
$t_{TXEN,IEEE\ 802.15.4}$	Time between TXEN task and READY event after channel FREQUENCY configured (IEEE 802.15.4 mode)		130		µs
$t_{TXEN,FAST,IEEE\ 802.15.4}$	Time between TXEN task and READY event after channel FREQUENCY configured (IEEE 802.15.4 mode with fast ramp-up)		40		µs
$t_{TXDIS,IEEE\ 802.15.4}$	When in TX, delay between DISABLE task and DISABLED event (IEEE 802.15.4 mode)		21		µs
$t_{RXEN,IEEE\ 802.15.4}$	Time between the RXEN task and READY event after channel FREQUENCY configured (IEEE 802.15.4 mode)		130		µs
$t_{RXEN,FAST,IEEE\ 802.15.4}$	Time between the RXEN task and READY event after channel FREQUENCY configured (IEEE 802.15.4 mode with fast ramp-up)		40		µs
$t_{RXDIS,IEEE\ 802.15.4}$	When in RX, delay between DISABLE task and DISABLED event (IEEE 802.15.4 mode)		0.5		µs
$t_{RX\text{-}to\text{-}TX\ turnaround}$	Maximum TX-to-RX or RX-to-TX turnaround time in IEEE 802.15.4 mode		40		µs

7.26.16.9 Received signal strength indicator (RSSI) specifications

Symbol	Description	Min.	Typ.	Max.	Units
$RSSI_{ACC}$	RSSI accuracy		±2		dB
$RSSI_{RESOLUTION}$	RSSI resolution		1		dB
$RSSI_{PERIOD}$	RSSI sampling time from RSSI_START task		0.25		µs
$RSSI_{SETTLE}$	RSSI settling time after signal level change		15		µs

7.26.16.10 Jitter

Symbol	Description	Min.	Typ.	Max.	Units
$t_{DISABLEDJITTER}$	Jitter on DISABLED event relative to END event when shortcut between END and DISABLE is enabled		0.25		µs
$t_{READYJITTER}$	Jitter on READY event relative to TXEN and RXEN task		0.25		µs

7.26.16.11 IEEE 802.15.4 mode energy detection constants

Symbol	Description	Min.	Typ.	Max.	Units
ED_RSSISCALE	Scaling value when converting between hardware-reported value and dBm		5		
ED_RSSIOFFS	Offset value when converting between hardware-reported value and dBm		-93		

7.27 RNG — Random number generator

The Random number generator (RNG) generates true non-deterministic random numbers based on internal thermal noise that are suitable for cryptographic purposes. The RNG does not require a seed value.

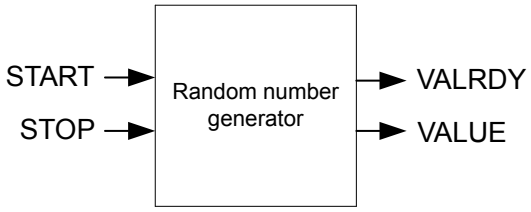


Figure 173: Random number generator

The RNG is started by triggering the START task and stopped by triggering the STOP task. When started, new random numbers are generated continuously and written to the VALUE register when ready. A VALRDY event is generated for every new random number that is written to the VALUE register. This means that after a VALRDY event is generated, the CPU has the time until the next VALRDY event to read out the random number from the VALUE register before it is overwritten by a new random number.

7.27.1 Bias correction

A bias correction algorithm is employed on the internal bit stream to remove any bias toward `1` or `0`. The bits are then queued into an eight-bit register for parallel readout from the VALUE register.

It is possible to enable bias correction in the CONFIG register. This will result in slower value generation, but will ensure a statistically uniform distribution of the random values.

7.27.2 Speed

The time needed to generate one random byte of data is unpredictable, and may vary from one byte to the next. This is especially true when bias correction is enabled.

7.27.3 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x41009000	NETWORK	RNG	RNG	NS	NA	Random number generator	

Table 132: Instances

Register	Offset	Security	Description
TASKS_START	0x000		Task starting the random number generator
TASKS_STOP	0x004		Task stopping the random number generator
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP

Register	Offset	Security	Description
EVENTS_VALRDY	0x100		Event being generated for every new random number written to the VALUE register
PUBLISH_VALRDY	0x180		Publish configuration for event VALRDY
SHORTS	0x200		Shortcuts between local events and tasks
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
CONFIG	0x504		Configuration register
VALUE	0x508		Output random number

Table 133: Register overview

7.27.3.1 TASKS_START

Address offset: 0x000

Task starting the random number generator

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_START			Task starting the random number generator
			Trigger	1	Trigger task

7.27.3.2 TASKS_STOP

Address offset: 0x004

Task stopping the random number generator

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Task stopping the random number generator
			Trigger	1	Trigger task

7.27.3.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task START

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	B A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task START will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.27.3.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.27.3.5 EVENTS_VALRDY

Address offset: 0x100

Event being generated for every new random number written to the VALUE register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_VALRDY			Event being generated for every new random number written to the VALUE register
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.27.3.6 PUBLISH_VALRDY

Address offset: 0x180

Publish configuration for event VALRDY

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event VALRDY will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.27.3.7 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	VALRDY_STOP			Shortcut between event VALRDY and task STOP
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.27.3.8 INTENSET

Address offset: 0x304

Enable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	VALRDY			Write '1' to enable interrupt for event VALRDY
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.27.3.9 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	VALRDY			Write '1' to disable interrupt for event VALRDY
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.27.3.10 CONFIG

Address offset: 0x504

Configuration register

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	DERCEN			Bias correction
			Disabled	0	Disabled
			Enabled	1	Enabled

7.27.3.11 VALUE

Address offset: 0x508

Output random number

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A A A A A A A A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	VALUE		[0..255]	Generated random number

7.27.4 Electrical specification

7.27.4.1 RNG Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
$t_{RNG,START}$	Time from setting the START task to generation begins. This is a one-time delay on START signal and does not apply between samples.		128		µs
$t_{RNG,RAW}$	Run time per byte without bias correction. Uniform distribution of 0 and 1 is not guaranteed.		32		µs
$t_{RNG,BC}$	Run time per byte with bias correction. Uniform distribution of 0 and 1 is guaranteed. Time to generate a byte cannot be guaranteed.			122	µs

7.28 RTC — Real-time counter

The real-time counter (RTC) module provides a generic, low-power timer on the low frequency clock source (LFCLK).

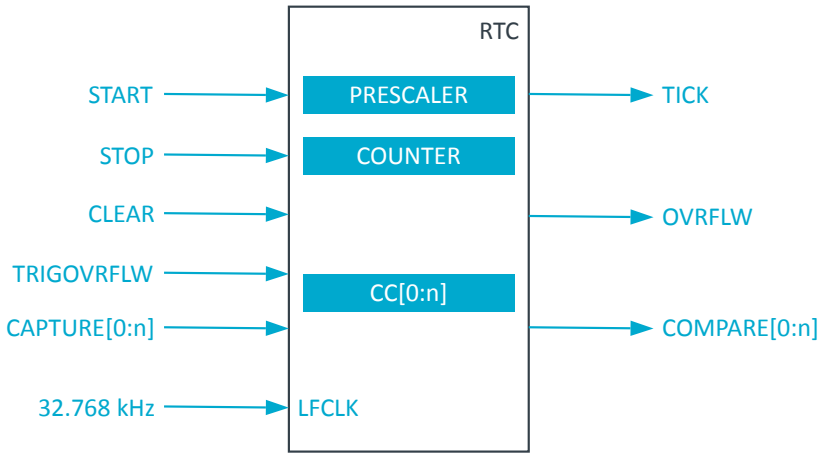


Figure 174: RTC block diagram

The RTC module features a 24-bit COUNTER, a 12-bit (1/X) prescaler, capture/compare registers, and a tick event generator.

7.28.1 Clock source

The RTC will run off the LFCLK.

When started, the RTC will automatically request the LFCLK source with RC oscillator if the LFCLK is not already running.

See CLOCK — Clock control on page 69 for more information about clock sources.

7.28.2 Resolution versus overflow and the prescaler

The relationship between the prescaler, counter resolution, and overflow is summarized in the following table.

Prescaler	Counter resolution	Overflow
0	30.517 µs	512 seconds
2^8-1	7812.5 µs	131072 seconds
2^{12}-1	125 ms	582.542 hours

Table 134: RTC resolution versus overflow

The counter increment frequency is given by the following equation:

$$f_{RTC}\ [kHz] = 32.768\ /\ (PRESCALER + 1\ )$$

The PRESCALER register can only be written when the RTC is stopped.

The prescaler is restarted on tasks START, CLEAR and TRIGOVRFLW. That is, the prescaler value is latched to an internal register (<<PRESC>>) on these tasks.

Examples:

1. Desired COUNTER frequency 100 Hz (10 ms counter period)

   PRESCALER = round(32.768 kHz / 100 Hz) - 1 = 327

   f_{RTC} = 99.9 Hz

   10009.576 µs counter period
2. Desired COUNTER frequency 8 Hz (125 ms counter period)

   PRESCALER = round(32.768 kHz / 8 Hz) – 1 = 4095

   f_{RTC} = 8 Hz

   125 ms counter period

7.28.3 Counter register

The internal <<COUNTER>> register increments on LFCLK when the internal PRESCALER register (<<PRESC>>) is 0x00. <<PRESC>> is reloaded from the PRESCALER register. If enabled, the TICK event occurs on each increment of the COUNTER.

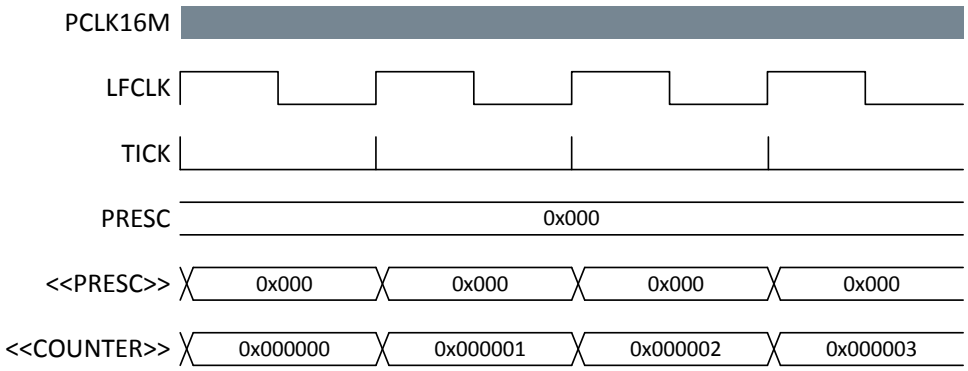


Figure 175: Timing diagram - COUNTER_PRESCALER_0

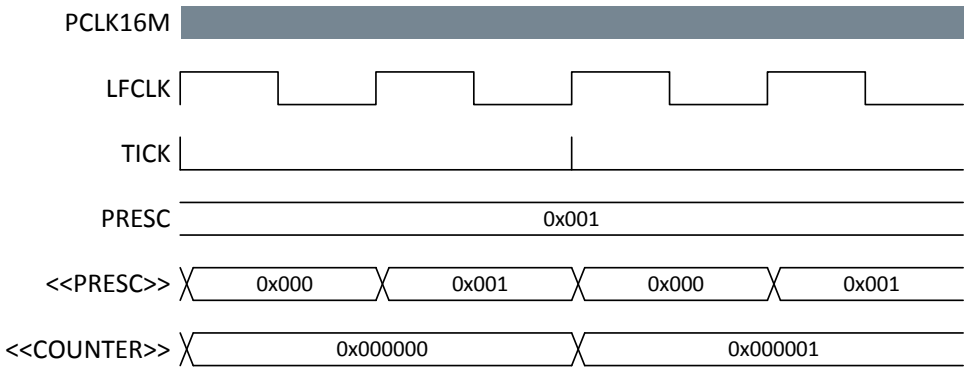


Figure 176: Timing diagram - COUNTER_PRESCALER_1

7.28.3.1 Reading the counter register

To read the COUNTER register, the internal <<COUNTER>> value is sampled.

To ensure that the <<COUNTER>> is safely sampled (considering that an LFCLK transition may occur during a read), the CPU and core memory bus are halted for four PCLK16M cycles. In addition, the read takes the CPU two PCLK16M cycles, resulting in the COUNTER register read taking maximum six PCLK16M clock cycles.

7.28.4 Overflow

An OVRFLW event is generated on COUNTER register overflow (overflowing from `0xFFFFFF` to `0`).

The TRIGOVRFLW task will set the COUNTER value to `0xFFFFF0`, to allow software test of the overflow condition.

Note: The OVRFLW event is disabled by default.

7.28.5 Tick event

The TICK event enables low-power tickless RTOS implementation, as it optionally provides a regular interrupt source for an RTOS with no need for use of the ARM SysTick feature.

Using the TICK event, rather than the SysTick, allows the CPU to be powered down while keeping RTOS scheduling active.

Note: The TICK event is disabled by default.

7.28.6 Event control

To optimize the RTC power consumption, events in the RTC can be individually disabled to prevent PCLK16M and HFCLK from being requested when those events are triggered. This is managed using the EVTEN register.

This means that the RTC implements a slightly different task and event system compared to the standard system described in Peripheral interface on page 146. The RTC task and event system is illustrated in the following figure.

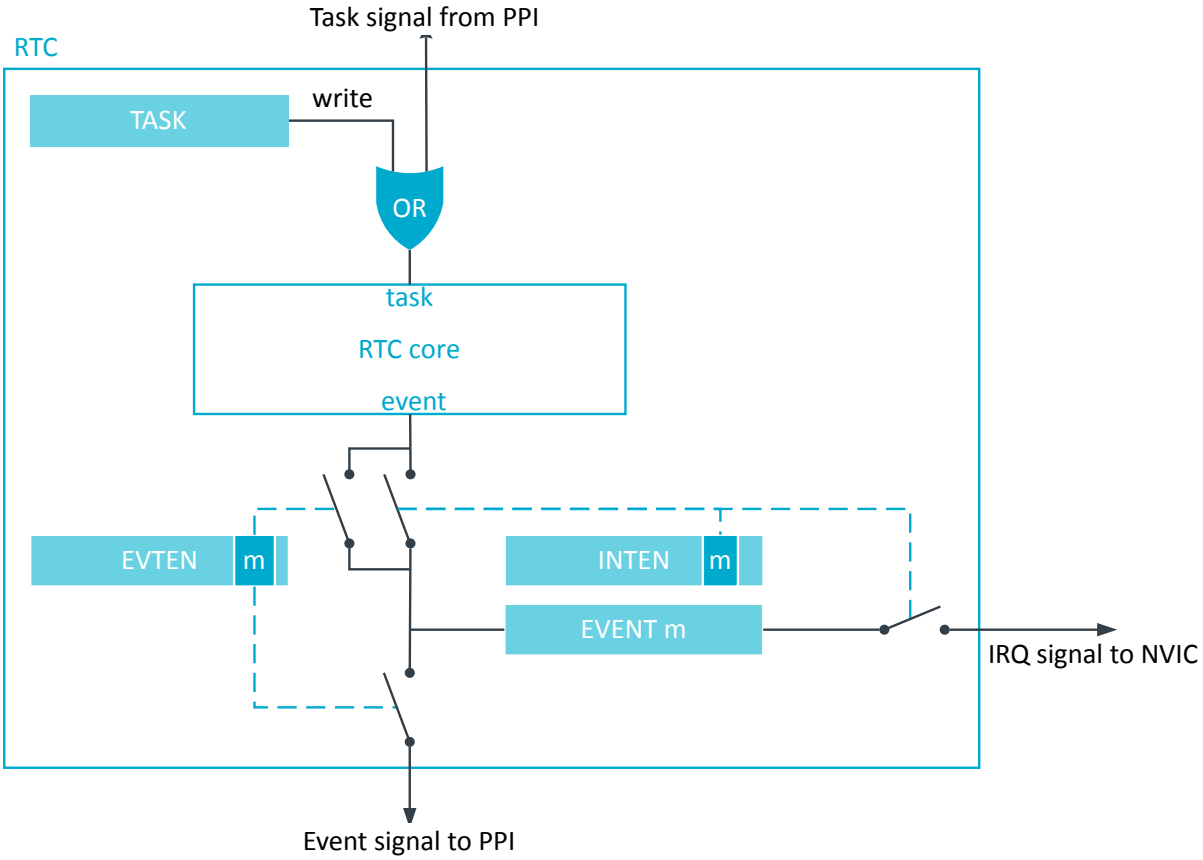


Figure 177: Tasks, events, and interrupts in the RTC

7.28.7 Capture

The RTC implements one capture task for every available capture/compare register.

Every time TASKS_CAPTURE[n] is triggered, <<COUNTER>> is copied to the corresponding CC[n] register.

If the CAPTURE and CLEAR tasks are triggered at the same time, the CAPTURE task will be prioritized. This means that the CC[n] register for the corresponding CAPTURE[n] task will be set to the captured value before the counter is reset. There is a delay of 6 PCLK16M periods from when the TASKS_CAPTURE[n] is triggered until the corresponding CC[n] register is updated.

The CAPTURE[n] tasks will not generate COMPARE[n] events, even though CC[n] will then equal the COUNTER.

7.28.8 Compare

The RTC implements one COMPARE event for every available compare register.

When the COUNTER is incremented and then becomes equal to the value specified in the register CC[n], the corresponding compare event COMPARE[n] is generated.

When writing a CC[n] register, the RTC COMPARE event exhibits several behaviors. See the following figures for more information.

If a CC value is 0 when a CLEAR task is set, this will not trigger a COMPARE event.

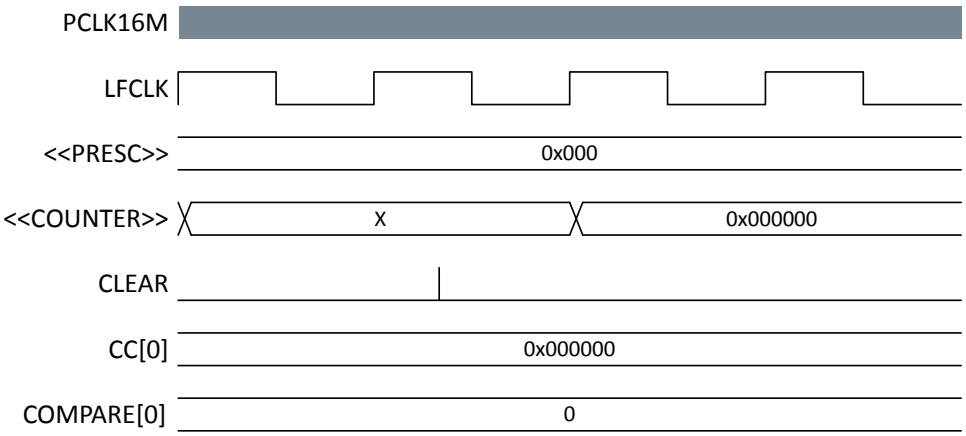


Figure 178: Timing diagram - COMPARE_CLEAR

If a CC value is N and the COUNTER value is N when the START task is set, this will not trigger a COMPARE event.

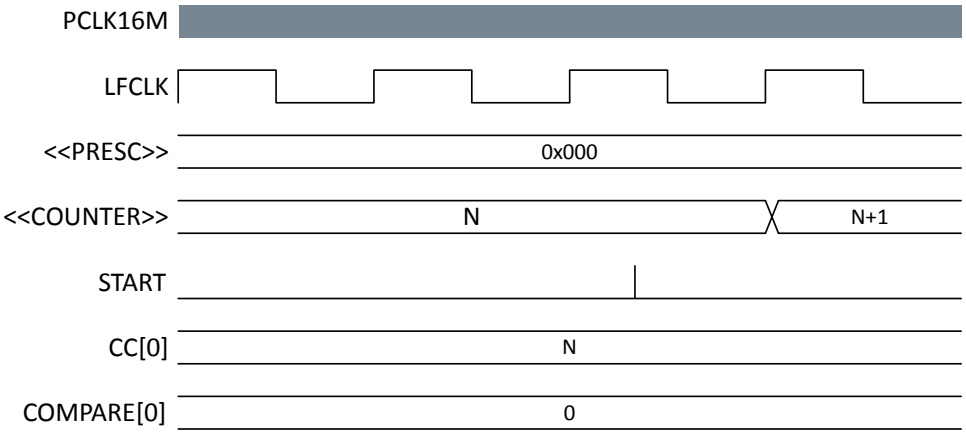


Figure 179: Timing diagram - COMPARE_START

A COMPARE event occurs when a CC value is N, and the COUNTER value transitions from N-1 to N.

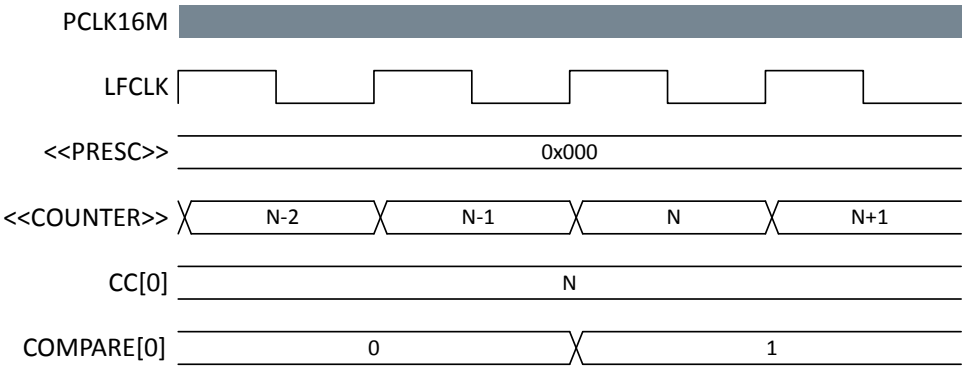


Figure 180: Timing diagram - COMPARE

If the COUNTER value is N, writing N+2 to a CC register is guaranteed to trigger a COMPARE event at N+2.

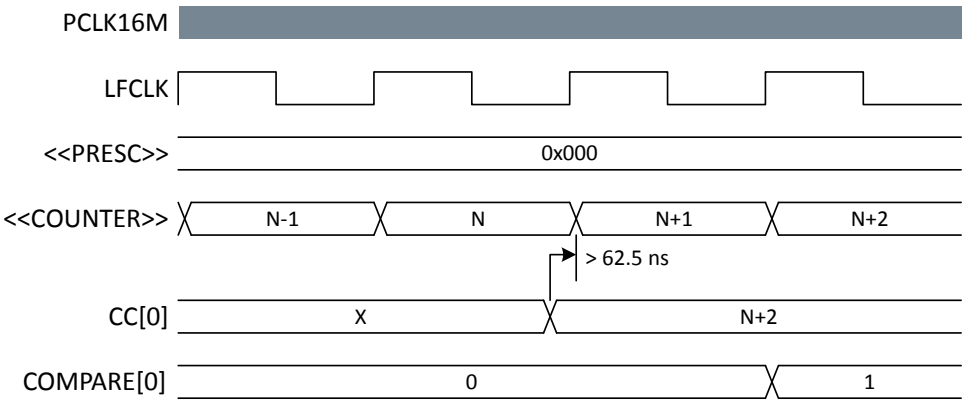


Figure 181: Timing diagram - COMPARE_N+2

If the COUNTER value is N, writing N or N+1 to a CC register may not trigger a COMPARE event.

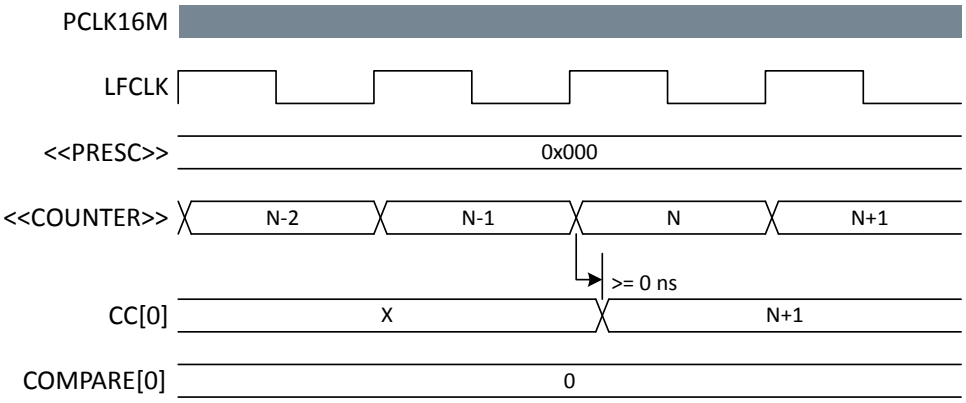


Figure 182: Timing diagram - COMPARE_N+1

If the COUNTER value is N, and the current CC value is N+1 or N+2 when a new CC value is written, a match may trigger on the previous CC value before the new value takes effect. If the current CC value is greater than N+2 when the new value is written, there will be no event due to the old value.

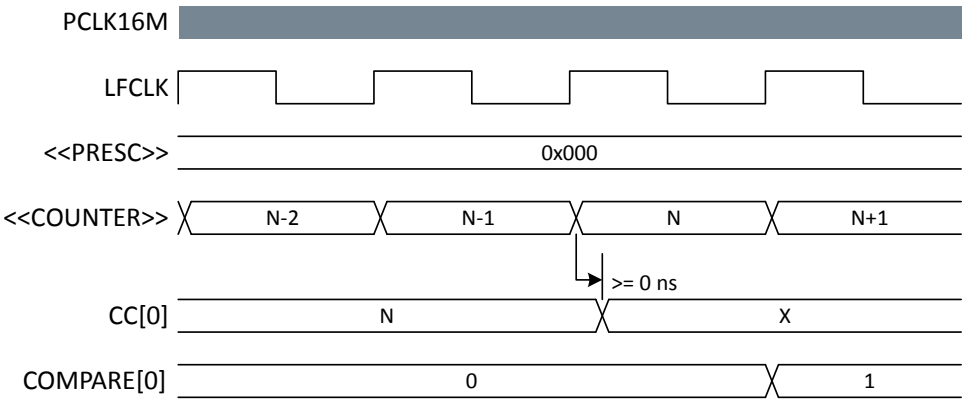


Figure 183: Timing diagram - COMPARE_N-1

If the COMPARE[i]_CLEAR short is enabled, the COUNTER will be cleared one LFClk after the COMPARE event.

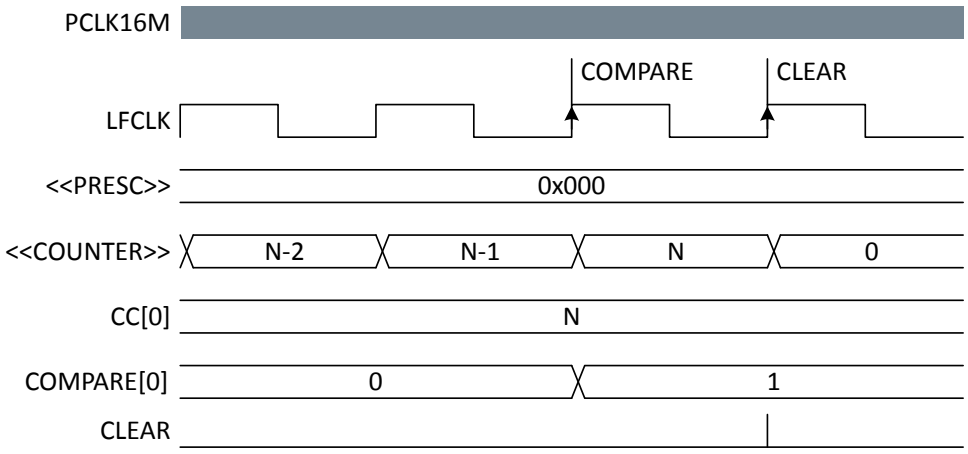


Figure 184: Timing diagram - COMPARE_CLEAR

7.28.9 Task and event jitter/delay

Jitter or delay in the RTC, is due to the peripheral clock being a low frequency clock (LFCLK), which is not synchronous to the faster PCLK16M.

Registers in the peripheral interface that are part of the PCLK16M domain, have a set of mirrored registers in the LFCLK domain. For example, the COUNTER value accessible from the CPU is in the PCLK16M domain, and is latched on a read from an internal COUNTER register in the LFCLK domain. The COUNTER register is modified each time the RTC ticks. The registers are synchronised between the two clock domains (PCLK16M and LFCLK).

CLEAR and STOP (and TRIGOVRFLW, which is not shown) will be delayed as long as it takes for the peripheral to clock a falling edge and a rising edge of the LFCLK. This is between 15.2585 µs and 45.7755 µs – rounded to 15 µs and 46 µs for the remainder of the section.

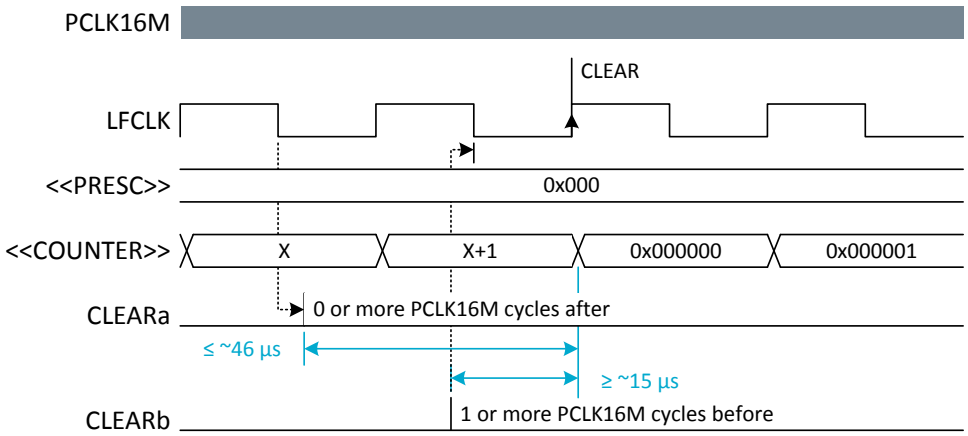


Figure 185: Timing diagram - DELAY_CLEAR

When a STOP task is triggered, the PCLK16M domain will immediately prevent the generation of any EVENTS from the RTC. However, as seen in the following figure, the COUNTER value can still increment one final time.

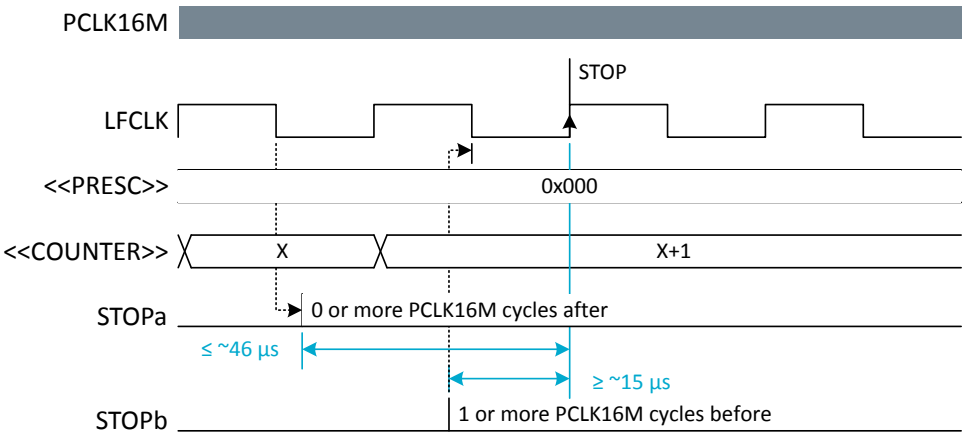


Figure 186: Timing diagram - DELAY_STOP

The START task will start the RTC. Assuming that the LFCLK was previously running and stable, the first increment of COUNTER (and instance of TICK event) will be typically after 30.5 µs ±15 µs. Additional delay will occur if the RTC is started before the LFCLK is running, see CLOCK — Clock control on page 69 for LFLK startup times. The software should therefore wait for the first TICK if it has to make sure that the RTC is running. Sending a TRIGOVRFLW task sets the COUNTER to a value close to overflow. However, since the update of COUNTER relies on a stable LFCLK, sending this task while LFCLK is not running will also add additional delay as previously described. The figures show the smallest and largest delays on the START task, appearing as a ±15 µs jitter on the first COUNTER increment.

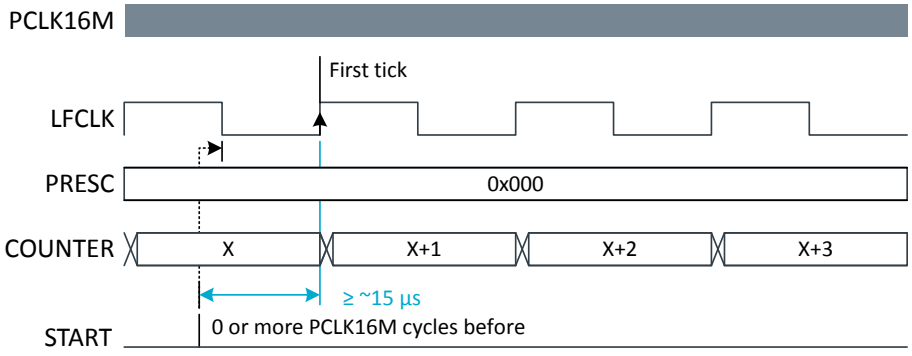


Figure 187: Timing diagram - JITTER_START-

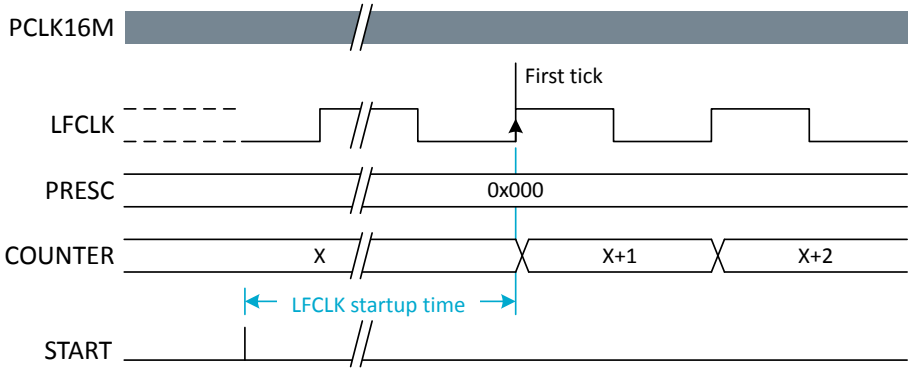


Figure 188: Timing diagram - JITTER_START+

The following tables summarize jitter introduced for tasks and events. Any 32.768 kHz clock jitter will come in addition to these numbers.

Task	Delay
CLEAR, START, STOP, TRIGOVRFLOW	+15 to 46 µs

Table 135: RTC jitter magnitudes on tasks

Operation/Function	Jitter
START to COUNTER increment	± 15 µs
COMPARE to COMPARE [16]	± 62.5 ns

Table 136: RTC jitter magnitudes on events

7.28.10 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50014000 0x40014000	APPLICATION	RTC	RTC0 : S RTC0 : NS	US	NA	Real time counter 0	
0x50015000 0x40015000	APPLICATION	RTC	RTC1 : S RTC1 : NS	US	NA	Real time counter 1	
0x41011000	NETWORK	RTC	RTC0	NS	NA	Real-time counter 0	
0x41016000	NETWORK	RTC	RTC1	NS	NA	Real-time counter 1	

Table 137: Instances

Register	Offset	Security	Description
TASKS_START	0x000		Start RTC counter
TASKS_STOP	0x004		Stop RTC counter
TASKS_CLEAR	0x008		Clear RTC counter
TASKS_TRIGOVRFLW	0x00C		Set counter to 0xFFFFF0
TASKS_CAPTURE[n]	0x040		Capture RTC counter to CC[n] register
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_CLEAR	0x088		Subscribe configuration for task CLEAR
SUBSCRIBE_TRIGOVRFLW	0x08C		Subscribe configuration for task TRIGOVRFLW
SUBSCRIBE_CAPTURE[n]	0x0C0		Subscribe configuration for task CAPTURE[n]
EVENTS_TICK	0x100		Event on counter increment
EVENTS_OVRFLW	0x104		Event on counter overflow
EVENTS_COMPARE[n]	0x140		Compare event on CC[n] match
PUBLISH_TICK	0x180		Publish configuration for event TICK
PUBLISH_OVRFLW	0x184		Publish configuration for event OVRFLW
PUBLISH_COMPARE[n]	0x1C0		Publish configuration for event COMPARE[n]
SHORTS	0x200		Shortcuts between local events and tasks
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
EVTEN	0x340		Enable or disable event routing
EVTENSET	0x344		Enable event routing
EVTENCLR	0x348		Disable event routing
COUNTER	0x504		Current counter value
PRESCALER	0x508		12-bit prescaler for counter frequency (32768 / (PRESCALER + 1)). Must be written when RTC is stopped.

[16] Assumes RTC runs continuously between these events.

Register	Offset	Security	Description
CC[n]	0x540		Compare register n

Table 138: Register overview

7.28.10.1 TASKS_START

Address offset: 0x000

Start RTC counter

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_START			Start RTC counter
			Trigger	1	Trigger task

7.28.10.2 TASKS_STOP

Address offset: 0x004

Stop RTC counter

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Stop RTC counter
			Trigger	1	Trigger task

7.28.10.3 TASKS_CLEAR

Address offset: 0x008

Clear RTC counter

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_CLEAR			Clear RTC counter
			Trigger	1	Trigger task

7.28.10.4 TASKS_TRIGOVRFLW

Address offset: 0x00C

Set counter to 0xFFFFF0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_TRIGOVRFLW			Set counter to 0xFFFFF0
			Trigger	1	Trigger task

7.28.10.5 TASKS_CAPTURE[n] (n=0..3)

Address offset: 0x040 + (n × 0x4)

Capture RTC counter to CC[n] register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_CAPTURE			Capture RTC counter to CC[n] register
			Trigger	1	Trigger task

7.28.10.6 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task START

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task START will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.28.10.7 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.28.10.8 SUBSCRIBE_CLEAR

Address offset: 0x088

Subscribe configuration for task CLEAR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task CLEAR will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.28.10.9 SUBSCRIBE_TRIGOVRFLW

Address offset: 0x08C

Subscribe configuration for task TRIGOVRFLW

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task TRIGOVRFLW will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.28.10.10 SUBSCRIBE_CAPTURE[n] (n=0..3)

Address offset: 0x0C0 + (n × 0x4)

Subscribe configuration for task CAPTURE[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task CAPTURE[n] will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.28.10.11 EVENTS_TICK

Address offset: 0x100

Event on counter increment

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_TICK			Event on counter increment
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.28.10.12 EVENTS_OVRFLW

Address offset: 0x104

Event on counter overflow

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_OVRFLW			Event on counter overflow
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.28.10.13 EVENTS_COMPARE[n] (n=0..3)

Address offset: 0x140 + (n × 0x4)

Compare event on CC[n] match

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_COMPARE			Compare event on CC[n] match
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.28.10.14 PUBLISH_TICK

Address offset: 0x180

Publish configuration for event TICK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event TICK will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.28.10.15 PUBLISH_OVRFLW

Address offset: 0x184

Publish configuration for event OVRFLW

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event OVRFLW will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.28.10.16 PUBLISH_COMPARE[n] (n=0..3)

Address offset: 0x1C0 + (n × 0x4)

Publish configuration for event COMPARE[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event COMPARE[n] will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.28.10.17 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-D	RW	COMPARE[i]_CLEAR (i=0..3)			Shortcut between event COMPARE[i] and task CLEAR
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.28.10.18 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TICK			Write '1' to enable interrupt for event TICK
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	OVRFLW			Write '1' to enable interrupt for event OVRFLW
			Set	1	Enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C-F	RW	COMPARE[i] (i=0..3)			Write '1' to enable interrupt for event COMPARE[i]
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.28.10.19 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TICK			Write '1' to disable interrupt for event TICK
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	OVRFLW			Write '1' to disable interrupt for event OVRFLW
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C-F	RW	COMPARE[i] (i=0..3)			Write '1' to disable interrupt for event COMPARE[i]
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.28.10.20 EVTEN

Address offset: 0x340

Enable or disable event routing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TICK			Enable or disable event routing for event TICK
			Disabled	0	Disable
			Enabled	1	Enable
B	RW	OVRFLW			Enable or disable event routing for event OVRFLW
			Disabled	0	Disable
			Enabled	1	Enable
C-F	RW	COMPARE[i] (i=0..3)			Enable or disable event routing for event COMPARE[i]
			Disabled	0	Disable
			Enabled	1	Enable

7.28.10.21 EVTENSET

Address offset: 0x344

Enable event routing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TICK			Write '1' to enable event routing for event TICK
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
			Set	1	Enable
B	RW	OVRFLW			Write '1' to enable event routing for event OVRFLW
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
			Set	1	Enable
C-F	RW	COMPARE[i] (i=0..3)			Write '1' to enable event routing for event COMPARE[i]
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
			Set	1	Enable

7.28.10.22 EVTENCLR

Address offset: 0x348

Disable event routing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TICK			Write '1' to disable event routing for event TICK
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
			Clear	1	Disable
B	RW	OVRFLW			Write '1' to disable event routing for event OVRFLW
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
			Clear	1	Disable
C-F	RW	COMPARE[i] (i=0..3)			Write '1' to disable event routing for event COMPARE[i]
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
			Clear	1	Disable

7.28.10.23 COUNTER

Address offset: 0x504

Current counter value

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	COUNTER			Counter value

7.28.10.24 PRESCALER

Address offset: 0x508

12-bit prescaler for counter frequency (32768 / (PRESCALER + 1)). Must be written when RTC is stopped.

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PRESCALER			Prescaler value

7.28.10.25 CC[n] (n=0..3)

Address offset: 0x540 + (n × 0x4)

Compare register n

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	COMPARE			Compare value

7.28.11 Electrical specification

7.29 SAADC — Successive approximation analog-to-digital converter

SAADC is a differential successive approximation register (SAR) analog-to-digital converter (ADC).

The main features of SAADC are the following:

- 8/10/12-bit resolution, 14-bit resolution with oversampling
- Multiple analog inputs:
  - **AIN0** to **AIN7** pins
  - **VDD** pin
  - VDDHDIV5 (through **VDDH** pin)
- Up to eight input channels:
  - One channel per single-ended input and two channels per differential input
  - Scan mode can be configured with both single-ended channels and differential channels
  - Each channel can be configured to select any of the above analog inputs
- Full scale input range (0 to VDD)
- Sampling triggered via a task from software or a PPI channel for full flexibility on sample frequency source from a low-power 32.768 kHz RTC or more accurate 1/16 MHz timers

- One-shot conversion mode to sample a single channel
- Scan mode to sample a series of channels in sequence with configurable sample delay
- Support for direct sample transfer to RAM using EasyDMA
- Interrupts on single sample and full buffer events
- Samples stored as 16-bit two's complement values for differential and single-ended sampling
- Continuous sampling without the need of an external timer
- Internal resistor string
- On-the-fly limit checking

7.29.1 Shared resources

The ADC can coexist with COMP and other peripherals using one of **AIN0**-**AIN7**, provided these are assigned to different pins.

It is not recommended to select the same analog input pin for both modules.

7.29.2 Overview

The ADC supports up to eight external analog input channels. It can be operated in One-shot mode with sampling under software control, or Continuous mode with a programmable sampling rate.

The analog inputs can be configured as eight single-ended inputs, four differential inputs, or a combination of these. Each channel can be configured to select one of the following:

- **AIN0** to **AIN7** pins
- **VDD** pin
- **VDDHDIV5** (through VDDH pin)

Channels can be sampled individually in One-shot or Continuous sampling modes. Using Scan mode, multiple channels can be sampled in sequence. Channels can also be oversampled to improve noise performance.

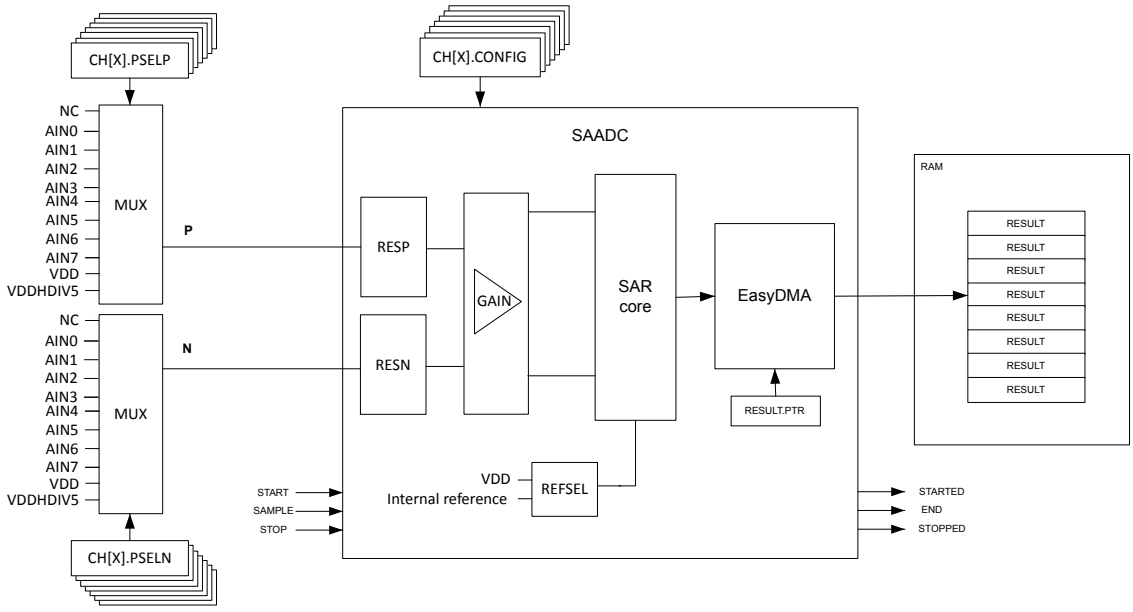


Figure 189: Simplified ADC block diagram

Internally, the ADC is always a differential analog-to-digital converter, but by default it is configured with single-ended input `SE` in the MODE field of the CH[n].CONFIG register. In Single-ended (SE) mode, the negative input will be shorted to ground internally.

In Single-ended mode, the assumption is that the internal ground of the ADC is the same as the external ground that the measured voltage is referred to. The ADC is thus sensitive to ground bounce on the PCB.

This can be reduced by configuring SAADC to use differential measurent setting the MODE field of the CH[n].CONFIG register to `Diff` (differential).

7.29.3 Digital output

The output result of the ADC depends on the settings in the CH[n].CONFIG and RESOLUTION registers as shown in the following equation.

$$RESULT = [V(P) - V(N)] * GAIN/REFERENCE * 2^{(RESOLUTION - m)}$$

Variable	Description
V(P)	Voltage at input P
V(N)	Voltage at input N
GAIN	Selected gain setting
m	Mode setting (Use m=0 if CONFIG.MODE=SE, or m=1 if CONFIG.MODE=Diff)
REFERENCE	Selected reference voltage

Table 139: Equation variables

The result generated by the ADC will deviate from the expected due DC errors like offset, gain, differential non-linearity (DNL), and integral non-linearity (INL). See Electrical specification for details on these parameters. The result can also vary due to AC errors like non-linearities in the GAIN block, settling errors due to high source impedance and sampling jitter. For battery measurement, the DC errors are most noticeable.

The ADC has a wide selection of gains controlled in the GAIN field of the CH[n].CONFIG register. If CH[n].CONFIG.REFSEL=0, the input range of the ADC core is nominally ±0.6 V differential, and the input must be scaled accordingly.

The ADC has a temperature dependent offset. If the ADC is to operate over a large temperature range, it is recommended to run CALIBRATEOFFSET at regular intervals. The CALIBRATEDONE, DONE, and RESULTDONE events will be generated when the calibration has completed.

7.29.4 Analog inputs and channels

Up to eight analog input channels, CH[n](n=0..7), can be configured.

See Shared resources on page 521 for shared input with comparators.

Any of the available channels can be enabled for the ADC to operate in One-shot mode. If more than one CH[n] is configured, the ADC enters Scan mode.

An analog input is selected as a positive converter input if CH[n].PSELP is set, setting CH[n].PSELP also enables the particular channel.

An analog input is selected as a negative converter input if CH[n].PSELN is set. The CH[n].PSELN register will have no effect unless Differential mode is enabled, see MODE field in the CH[n].CONFIG register.

If more than one of the CH[n].PSELP registers is set, the device enters Scan mode. Input selections in Scan mode are controlled by the CH[n].PSELP and CH[n].PSELN registers, where CH[n].PSELN is only used if the particular scan channel is specified as differential, see MODE field in the CH[n].CONFIG register.

Note: Channels selected for COMP cannot be used at the same time for ADC sampling, though channels not selected for use by these blocks can be used by the ADC.

7.29.5 Operation modes

The ADC input configuration supports One-shot mode, Continuous mode, and Scan mode.

The ADC indicates a single ongoing conversion via the register STATUS on page 541. During Scan mode, oversampling, or Continuous modes, more than a single conversion take place in the ADC. As a consequence, the value reflected in the STATUS register will toggle at the end of each single conversion.

Note: Scan mode and oversampling cannot be combined.

7.29.5.1 One-shot mode

One-shot operation is configured by enabling only one of the available channels defined by the CH[n].PSELP, CH[n].PSELN, and CH[n].CONFIG registers.

Once a SAMPLE task is triggered, the ADC starts to sample the input voltage. The CH[n].CONFIG.TACQ controls the acquisition time.

A DONE event signals that one sample has been taken.

In this mode, the RESULTDONE event has the same meaning as DONE when no oversampling takes place. Both events may occur before the actual value has been transferred into RAM by EasyDMA. For more information, see EasyDMA on page 524.

7.29.5.2 Continuous mode

Continuous sampling can be achieved by using the internal timer in the ADC, or triggering the SAMPLE task from one of the general purpose timers through the PPI system.

Care shall be taken to ensure that the sample rate fulfills the following criteria, depending on how many channels are active.

$$f_{SAMPLE} < 1/(t_{ACQ} + t_{conv})$$

The SAMPLERATE register can be used as a local timer instead of triggering individual SAMPLE tasks. When SAMPLERATE.MODE is set to Timers, it is sufficient to trigger SAMPLE task only once in order to start the SAADC and triggering the STOP task will stop sampling. The SAMPLERATE.CC field controls the sample rate.

The SAMPLERATE timer mode cannot be combined with Scan mode, and only one channel can be enabled in this mode.

A DONE event signals that one sample has been taken.

In this mode, the RESULTDONE event has the same meaning as DONE when no oversampling takes place. Both events may occur before the actual value has been transferred into RAM by EasyDMA.

7.29.5.3 Oversampling

An accumulator in the ADC can be used to average noise on the analog input. In general, oversampling improves the signal-to-noise ratio (SNR). Oversampling, however, does not improve the integral non-linearity (INL), or differential non-linearity (DNL).

Oversampling and Scan mode should not be combined, since oversampling and scan will average over input channels.

The accumulator is controlled in the OVERSAMPLE register. The SAMPLE task must be set $2^{OVERSAMPLE}$ number of times before the result is written to RAM. This can be achieved by the following:

- Configuring a fixed sampling rate using the local timer or a general purpose timer and the PPI system to trigger a SAMPLE task
- Triggering SAMPLE $2^{OVERSAMPLE}$ times from software
- Enabling BURST mode

CH[n].CONFIG.BURST can be enabled to avoid setting SAMPLE task $2^{OVERSAMPLE}$ times. With BURST = 1 the ADC will sample the input $2^{OVERSAMPLE}$ times as fast as it can (actual timing: $<(t_{ACQ}+t_{CONV})\times 2^{OVERSAMPLE}$). Thus, for the user it will just appear like the conversion took a bit longer time, but other than that, it is similar to One-shot mode.

A DONE event signals that one sample has been taken.

In this mode, the RESULTDONE event signals that enough conversions have taken place for an oversampled result to get transferred into RAM. Both events may occur before the actual value has been transferred into RAM by EasyDMA.

7.29.5.4 Scan mode

A channel is considered enabled if CH[n].PSELP is set. If more than one channel, CH[n], is enabled, the ADC enters Scan mode.

In Scan mode, one SAMPLE task will trigger one conversion per enabled channel. The time it takes to sample all channels is given by the following equation:

```
Total time < Sum(CH[x].tACQ+tCONV), x=0..enabled channels
```

A DONE event signals that one sample has been taken.

In this mode, the RESULTDONE event signals has the same meaning as DONE when no oversampling takes place. Both events may occur before the actual values have been transferred into RAM by EasyDMA.

The following figure shows an example of the placement of results in Data RAM, with an even RESULT.MAXCNT. In this example, channels 1, 2, and 5 are enabled, all others are disabled.

	31 … 16	15 … 0
RESULT.PTR	CH[2] 1st result	CH[1] 1st result
RESULT.PTR + 4	CH[1] 2nd result	CH[5] 1st result
RESULT.PTR + 8	CH[5] 2nd result	CH[2] 2nd result
	(…)	
RESULT.PTR + 2*(RESULT.MAXCNT – 2)	CH[5] last result	CH[2] last result

Figure 190: Example of RAM placement (even RESULT.MAXCNT), channels 1, 2 and, 5 enabled

The following figure shows an example of the placement of results in Data RAM, with an odd RESULT.MAXCNT. In this example, channels 1, 2 and, 5 are enabled, all others are disabled. The last 32-bit word is populated only with one 16-bit result.

	31 … 16	15 … 0
RESULT.PTR	CH[2] 1st result	CH[1] 1st result
RESULT.PTR + 4	CH[1] 2nd result	CH[5] 1st result
RESULT.PTR + 8	CH[5] 2nd result	CH[2] 2nd result
	(…)	
RESULT.PTR + 2*(RESULT.MAXCNT – 1)		CH[5] last result

Figure 191: Example of RAM placement (odd RESULT.MAXCNT), channels 1, 2 and, 5 enabled

7.29.6 EasyDMA

After configuring RESULT.PTR and RESULT.MAXCNT, the ADC resources are started by triggering the START task. The ADC is using EasyDMA to store results in a Result buffer in RAM.

The Result buffer is located at the address specified in the RESULT.PTR register. The RESULT.PTR register is double-buffered and it can be updated and prepared for the next START task immediately after the STARTED event is generated. The size of the Result buffer is specified in the RESULT.MAXCNT register and the ADC will generate an END event when it has filled up the Result buffer, see ADC on page 525. Results are stored in little-endian byte order in Data RAM. Every sample will be sign extended to 16 bit before stored in the Result buffer.

The ADC is stopped by triggering the STOP task. The STOP task will terminate an ongoing sampling. The ADC will generate a STOPPED event when it has stopped. If the ADC is already stopped when the STOP task is triggered, the STOPPED event will still be generated.

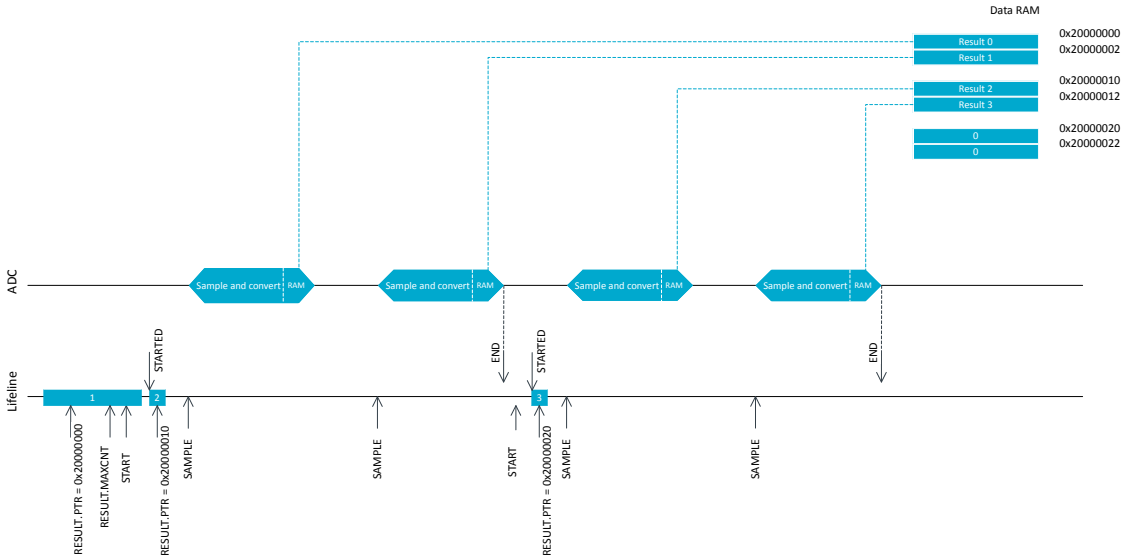


Figure 192: ADC

If the RESULT.PTR is not pointing to a RAM region accessible from the peripheral, an EasyDMA transfer may result in a HardFault and/or memory corruption. See Memory on page 18 for more information about the different memory regions.

The EasyDMA will have finished accessing the RAM when the END or STOPPED event has been generated.

The RESULT.AMOUNT register can be read following an END event or a STOPPED event to see how many results have been transferred to the Result buffer in RAM since the START task was triggered.

In Scan mode, SAMPLE tasks can be triggered once the START task is triggered. The END event is generated when the number of samples transferred to memory reaches the value specified by RESULT.MAXCNT. After an END event, the START task needs to be triggered again before new samples can be taken. By specifying RESULT.MAXCNT ≥ number of channels enabled, the size of the Result buffer should be large enough for a minimum of one result from each of the enabled channels. See Scan mode on page 524 for more information.

7.29.7 Resistor ladder

The ADC has an internal resistor string for positive and negative input.

The resistors in the following figure are controlled in the CH[n].CONFIG.RESP and CH[n].CONFIG.RESN registers.

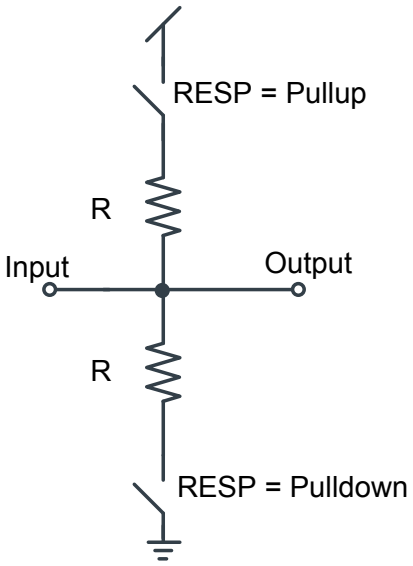


Figure 193: Resistor ladder for positive input

7.29.8 Reference

The ADC can use the following two references, controlled in the REFSEL field of the CH[n].CONFIG register.

- Internal reference
- VDD as reference

The internal reference results in an input range of ±0.6 V on the ADC core. VDD as reference results in an input range of ±VDD/4 on the ADC core. The gain block can be used to change the effective input range of the ADC.

```
Input range = (+- 0.6 V or +-VDD/4)/Gain
```

For example, choosing VDD as reference, single ended input (grounded negative input), and a gain of 1/4 the input range is the following:

```
Input range = (VDD/4)/(1/4) = VDD
```

With internal reference, single ended input (grounded negative input), and a gain of 1/6 the input range is the following:

```
Input range = (0.6 V)/(1/6) = 3.6 V
```

The **AIN0** - **AIN7** inputs cannot exceed VDD, or be lower than VSS.

7.29.9 Acquisition time

To sample the input voltage, the ADC connects a capacitor to the input.

The acquisition time indicates how long the capacitor is connected, see TACQ field in register CH[n].CONFIG (n=0..7) on page 542. The required acquisition time depends on the source (R_{source}) resistance. For high source resistance, the acquisition time should be increased. See the following table for more information.

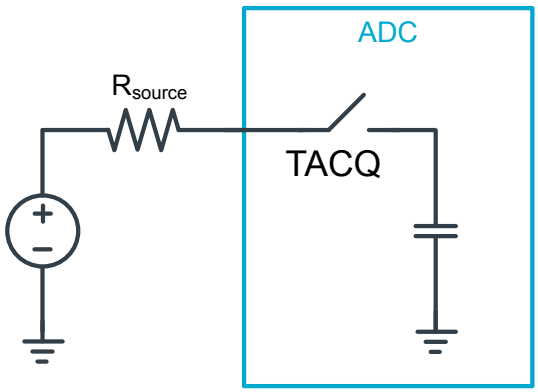


Figure 194: Simplified ADC sample network

TACQ [µs]	Maximum source resistance [kOhm]
3	10
5	40
10	100
15	200
20	400
40	800

Table 140: Acquisition time

When using **VDDHDIV5** as input, the acquisition time needs to be 10 µs or higher.

7.29.10 Limits event monitoring

A channel can be event monitored by configuring limit register CH[n].LIMIT.

If the conversion result is higher than the defined high limit, or lower than the defined low limit, the appropriate event will get fired.

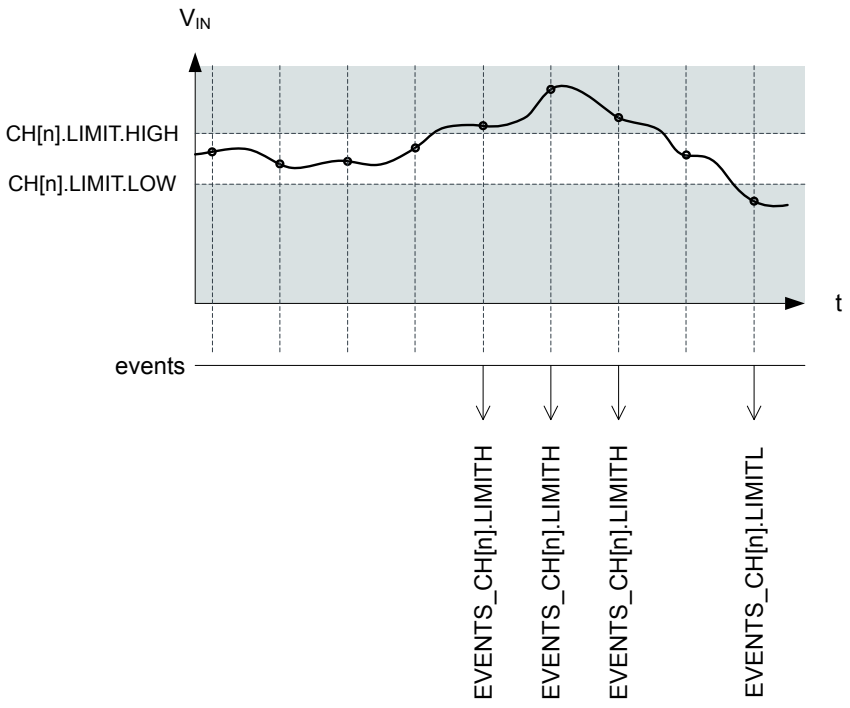


Figure 195: Example of limits monitoring on channel 'n'

The CH[n].LIMIT.HIGH shall always be higher than or equal to CH[n].LIMIT.LOW . In other words, an event can be generated only when the input signal has been sampled outside of the defined limits. It is not possible to generate an event when the input signal is inside a defined range by swapping high and low limits.

The comparison to limits always takes place, there is no need to enable it. If comparison is not required on a channel, the software shall simply ignore the related events. In that situation, the value of the limits registers is irrelevant, so it does not matter if CH[n].LIMIT.LOW is lower than CH[n].LIMIT.HIGH or not.

7.29.11 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x5000E000 0x4000E000	APPLICATION	SAADC	SAADC : S SAADC : NS	US	SA	Successive approximation analog-to-digital converter	

Table 141: Instances

Register	Offset	Security	Description
TASKS_START	0x000		Start the ADC and prepare the result buffer in RAM
TASKS_SAMPLE	0x004		Take one ADC sample, if scan is enabled all channels are sampled
TASKS_STOP	0x008		Stop the ADC and terminate any ongoing conversion
TASKS_CALIBRATEOFFSET	0x00C		Starts offset auto-calibration
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_SAMPLE	0x084		Subscribe configuration for task SAMPLE
SUBSCRIBE_STOP	0x088		Subscribe configuration for task STOP
SUBSCRIBE_CALIBRATEOFFSET	0x08C		Subscribe configuration for task CALIBRATEOFFSET
EVENTS_STARTED	0x100		The ADC has started
EVENTS_END	0x104		The ADC has filled up the Result buffer
EVENTS_DONE	0x108		A conversion task has been completed. Depending on the mode, multiple conversions might be needed for a result to be transferred to RAM.
EVENTS_RESULTDONE	0x10C		A result is ready to get transferred to RAM
EVENTS_CALIBRATEDONE	0x110		Calibration is complete
EVENTS_STOPPED	0x114		The ADC has stopped
EVENTS_CH[n].LIMITH	0x118		Last results is equal or above CH[n].LIMIT.HIGH
EVENTS_CH[n].LIMITL	0x11C		Last results is equal or below CH[n].LIMIT.LOW
PUBLISH_STARTED	0x180		Publish configuration for event STARTED
PUBLISH_END	0x184		Publish configuration for event END
PUBLISH_DONE	0x188		Publish configuration for event DONE
PUBLISH_RESULTDONE	0x18C		Publish configuration for event RESULTDONE
PUBLISH_CALIBRATEDONE	0x190		Publish configuration for event CALIBRATEDONE
PUBLISH_STOPPED	0x194		Publish configuration for event STOPPED
PUBLISH_CH[n].LIMITH	0x198		Publish configuration for event CH[n].LIMITH
PUBLISH_CH[n].LIMITL	0x19C		Publish configuration for event CH[n].LIMITL
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
STATUS	0x400		Status
ENABLE	0x500		Enable or disable ADC
CH[n].PSELP	0x510		Input positive pin selection for CH[n]
CH[n].PSELN	0x514		Input negative pin selection for CH[n]
CH[n].CONFIG	0x518		Input configuration for CH[n]
CH[n].LIMIT	0x51C		High/low limits for event monitoring a channel
RESOLUTION	0x5F0		Resolution configuration

Register	Offset	Security	Description
OVERSAMPLE	0x5F4		Oversampling configuration. OVERSAMPLE should not be combined with SCAN. The RESOLUTION is applied before averaging, thus for high OVERSAMPLE a higher RESOLUTION should be used.
SAMPLERATE	0x5F8		Controls normal or continuous sample rate
RESULT.PTR	0x62C		Data pointer
RESULT.MAXCNT	0x630		Maximum number of buffer words to transfer
RESULT.AMOUNT	0x634		Number of buffer words transferred since last START

Table 142: Register overview

7.29.11.1 TASKS_START

Address offset: 0x000

Start the ADC and prepare the result buffer in RAM

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_START			Start the ADC and prepare the result buffer in RAM
			Trigger	1	Trigger task

7.29.11.2 TASKS_SAMPLE

Address offset: 0x004

Take one ADC sample, if scan is enabled all channels are sampled

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_SAMPLE			Take one ADC sample, if scan is enabled all channels are sampled
			Trigger	1	Trigger task

7.29.11.3 TASKS_STOP

Address offset: 0x008

Stop the ADC and terminate any ongoing conversion

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Stop the ADC and terminate any ongoing conversion
			Trigger	1	Trigger task

7.29.11.4 TASKS_CALIBRATEOFFSET

Address offset: 0x00C

Starts offset auto-calibration

Do not trigger when the ADC has been started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_CALIBRATEOFFSET			Starts offset auto-calibration
					Do not trigger when the ADC has been started
			Trigger	1	Trigger task

7.29.11.5 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task START

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task START will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.29.11.6 SUBSCRIBE_SAMPLE

Address offset: 0x084

Subscribe configuration for task SAMPLE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task SAMPLE will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.29.11.7 SUBSCRIBE_STOP

Address offset: 0x088

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.29.11.8 SUBSCRIBE_CALIBRATEOFFSET

Address offset: 0x08C

Subscribe configuration for task CALIBRATEOFFSET

Do not trigger when the ADC has been started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task CALIBRATEOFFSET will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.29.11.9 EVENTS_STARTED

Address offset: 0x100

The ADC has started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_STARTED			The ADC has started
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.29.11.10 EVENTS_END

Address offset: 0x104

The ADC has filled up the Result buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_END			The ADC has filled up the Result buffer
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.29.11.11 EVENTS_DONE

Address offset: 0x108

A conversion task has been completed. Depending on the mode, multiple conversions might be needed for a result to be transferred to RAM.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_DONE			A conversion task has been completed. Depending on the mode, multiple conversions might be needed for a result to be transferred to RAM.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.29.11.12 EVENTS_RESULTDONE

Address offset: 0x10C

A result is ready to get transferred to RAM

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RESULTDONE			A result is ready to get transferred to RAM
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.29.11.13 EVENTS_CALIBRATEDONE

Address offset: 0x110

Calibration is complete

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_CALIBRATEDONE			Calibration is complete
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.29.11.14 EVENTS_STOPPED

Address offset: 0x114

The ADC has stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_STOPPED			The ADC has stopped
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.29.11.15 EVENTS_CH[n].LIMITH (n=0..7)

Address offset: 0x118 + (n × 0x8)

Last results is equal or above CH[n].LIMIT.HIGH

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LIMITH			Last results is equal or above CH[n].LIMIT.HIGH
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.29.11.16 EVENTS_CH[n].LIMITL (n=0..7)

Address offset: 0x11C + (n × 0x8)

Last results is equal or below CH[n].LIMIT.LOW

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LIMITL			Last results is equal or below CH[n].LIMIT.LOW
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.29.11.17 PUBLISH_STARTED

Address offset: 0x180

Publish configuration for event STARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event STARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.29.11.18 PUBLISH_END

Address offset: 0x184

Publish configuration for event END

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event END will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.29.11.19 PUBLISH_DONE

Address offset: 0x188

Publish configuration for event DONE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event DONE will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.29.11.20 PUBLISH_RESULTDONE

Address offset: 0x18C

Publish configuration for event RESULTDONE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RESULTDONE will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.29.11.21 PUBLISH_CALIBRATEDONE

Address offset: 0x190

Publish configuration for event CALIBRATEDONE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CALIBRATEDONE will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.29.11.22 PUBLISH_STOPPED

Address offset: 0x194

Publish configuration for event STOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event STOPPED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.29.11.23 PUBLISH_CH[n].LIMITH (n=0..7)

Address offset: 0x198 + (n × 0x8)

Publish configuration for event CH[n].LIMITH

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CH[n].LIMITH will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.29.11.24 PUBLISH_CH[n].LIMITL (n=0..7)

Address offset: 0x19C + (n × 0x8)

Publish configuration for event CH[n].LIMITL

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CH[n].LIMITL will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.29.11.25 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STARTED			Enable or disable interrupt for event STARTED
			Disabled	0	Disable
			Enabled	1	Enable
B	RW	END			Enable or disable interrupt for event END
			Disabled	0	Disable
			Enabled	1	Enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
C	RW	DONE			Enable or disable interrupt for event DONE
			Disabled	0	Disable
			Enabled	1	Enable
D	RW	RESULTDONE			Enable or disable interrupt for event RESULTDONE
			Disabled	0	Disable
			Enabled	1	Enable
E	RW	CALIBRATEDONE			Enable or disable interrupt for event CALIBRATEDONE
			Disabled	0	Disable
			Enabled	1	Enable
F	RW	STOPPED			Enable or disable interrupt for event STOPPED
			Disabled	0	Disable
			Enabled	1	Enable
G	RW	CH0LIMITH			Enable or disable interrupt for event CH0LIMITH
			Disabled	0	Disable
			Enabled	1	Enable
H	RW	CH0LIMITL			Enable or disable interrupt for event CH0LIMITL
			Disabled	0	Disable
			Enabled	1	Enable
I	RW	CH1LIMITH			Enable or disable interrupt for event CH1LIMITH
			Disabled	0	Disable
			Enabled	1	Enable
J	RW	CH1LIMITL			Enable or disable interrupt for event CH1LIMITL
			Disabled	0	Disable
			Enabled	1	Enable
K	RW	CH2LIMITH			Enable or disable interrupt for event CH2LIMITH
			Disabled	0	Disable
			Enabled	1	Enable
L	RW	CH2LIMITL			Enable or disable interrupt for event CH2LIMITL
			Disabled	0	Disable
			Enabled	1	Enable
M	RW	CH3LIMITH			Enable or disable interrupt for event CH3LIMITH
			Disabled	0	Disable
			Enabled	1	Enable
N	RW	CH3LIMITL			Enable or disable interrupt for event CH3LIMITL
			Disabled	0	Disable
			Enabled	1	Enable
O	RW	CH4LIMITH			Enable or disable interrupt for event CH4LIMITH
			Disabled	0	Disable
			Enabled	1	Enable
P	RW	CH4LIMITL			Enable or disable interrupt for event CH4LIMITL
			Disabled	0	Disable
			Enabled	1	Enable
Q	RW	CH5LIMITH			Enable or disable interrupt for event CH5LIMITH
			Disabled	0	Disable
			Enabled	1	Enable
R	RW	CH5LIMITL			Enable or disable interrupt for event CH5LIMITL
			Disabled	0	Disable
			Enabled	1	Enable
S	RW	CH6LIMITH			Enable or disable interrupt for event CH6LIMITH
			Disabled	0	Disable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Enabled	1	Enable
T	RW	CH6LIMITL			Enable or disable interrupt for event CH6LIMITL
			Disabled	0	Disable
			Enabled	1	Enable
U	RW	CH7LIMITH			Enable or disable interrupt for event CH7LIMITH
			Disabled	0	Disable
			Enabled	1	Enable
V	RW	CH7LIMITL			Enable or disable interrupt for event CH7LIMITL
			Disabled	0	Disable
			Enabled	1	Enable

7.29.11.26 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	STARTED			Write '1' to enable interrupt for event STARTED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	END			Write '1' to enable interrupt for event END
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	DONE			Write '1' to enable interrupt for event DONE
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	RESULTDONE			Write '1' to enable interrupt for event RESULTDONE
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	CALIBRATEDONE			Write '1' to enable interrupt for event CALIBRATEDONE
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	STOPPED			Write '1' to enable interrupt for event STOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	CH0LIMITH			Write '1' to enable interrupt for event CH0LIMITH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	CH0LIMITL			Write '1' to enable interrupt for event CH0LIMITL

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
I	RW	CH1LIMITH			Write '1' to enable interrupt for event CH1LIMITH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
J	RW	CH1LIMITL			Write '1' to enable interrupt for event CH1LIMITL
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
K	RW	CH2LIMITH			Write '1' to enable interrupt for event CH2LIMITH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
L	RW	CH2LIMITL			Write '1' to enable interrupt for event CH2LIMITL
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
M	RW	CH3LIMITH			Write '1' to enable interrupt for event CH3LIMITH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
N	RW	CH3LIMITL			Write '1' to enable interrupt for event CH3LIMITL
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
O	RW	CH4LIMITH			Write '1' to enable interrupt for event CH4LIMITH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
P	RW	CH4LIMITL			Write '1' to enable interrupt for event CH4LIMITL
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Q	RW	CH5LIMITH			Write '1' to enable interrupt for event CH5LIMITH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
R	RW	CH5LIMITL			Write '1' to enable interrupt for event CH5LIMITL
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
S	RW	CH6LIMITH			Write '1' to enable interrupt for event CH6LIMITH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
T	RW	CH6LIMITL			Write '1' to enable interrupt for event CH6LIMITL
			Set	1	Enable
			Disabled	0	Read: Disabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
			Enabled	1	Read: Enabled
U	RW	CH7LIMITH			Write '1' to enable interrupt for event CH7LIMITH
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
V	RW	CH7LIMITL			Write '1' to enable interrupt for event CH7LIMITL
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.29.11.27 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STARTED			Write '1' to disable interrupt for event STARTED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	END			Write '1' to disable interrupt for event END
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	DONE			Write '1' to disable interrupt for event DONE
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	RESULTDONE			Write '1' to disable interrupt for event RESULTDONE
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	CALIBRATEDONE			Write '1' to disable interrupt for event CALIBRATEDONE
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	STOPPED			Write '1' to disable interrupt for event STOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	CH0LIMITH			Write '1' to disable interrupt for event CH0LIMITH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	CH0LIMITL			Write '1' to disable interrupt for event CH0LIMITL
			Clear	1	Disable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
I	RW	CH1LIMITH			Write '1' to disable interrupt for event CH1LIMITH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
J	RW	CH1LIMITL			Write '1' to disable interrupt for event CH1LIMITL
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
K	RW	CH2LIMITH			Write '1' to disable interrupt for event CH2LIMITH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
L	RW	CH2LIMITL			Write '1' to disable interrupt for event CH2LIMITL
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
M	RW	CH3LIMITH			Write '1' to disable interrupt for event CH3LIMITH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
N	RW	CH3LIMITL			Write '1' to disable interrupt for event CH3LIMITL
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
O	RW	CH4LIMITH			Write '1' to disable interrupt for event CH4LIMITH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
P	RW	CH4LIMITL			Write '1' to disable interrupt for event CH4LIMITL
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Q	RW	CH5LIMITH			Write '1' to disable interrupt for event CH5LIMITH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
R	RW	CH5LIMITL			Write '1' to disable interrupt for event CH5LIMITL
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
S	RW	CH6LIMITH			Write '1' to disable interrupt for event CH6LIMITH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
T	RW	CH6LIMITL			Write '1' to disable interrupt for event CH6LIMITL
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
U	RW	CH7LIMITH			Write '1' to disable interrupt for event CH7LIMITH
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
V	RW	CH7LIMITL			Write '1' to disable interrupt for event CH7LIMITL
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.29.11.28 STATUS

Address offset: 0x400

Status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	STATUS			Status
			Ready	0	ADC is ready. No ongoing conversion.
			Busy	1	ADC is busy. Single conversion in progress.

7.29.11.29 ENABLE

Address offset: 0x500

Enable or disable ADC

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	ENABLE			Enable or disable ADC
			Disabled	0	Disable ADC
			Enabled	1	Enable ADC
					When enabled, the ADC will acquire access to the analog input pins specified in the CH[n].PSELP and CH[n].PSELN registers.

7.29.11.30 CH[n].PSELP (n=0..7)

Address offset: 0x510 + (n × 0x10)

Input positive pin selection for CH[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	PSELP			Analog positive input channel
			NC	0	Not connected
			AnalogInput0	1	AIN0
			AnalogInput1	2	AIN1
			AnalogInput2	3	AIN2
			AnalogInput3	4	AIN3
			AnalogInput4	5	AIN4
			AnalogInput5	6	AIN5
			AnalogInput6	7	AIN6
			AnalogInput7	8	AIN7
			VDD	9	VDD
			VDDHDIV5	0xD	VDDH/5

7.29.11.31 CH[n].PSELN (n=0..7)

Address offset: 0x514 + (n × 0x10)

Input negative pin selection for CH[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	PSELN			Analog negative input, enables differential channel
			NC	0	Not connected
			AnalogInput0	1	AIN0
			AnalogInput1	2	AIN1
			AnalogInput2	3	AIN2
			AnalogInput3	4	AIN3
			AnalogInput4	5	AIN4
			AnalogInput5	6	AIN5
			AnalogInput6	7	AIN6
			AnalogInput7	8	AIN7
			VDD	9	VDD
			VDDHDIV5	0xD	VDDH/5

7.29.11.32 CH[n].CONFIG (n=0..7)

Address offset: 0x518 + (n × 0x10)

Input configuration for CH[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				G F E E E D C C C B B A A
Reset 0x00020000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	RESP			Positive channel resistor control
			Bypass	0	Bypass resistor ladder
			Pulldown	1	Pull-down to GND
			Pullup	2	Pull-up to VDD
			VDD1_2	3	Set input at VDD/2

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				G F E E E D C C C B B A A
Reset 0x00020000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
B	RW	RESN			Negative channel resistor control
			Bypass	0	Bypass resistor ladder
			Pulldown	1	Pull-down to GND
			Pullup	2	Pull-up to VDD
			VDD1_2	3	Set input at VDD/2
C	RW	GAIN			Gain control
			Gain1_6	0	1/6
			Gain1_5	1	1/5
			Gain1_4	2	1/4
			Gain1_3	3	1/3
			Gain1_2	4	1/2
			Gain1	5	1
			Gain2	6	2
			Gain4	7	4
D	RW	REFSEL			Reference control
			Internal	0	Internal reference (0.6 V)
			VDD1_4	1	VDD/4 as reference
E	RW	TACQ			Acquisition time, the time the ADC uses to sample the input voltage
			3us	0	3 µs
			5us	1	5 µs
			10us	2	10 µs
			15us	3	15 µs
			20us	4	20 µs
			40us	5	40 µs
F	RW	MODE			Enable differential mode
			SE	0	Single-ended, PSELN will be ignored, negative input to ADC shorted to GND
			Diff	1	Differential
G	RW	BURST			Enable burst mode
			Disabled	0	Burst mode is disabled (normal operation)
			Enabled	1	Burst mode is enabled. SAADC takes 2^OVERSAMPLE number of samples as fast as it can, and sends the average to Data RAM.

7.29.11.33 CH[n].LIMIT (n=0..7)

Address offset: 0x51C + (n × 0x10)

High/low limits for event monitoring a channel

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B B B B B B B B B B B B B B B A A A A A A A A A A A A A A
Reset 0x7FFF8000				**0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LOW		[-32768 to +32767]	Low level limit
B	RW	HIGH		[-32768 to +32767]	High level limit

7.29.11.34 RESOLUTION

Address offset: 0x5F0

Resolution configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**

ID	R/W	Field	Value ID	Value	Description
A	RW	VAL			Set the resolution
			8bit	0	8 bit
			10bit	1	10 bit
			12bit	2	12 bit
			14bit	3	14 bit

7.29.11.35 OVERSAMPLE

Address offset: 0x5F4

Oversampling configuration. OVERSAMPLE should not be combined with SCAN. The RESOLUTION is applied before averaging, thus for high OVERSAMPLE a higher RESOLUTION should be used.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	OVERSAMPLE			Oversample control
			Bypass	0	Bypass oversampling
			Over2x	1	Oversample 2x
			Over4x	2	Oversample 4x
			Over8x	3	Oversample 8x
			Over16x	4	Oversample 16x
			Over32x	5	Oversample 32x
			Over64x	6	Oversample 64x
			Over128x	7	Oversample 128x
			Over256x	8	Oversample 256x

7.29.11.36 SAMPLERATE

Address offset: 0x5F8

Controls normal or continuous sample rate

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CC		[80..2047]	Capture and compare value; sample rate is 16 MHz/CC
B	RW	MODE			Select mode for sample rate control
			Task	0	Rate is controlled from SAMPLE task
			Timers	1	Rate is controlled from local timer (use CC to control the rate)

7.29.11.37 RESULT.PTR

Address offset: 0x62C

Data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.29.11.38 RESULT.MAXCNT

Address offset: 0x630

Maximum number of buffer words to transfer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT			Maximum number of buffer words to transfer

7.29.11.39 RESULT.AMOUNT

Address offset: 0x634

Number of buffer words transferred since last START

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT			Number of buffer words transferred since last START. This register can be read after an END or STOPPED event.

7.29.12 Electrical specification

7.29.12.1 SAADC Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
DNL_{10}	Differential non-linearity, 10-bit resolution	-0.9	<1		LSB10b
INL_{10}	Integral non-linearity, 10-bit resolution		±1		LSB10b
V_{OS}	Differential offset error (calibrated), 10-bit resolution [17]		±2		LSB10b
DNL_{12}	Differential non-linearity, 12-bit resolution	-0.9	1.3		LSB12b
INL_{12}	Integral non-linearity, 12-bit resolution		3.7		LSB12b
$E_{VDDHDIV5}$	Error on VDDHDIV5 input		±1		%
C_{EG}	Gain error temperature coefficient		0.02		%/°C
f_{SAMPLE}	Maximum sampling rate			200	kHz
$t_{ACQ,10k}$	Acquisition time (configurable), source Resistance ≤ 10 kΩ		3		µs
$t_{ACQ,40k}$	Acquisition time (configurable), source Resistance ≤ 40 kΩ		5		µs
$t_{ACQ,100k}$	Acquisition time (configurable), source Resistance ≤ 100 kΩ		10		µs
$t_{ACQ,200k}$	Acquisition time (configurable), source Resistance ≤ 200 kΩ		15		µs
$t_{ACQ,400k}$	Acquisition time (configurable), source Resistance ≤ 400 kΩ		20		µs

[17] Digital output code at zero volt differential input.

Symbol	Description	Min.	Typ.	Max.	Units
$t_{ACQ,800k}$	Acquisition time (configurable), source Resistance ≤ 800 kΩ		40		µs
t_{CONV}	Conversion time		2		µs
$E_{G1/6}$	Error[18] for Gain = 1/6	-3		3	%
$E_{G1/4}$	Error[18] for Gain = 1/4	-3		3	%
$E_{G1/2}$	Error[18] for Gain = 1/2	-3		4	%
E_{G1}	Error[18] for Gain = 1	-3		4	%
C_{SAMPLE}	Sample and hold capacitance at maximum gain[19]		2.5		pF
R_{INPUT}	Input resistance		>1		MΩ
E_{NOB}	Effective number of bits, differential mode, 12-bit resolution, 1/1 gain, 3 µs acquisition time, HFXO, 32 ksps, F_{in} = 3 kHz		9.8		Bit
S_{NDR}	Peak signal to noise and distortion ratio, differential mode, 12-bit resolution, 1/1 gain, 3 µs acquisition time, HFXO, 32 ksps, F_{in} = 3 kHz		61		dB
S_{FDR}	Spurious free dynamic range, differential mode, 12-bit resolution, 1/1 gain, 3 µs acquisition time, HFXO, 32 ksps, F_{in} = 3 kHz		73		dBc
R_{LADDER}	Ladder resistance		160		kΩ

7.29.13 Performance factors

Clock jitter, affecting sample timing accuracy, and circuit noise can affect ADC performance.

Jitter can be between START tasks or from START task to acquisition. START timer accuracy and startup times of regulators and references will contribute to variability. Sources of circuit noise may include CPU activity and the DC/DC regulator. The best ADC performance is achieved using START timing based on the TIMER module, HFXO clock source, and Constant Latency mode.

7.30 SPIM — Serial peripheral interface master with EasyDMA

The SPI master can communicate with multiple SPI slaves using individual chip select signals for each slave.

The the main features of SPIM are:

- EasyDMA direct transfer to/from RAM
- SPI mode 0-3
- Individual selection of I/O pins
- Optional D/CX output line for distinguishing between command and data bytes

[18] Temperature drift is not included.

[19] Maximum gain corresponds to highest capacitance.

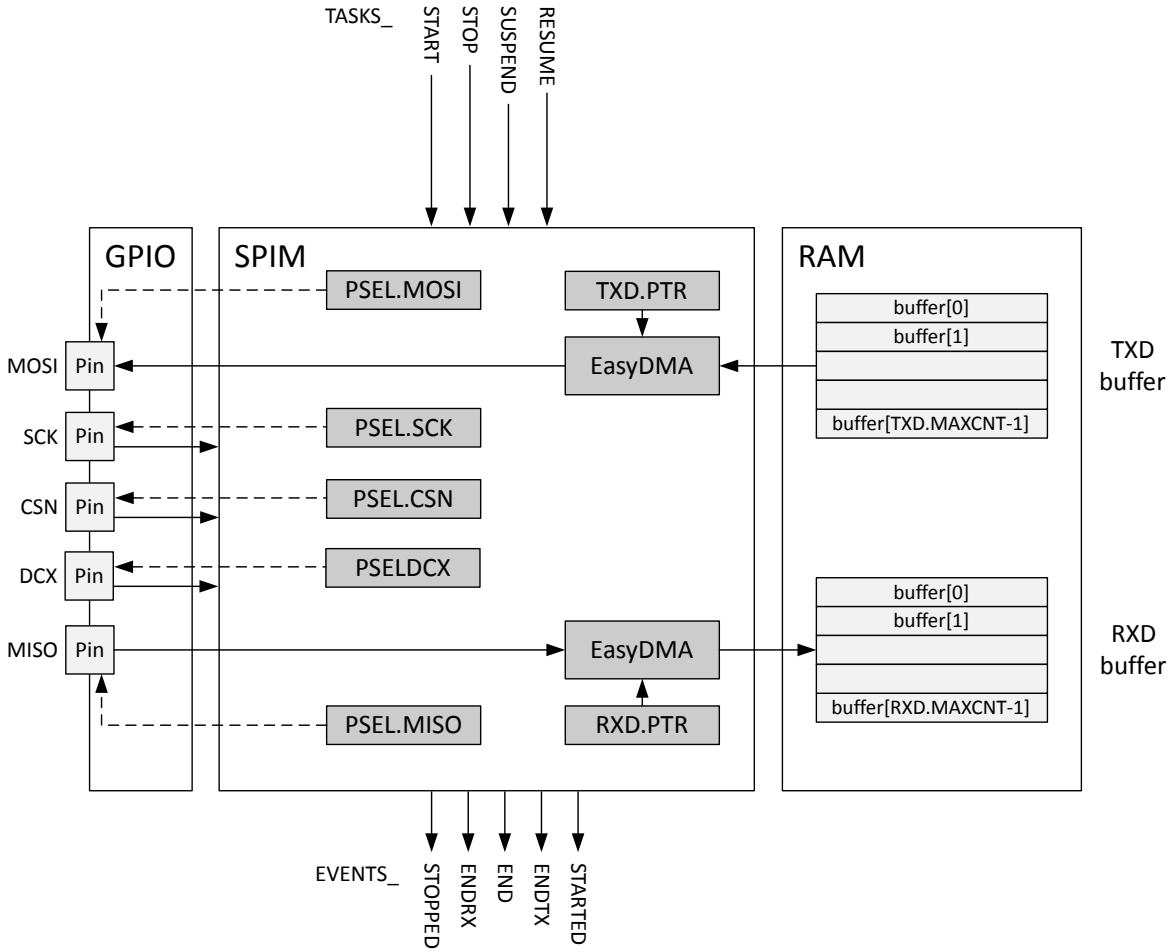


Figure 196: SPIM — SPI master with EasyDMA

7.30.1 SPI master transaction sequence

An SPI master transaction is started by triggering the START task. When started, a number of bytes will be transmitted/received on MOSI/MISO.

The following figure illustrates an SPI master transaction.

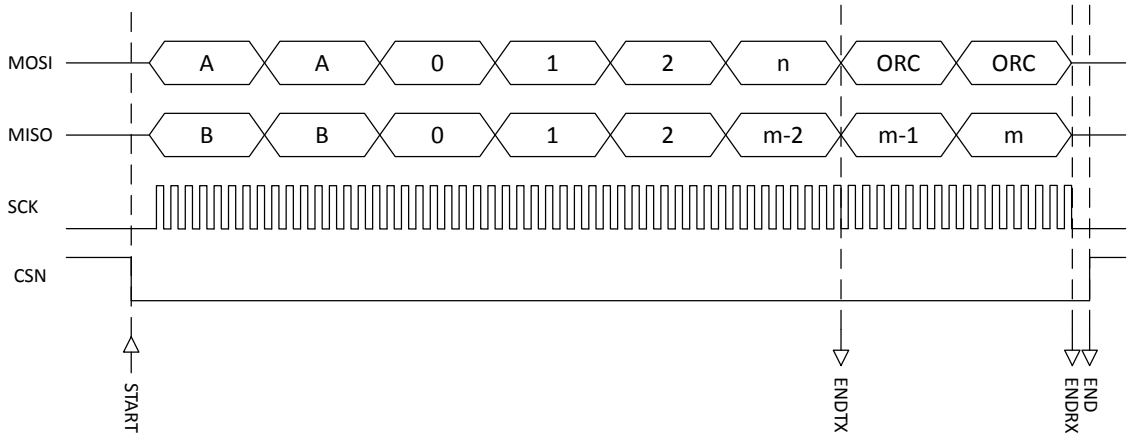


Figure 197: SPI master transaction

The ENDTX is generated when all bytes in buffer TXD.PTR on page 562 are transmitted. The number of bytes in the transmit buffer is specified in register TXD.MAXCNT on page 562. The ENDRX event will be generated when buffer RXD.PTR on page 561 is full, that is when the number of bytes specified in register RXD.MAXCNT on page 561 have been received. The transaction stops automatically after all bytes have been transmitted/received. When the maximum number of bytes in receive buffer is larger than the number of bytes in the transmit buffer, the contents of register ORC on page 565 will be transmitted after the last byte in the transmit buffer has been transmitted.

The END event will be generated after both the ENDRX and ENDTX events have been generated.

The SPI master can be stopped by triggering the STOP task. A STOPPED event is generated when the SPI master has stopped. If the STOP task is triggered in the middle of a transaction, SPIM will complete the transmission/reception of the current byte before stopping. The STOPPED event is generated even if the STOP task is triggered while there is no ongoing transaction.

If the ENDTX event has not already been generated when the SPI master has come to a stop, the ENDTX event will be generated even if all bytes in the buffer TXD.PTR on page 562 have not been transmitted.

If the ENDRX event has not already been generated when the SPI master has come to a stop, the ENDRX event will be generated even if the buffer RXD.PTR on page 561 is not full.

A transaction can be suspended and resumed using the SUSPEND and RESUME tasks, receptively. When the SUSPEND task is triggered, the SPI master will complete transmitting and receiving the current ongoing byte before it is suspended.

7.30.2 D/CX functionality

Some SPI slaves, for example display drivers, require an additional signal from the SPI master to distinguish between command and data bytes. For display drivers this line is often called D/CX.

The SPIM provides support for such a D/CX output line. The D/CX line is set low during transmission of command bytes and high during transmission of data bytes.

The D/CX pin number is selected using PSELDCX on page 564 and the number of command bytes preceding the data bytes is configured using DCXCNT on page 565.

It is not allowed to write to the DCXCNT on page 565 during an ongoing transmission.

The following figure shows the use of D/CX, using SPIM.DCXCNT=1.

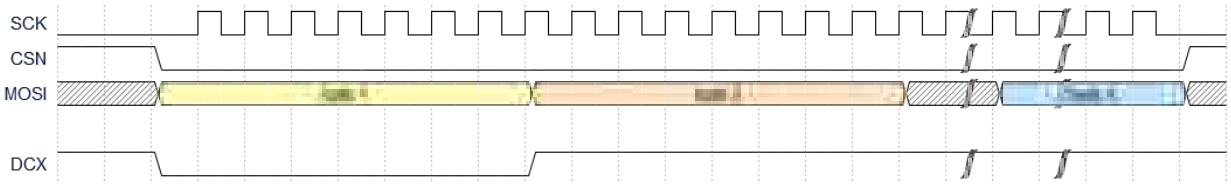


Figure 198: D/CX example. SPIM.DCXCNT = 1.

7.30.3 Pin configuration

The SCK, CSN, DCX, MOSI, and MISO signals associated with the SPIM are mapped to physical pins according to the configuration specified in the PSEL.n registers.

The contents of registers PSEL.SCK on page 559, PSEL.CSN on page 560, PSELDCX on page 564, PSEL.MOSI on page 560, and PSEL.MISO on page 560 are only used when the SPIM is enabled and retained only as long as the device is in System ON mode. The PSEL.n registers can only be configured when the SPIM is disabled. Enabling/disabling is done using register ENABLE on page 559.

To ensure correct behavior, the pins used by the SPIM must be configured in the GPIO peripheral as described in GPIO configuration on page 549 before the SPIM is enabled. Make sure to activate the dedicated peripheral setting of the GPIO pin. See GPIO — General purpose input/output on page 220 for details on how to assign pins between cores, peripherals, or subsystems. For pin recommendations, see Pin assignments on page 783.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

SPI master signal	SPI master pin	Direction	Output value
SCK	As specified in PSEL.SCK on page 559	Output	Same as CONFIG.CPOL
CSN	As specified in PSEL.CSN on page 560	Output	Same as CONFIG.CPOL
DCX	As specified in PSELDCX on page 564	Output	1
MOSI	As specified in PSEL.MOSI on page 560	Output	0
MISO	As specified in PSEL.MISO on page 560	Input	Not applicable

Table 143: GPIO configuration

Some SPIM instances do not support automatic control of CSN, and for those the available GPIO pins need to be used to control CSN directly. See Instances on page 551 for information about what features are supported in the various SPIM instances.

The SPIM supports SPI modes 0 through 3. The clock polarity (CPOL) and the clock phase (CPHA) are configured in register CONFIG on page 563.

Mode	Clock polarity	Clock phase
	CPOL	CPHA
SPI_MODE0	0 (Active High)	0 (Leading)
SPI_MODE1	0 (Active High)	1 (Trailing)
SPI_MODE2	1 (Active Low)	0 (Leading)
SPI_MODE3	1 (Active Low)	1 (Trailing)

Table 144: SPI modes

7.30.4 Shared resources

The SPI shares registers and other resources with other peripherals that have the same ID as the SPI. Therefore, the user must disable all peripherals that have the same ID as the SPI before the SPI can be configured and used.

Disabling a peripheral that has the same ID as the SPI will not reset any of the registers that are shared with the SPI. It is therefore important to configure all relevant SPI registers explicitly to secure that it operates correctly.

See the Instantiation table in Peripherals on page 146 for details on peripherals and their IDs.

7.30.5 EasyDMA

SPIM implements EasyDMA for accessing RAM without CPU involvement.

SPIM implements the following EasyDMA channels.

Channel	Type	Register Cluster
TXD	READER	TXD
RXD	WRITER	RXD

Table 145: SPIM EasyDMA Channels

For detailed information regarding the use of EasyDMA, see EasyDMA on page 150.

The .PTR and .MAXCNT registers are double-buffered. They can be updated and prepared for the next transmission immediately after having received the STARTED event.

The SPI master will automatically stop transmitting after TXD.MAXCNT bytes have been transmitted and RXD.MAXCNT bytes have been received. If RXD.MAXCNT is larger than TXD.MAXCNT, the remaining transmitted bytes will contain the value defined in the ORC register. If TXD.MAXCNT is larger than RXD.MAXCNT, the additional received bytes will be discarded.

The ENDRX/ENDTX events indicate that EasyDMA has finished accessing respectively the RX/TX buffer in RAM. The END event gets generated when both RX and TX are finished accessing the buffers in RAM.

If several AHB bus masters try to access the same AHB slave at the same time, AHB bus congestion might occur, and the behavior of the EasyDMA channel will depend on the SPIM instance. Refer to Instances on page 551 for information about what behavior is supported in the various instances.

7.30.6 Low power

To ensure lowest possible power consumption when the peripheral is not needed stop and disable SPIM.

The STOP task may not be always needed (the peripheral might already be stopped), but if it is sent, software shall wait until the STOPPED event was received as a response before disabling the peripheral through the ENABLE register.

7.30.7 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50008000 0x40008000	APPLICATION	SPIM	SPIM0 : S SPIM0 : NS	US	SA	SPI master 0	Not supported: > 8 Mbps data rate, CSNPOL register, DCX functionality, IFTIMING.x registers, hardware CSN control (PSEL.CSN), stalling mechanism during AHB bus contention
0x50009000 0x40009000	APPLICATION	SPIM	SPIM1 : S SPIM1 : NS	US	SA	SPI master 1	Not supported: > 8 Mbps data rate, CSNPOL register, DCX functionality, IFTIMING.x registers, hardware CSN control (PSEL.CSN), stalling mechanism during AHB bus contention
0x5000A000 0x4000A000	APPLICATION	SPIM	SPIM4 : S SPIM4 : NS	US	SA	SPI master 4 (high-speed)	Up to 32 Mbps SPI when using dedicated pins
0x5000B000 0x4000B000	APPLICATION	SPIM	SPIM2 : S SPIM2 : NS	US	SA	SPI master 2	Not supported: > 8 Mbps data rate, CSNPOL register, DCX functionality, IFTIMING.x registers, hardware CSN control (PSEL.CSN), stalling mechanism during AHB bus contention
0x5000C000 0x4000C000	APPLICATION	SPIM	SPIM3 : S SPIM3 : NS	US	SA	SPI master 3	Not supported: > 8 Mbps data rate, CSNPOL register, DCX functionality, IFTIMING.x registers, hardware CSN control (PSEL.CSN), stalling mechanism during AHB bus contention
0x41013000	NETWORK	SPIM	SPIM0	NS	NA	SPI master 0	Not supported: > 8 Mbps data rate, IFTIMING.x registers, hardware CSN control (PSEL.CSN), stalling mechanism during AHB bus contention

Table 146: Instances

Register	Offset	Security	Description
TASKS_START	0x010		Start SPI transaction
TASKS_STOP	0x014		Stop SPI transaction
TASKS_SUSPEND	0x01C		Suspend SPI transaction
TASKS_RESUME	0x020		Resume SPI transaction
SUBSCRIBE_START	0x090		Subscribe configuration for task START
SUBSCRIBE_STOP	0x094		Subscribe configuration for task STOP

Register	Offset	Security	Description
SUBSCRIBE_SUSPEND	0x09C		Subscribe configuration for task SUSPEND
SUBSCRIBE_RESUME	0x0A0		Subscribe configuration for task RESUME
EVENTS_STOPPED	0x104		SPI transaction has stopped
EVENTS_ENDRX	0x110		End of RXD buffer reached
EVENTS_END	0x118		End of RXD buffer and TXD buffer reached
EVENTS_ENDTX	0x120		End of TXD buffer reached
EVENTS_STARTED	0x14C		Transaction started
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
PUBLISH_ENDRX	0x190		Publish configuration for event ENDRX
PUBLISH_END	0x198		Publish configuration for event END
PUBLISH_ENDTX	0x1A0		Publish configuration for event ENDTX
PUBLISH_STARTED	0x1CC		Publish configuration for event STARTED
SHORTS	0x200		Shortcuts between local events and tasks
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
STALLSTAT	0x400		Stall status for EasyDMA RAM accesses. The fields in this register are set to STALL by hardware whenever a stall occurres and can be cleared (set to NOSTALL) by the CPU.
ENABLE	0x500		Enable SPIM
PSEL.SCK	0x508		Pin select for SCK
PSEL.MOSI	0x50C		Pin select for MOSI signal
PSEL.MISO	0x510		Pin select for MISO signal
PSEL.CSN	0x514		Pin select for CSN
FREQUENCY	0x524		SPI frequency. Accuracy depends on the HFCLK source selected.
RXD.PTR	0x534		Data pointer
RXD.MAXCNT	0x538		Maximum number of bytes in receive buffer
RXD.AMOUNT	0x53C		Number of bytes transferred in the last transaction
RXD.LIST	0x540		EasyDMA list type
TXD.PTR	0x544		Data pointer
TXD.MAXCNT	0x548		Number of bytes in transmit buffer
TXD.AMOUNT	0x54C		Number of bytes transferred in the last transaction
TXD.LIST	0x550		EasyDMA list type
CONFIG	0x554		Configuration register
IFTIMING.RXDELAY	0x560		Sample delay for input serial data on MISO
IFTIMING.CSNDUR	0x564		Minimum duration between edge of CSN and edge of SCK and minimum duration CSN must stay high between transactions
CSNPOL	0x568		Polarity of CSN output
PSELDCX	0x56C		Pin select for DCX signal
DCXCNT	0x570		DCX configuration
ORC	0x5C0		Byte transmitted after TXD.MAXCNT bytes have been transmitted in the case when RXD.MAXCNT is greater than TXD.MAXCNT

Table 147: Register overview

7.30.7.1 TASKS_START

Address offset: 0x010

Start SPI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_START			Start SPI transaction
			Trigger	1	Trigger task

7.30.7.2 TASKS_STOP

Address offset: 0x014

Stop SPI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Stop SPI transaction
			Trigger	1	Trigger task

7.30.7.3 TASKS_SUSPEND

Address offset: 0x01C

Suspend SPI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_SUSPEND			Suspend SPI transaction
			Trigger	1	Trigger task

7.30.7.4 TASKS_RESUME

Address offset: 0x020

Resume SPI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_RESUME			Resume SPI transaction
			Trigger	1	Trigger task

7.30.7.5 SUBSCRIBE_START

Address offset: 0x090

Subscribe configuration for task START

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task START will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.30.7.6 SUBSCRIBE_STOP

Address offset: 0x094

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.30.7.7 SUBSCRIBE_SUSPEND

Address offset: 0x09C

Subscribe configuration for task SUSPEND

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task SUSPEND will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.30.7.8 SUBSCRIBE_RESUME

Address offset: 0x0A0

Subscribe configuration for task RESUME

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task RESUME will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.30.7.9 EVENTS_STOPPED

Address offset: 0x104

SPI transaction has stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_STOPPED			SPI transaction has stopped
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.30.7.10 EVENTS_ENDRX

Address offset: 0x110

End of RXD buffer reached

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDRX			End of RXD buffer reached
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.30.7.11 EVENTS_END

Address offset: 0x118

End of RXD buffer and TXD buffer reached

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_END			End of RXD buffer and TXD buffer reached
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.30.7.12 EVENTS_ENDTX

Address offset: 0x120

End of TXD buffer reached

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDTX			End of TXD buffer reached
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.30.7.13 EVENTS_STARTED

Address offset: 0x14C

Transaction started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_STARTED			Transaction started
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.30.7.14 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event STOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event STOPPED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.30.7.15 PUBLISH_ENDRX

Address offset: 0x190

Publish configuration for event ENDRX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDRX will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.30.7.16 PUBLISH_END

Address offset: 0x198

Publish configuration for event END

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event END will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.30.7.17 PUBLISH_ENDTX

Address offset: 0x1A0

Publish configuration for event ENDTX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDTX will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.30.7.18 PUBLISH_STARTED

Address offset: 0x1CC

Publish configuration for event STARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event STARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.30.7.19 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	END_START			Shortcut between event END and task START
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.30.7.20 INTENSET

Address offset: 0x304

Enable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					E D C B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STOPPED			Write '1' to enable interrupt for event STOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	ENDRX			Write '1' to enable interrupt for event ENDRX
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	END			Write '1' to enable interrupt for event END
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	ENDTX			Write '1' to enable interrupt for event ENDTX
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	STARTED			Write '1' to enable interrupt for event STARTED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.30.7.21 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					E D C B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STOPPED			Write '1' to disable interrupt for event STOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	ENDRX			Write '1' to disable interrupt for event ENDRX
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	END			Write '1' to disable interrupt for event END
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	ENDTX			Write '1' to disable interrupt for event ENDTX
			Clear	1	Disable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	STARTED			Write '1' to disable interrupt for event STARTED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.30.7.22 STALLSTAT

Address offset: 0x400

Stall status for EasyDMA RAM accesses. The fields in this register are set to STALL by hardware whenever a stall occurres and can be cleared (set to NOSTALL) by the CPU.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	TX		[1..0]	Stall status for EasyDMA RAM reads
			NOSTALL	0	No stall
			STALL	1	A stall has occurred
B	RW	RX		[1..0]	Stall status for EasyDMA RAM writes
			NOSTALL	0	No stall
			STALL	1	A stall has occurred

7.30.7.23 ENABLE

Address offset: 0x500

Enable SPIM

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ENABLE			Enable or disable SPIM
			Disabled	0	Disable SPIM
			Enabled	7	Enable SPIM

7.30.7.24 PSEL.SCK

Address offset: 0x508

Pin select for SCK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.30.7.25 PSEL.MOSI

Address offset: 0x50C

Pin select for MOSI signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.30.7.26 PSEL.MISO

Address offset: 0x510

Pin select for MISO signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.30.7.27 PSEL.CSN

Address offset: 0x514

Pin select for CSN

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.30.7.28 FREQUENCY

Address offset: 0x524

SPI frequency. Accuracy depends on the HFCLK source selected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x04000000				**0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	FREQUENCY			SPI master data rate
			K125	0x02000000	125 kbps
			K250	0x04000000	250 kbps
			K500	0x08000000	500 kbps
			M1	0x10000000	1 Mbps
			M2	0x20000000	2 Mbps
			M4	0x40000000	4 Mbps
			M8	0x80000000	8 Mbps
			M16	0x0A000000	16 Mbps
			M32	0x14000000	32 Mbps

7.30.7.29 RXD.PTR

Address offset: 0x534

Data pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.30.7.30 RXD.MAXCNT

Address offset: 0x538

Maximum number of bytes in receive buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in receive buffer

7.30.7.31 RXD.AMOUNT

Address offset: 0x53C

Number of bytes transferred in the last transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transferred in the last transaction

7.30.7.32 RXD.LIST

Address offset: 0x540

EasyDMA list type

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LIST			List type
			Disabled	0	Disable EasyDMA list
			ArrayList	1	Use array list

7.30.7.33 TXD.PTR

Address offset: 0x544

Data pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.30.7.34 TXD.MAXCNT

Address offset: 0x548

Number of bytes in transmit buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in transmit buffer

7.30.7.35 TXD.AMOUNT

Address offset: 0x54C

Number of bytes transferred in the last transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transferred in the last transaction

7.30.7.36 TXD.LIST

Address offset: 0x550

EasyDMA list type

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LIST			List type
			Disabled	0	Disable EasyDMA list
			ArrayList	1	Use array list

7.30.7.37 CONFIG

Address offset: 0x554

Configuration register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	ORDER			Bit order
			MsbFirst	0	Most significant bit shifted out first
			LsbFirst	1	Least significant bit shifted out first
B	RW	CPHA			Serial clock (SCK) phase
			Leading	0	Sample on leading edge of clock, shift serial data on trailing edge
			Trailing	1	Sample on trailing edge of clock, shift serial data on leading edge
C	RW	CPOL			Serial clock (SCK) polarity
			ActiveHigh	0	Active high
			ActiveLow	1	Active low

7.30.7.38 IFTIMING.RXDELAY

Address offset: 0x560

Sample delay for input serial data on MISO

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A
Reset 0x00000002				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	RXDELAY		[7..0]	Sample delay for input serial data on MISO. The value specifies the number of 64 MHz clock cycles (15.625 ns) delay from the the sampling edge of SCK (leading edge for CONFIG.CPHA = 0, trailing edge for CONFIG.CPHA = 1) until the input serial data is sampled. As en example, if RXDELAY = 0 and CONFIG.CPHA = 0, the input serial data is sampled on the rising edge of SCK.

7.30.7.39 IFTIMING.CSNDUR

Address offset: 0x564

Minimum duration between edge of CSN and edge of SCK and minimum duration CSN must stay high between transactions

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000002				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CSNDUR		[0xFF..0]	Minimum duration between edge of CSN and edge of SCK and minimum duration CSN must stay high between transactions. The value is specified in number of 64 MHz clock cycles (15.625 ns).

7.30.7.40 CSNPOL

Address offset: 0x568

Polarity of CSN output

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CSNPOL			Polarity of CSN output
			LOW	0	Active low (idle state high)
			HIGH	1	Active high (idle state low)

7.30.7.41 PSELDCX

Address offset: 0x56C

Pin select for DCX signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.30.7.42 DCXCNT

Address offset: 0x570

DCX configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	DCXCNT		0x0..0xF	This register specifies the number of command bytes preceding the data bytes. The PSEL.DCX line will be low during transmission of command bytes and high during transmission of data bytes. Value 0xF indicates that all bytes are command bytes.

7.30.7.43 ORC

Address offset: 0x5C0

Byte transmitted after TXD.MAXCNT bytes have been transmitted in the case when RXD.MAXCNT is greater than TXD.MAXCNT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	ORC			Byte transmitted after TXD.MAXCNT bytes have been transmitted in the case when RXD.MAXCNT is greater than TXD.MAXCNT.

7.30.8 Electrical specification

7.30.8.1 Timing specifications

Symbol	Description	Min.	Typ.	Max.	Units
f_{SPIM}	Bit rates for SPIM[20]			16[21]	Mbps
$t_{SPIM,START}$	Time from START task to transmission started		1		µs
$t_{SPIM,CSCK}$	SCK period	125			ns

[20] High bit rates may require GPIOs to be set as High Drive, see GPIO chapter for more details.
[21] Application core SPIM4 supports 32 Mbps write speed when running at 128 MHz.

Symbol	Description	Min.	Typ.	Max.	Units
$t_{SPIM,RSCK,LD}$	SCK rise time, standard drive[22]			$t_{RF,25pF}$	
$t_{SPIM,RSCK,HD}$	SCK rise time, high drive[22]			$t_{HRF,25pF}$	
$t_{SPIM,FSCK,LD}$	SCK fall time, standard drive[22]			$t_{RF,25pF}$	
$t_{SPIM,FSCK,HD}$	SCK fall time, high drive[22]			$t_{HRF,25pF}$	
$t_{SPIM,WHSCK}$	SCK high time[22]	(t_{CSCK}/2) - t_{RSCK} - 1.5 ns			
$t_{SPIM,WLSCK}$	SCK low time[22]	(t_{CSCK}/2) - t_{FSCK} - 1.5 ns			
$t_{SPIM,SUMI}$	MISO to CLK edge setup time	19			ns
$t_{SPIM,HMI}$	CLK edge to MISO hold time	10			ns
$t_{SPIM,VMO}$	CLK edge to MOSI valid, SCK frequency ≤ 8 MHz			59	ns
$t_{SPIM,VMO,HS}$	CLK edge to MOSI valid, SCK frequency > 8 MHz	..	..	..	ns
$t_{SPIM,HMO}$	MOSI hold time after CLK edge	10			ns

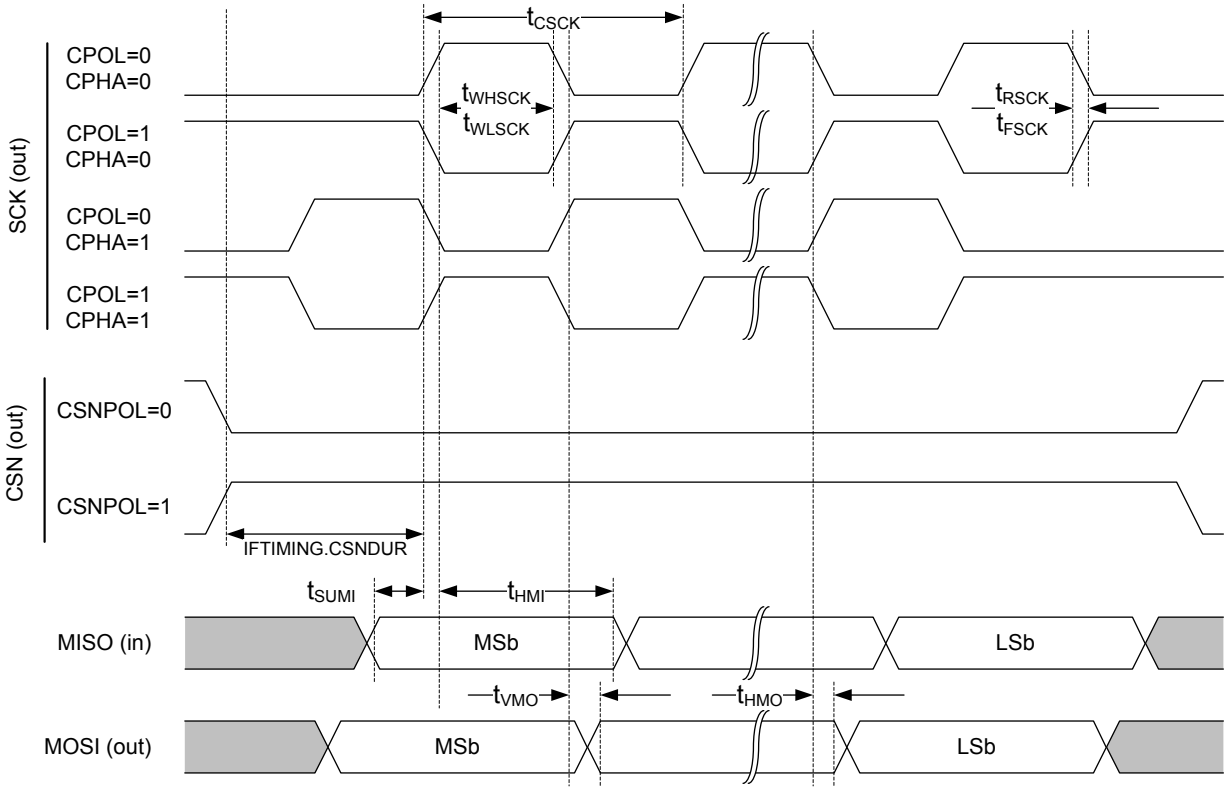


Figure 199: SPIM timing diagram

7.31 SPIS — Serial peripheral interface slave with EasyDMA

SPI slave (SPIS) is implemented with EasyDMA support for ultra-low power serial communication from an external SPI master. EasyDMA in conjunction with hardware-based semaphore mechanisms removes all real-time requirements associated with controlling the SPI slave from a low priority CPU execution context.

[22] At 25pF load, including GPIO pin capacitance, see GPIO spec.

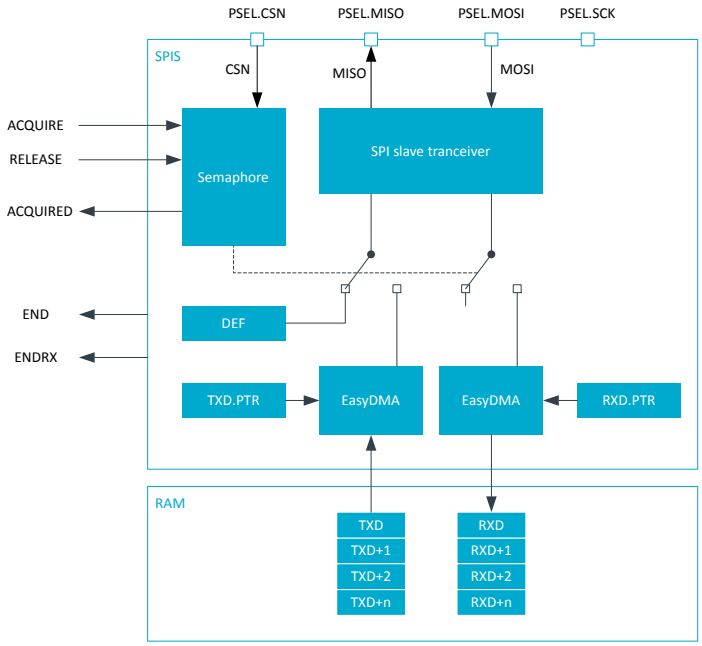


Figure 200: SPI slave

SPIS supports SPI modes 0 through 3. The CONFIG register allows setting CPOL and CPHA appropriately.

Mode	Clock polarity	Clock phase
	CPOL	CPHA
SPI_MODE0	0 (Active High)	0 (Trailing Edge)
SPI_MODE1	0 (Active High)	1 (Leading Edge)
SPI_MODE2	1 (Active Low)	0 (Trailing Edge)
SPI_MODE3	1 (Active Low)	1 (Leading Edge)

Table 148: SPI modes

7.31.1 Shared resources

The SPI slave shares registers and other resources with other peripherals that have the same ID as the SPI slave. Therefore, you must disable all peripherals that have the same ID as the SPI slave before the SPI slave can be configured and used.

Disabling a peripheral that has the same ID as the SPI slave will not reset any of the registers that are shared with the SPI slave. It is important to configure all relevant SPI slave registers explicitly to secure that it operates correctly.

The Instantiation table in Peripherals on page 146 shows which peripherals have the same ID as the SPI slave.

7.31.2 EasyDMA

SPIS implements EasyDMA for accessing RAM without CPU involvement.

SPIS implements the EasyDMA channels found in the following table.

Channel	Type	Register Cluster
TXD	READER	TXD
RXD	WRITER	RXD

Table 149: SPIS EasyDMA Channels

For detailed information regarding the use of EasyDMA, see EasyDMA on page 150.

If RXD.MAXCNT is larger than TXD.MAXCNT, the remaining transmitted bytes will contain the value defined in the ORC register.

The END event indicates that EasyDMA has finished accessing the buffer in RAM.

7.31.3 SPI slave operation

SPI slave uses two memory pointers, RXD.PTR and TXD.PTR, that point to the RXD buffer (receive buffer) and TXD buffer (transmit buffer) respectively. Since these buffers are located in RAM, which can be accessed by both the SPI slave and the CPU, a hardware based semaphore mechanism is implemented to enable safe sharing.

Before the CPU can safely update the RXD.PTR and TXD.PTR pointers it must first acquire the SPI semaphore. The CPU can acquire the semaphore by triggering the ACQUIRE task and then receiving the ACQUIRED event. When the CPU has updated the RXD.PTR and TXD.PTR pointers the CPU must release the semaphore before the SPI slave will be able to acquire it.

The CPU releases the semaphore by triggering the RELEASE task, this is illustrated in SPI transaction when shortcut between END and ACQUIRE is enabled on page 569. Triggering the RELEASE task when the semaphore is not granted to the CPU will have no effect. See Semaphore operation on page 569 for more information

If the CPU is not able to reconfigure the TXD.PTR and RXD.PTR between granted transactions, the same TX data will be clocked out and the RX buffers will be overwritten. To prevent this from happening, the END_ACQUIRE shortcut can be used. With this shortcut enabled, the semaphore will be handed over to the CPU automatically after the granted transaction has completed, giving the CPU the ability to update the TXPTR and RXPTR between every granted transaction.

The ENDRX event is generated when the RX buffer has been filled.

The RXD.MAXCNT register specifies the maximum number of bytes the SPI slave can receive in one granted transaction. If the SPI slave receives more than RXD.MAXCNT number of bytes, an OVERFLOW will be indicated in the STATUS register and the incoming bytes will be discarded.

The TXD.MAXCNT parameter specifies the maximum number of bytes the SPI slave can transmit in one granted transaction. If the SPI slave is forced to transmit more than TXD.MAXCNT number of bytes, an OVERREAD will be indicated in the STATUS register and the ORC character will be clocked out.

The RXD.AMOUNT and TXD.AMOUNT registers are updated when a granted transaction is completed. The TXD.AMOUNT register indicates how many bytes were read from the TX buffer in the last transaction, that is, ORC (over-read) characters are not included in this number. Similarly, the RXD.AMOUNT register indicates how many bytes were written into the RX buffer in the last transaction.

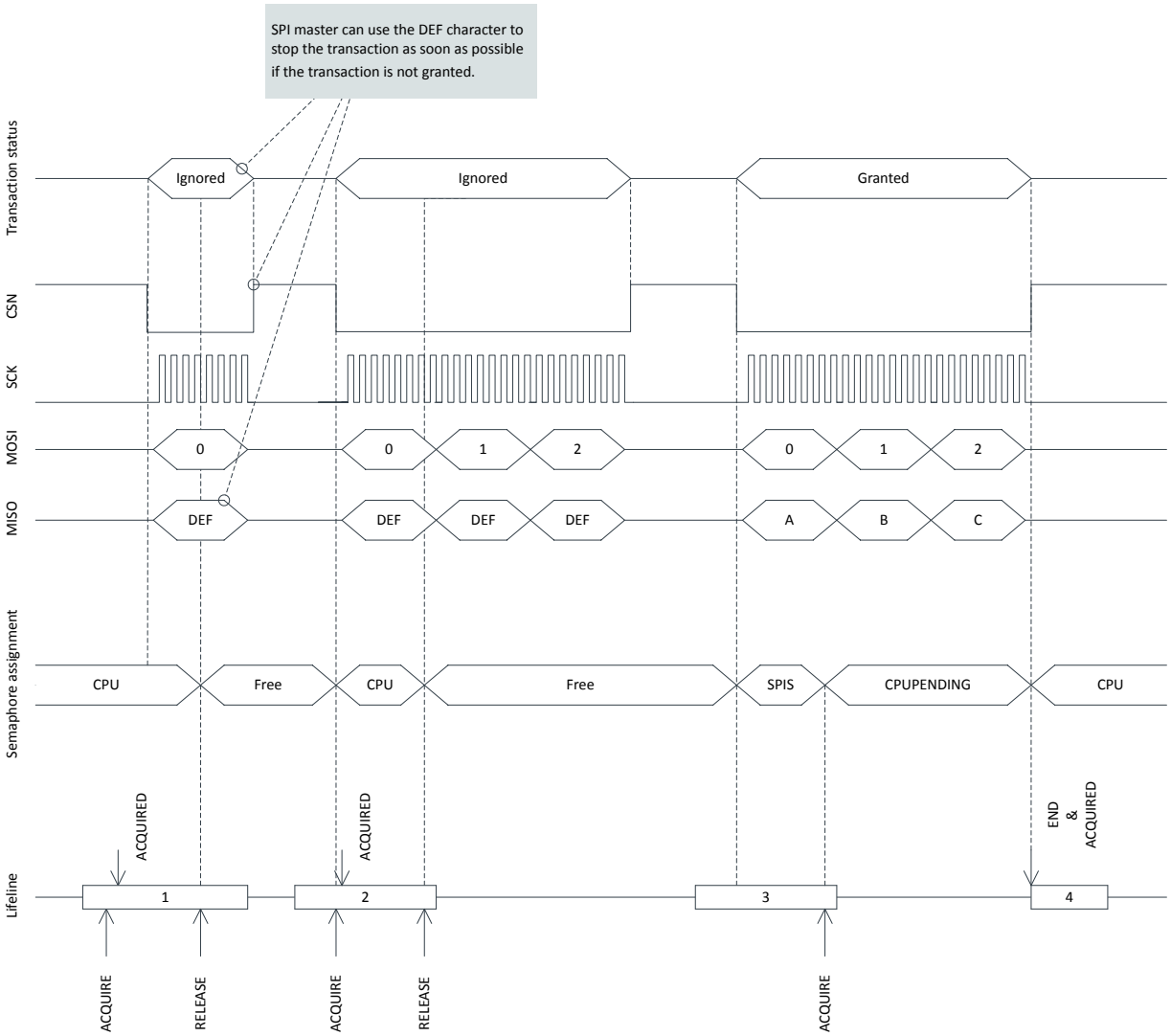


Figure 201: SPI transaction when shortcut between END and ACQUIRE is enabled

7.31.4 Semaphore operation

The semaphore is a mechanism implemented inside the SPI slave that prevents simultaneous access to the data buffers by the SPI slave and CPU.

The semaphore is by default assigned to the CPU after the SPI slave is enabled. No ACQUIRED event will be generated for this initial semaphore handover. An ACQUIRED event will be generated immediately if the ACQUIRE task is triggered while the semaphore is assigned to the CPU. The figure SPI semaphore FSM on page 570 illustrates the transitions between states in the semaphore based on the relevant tasks and events.

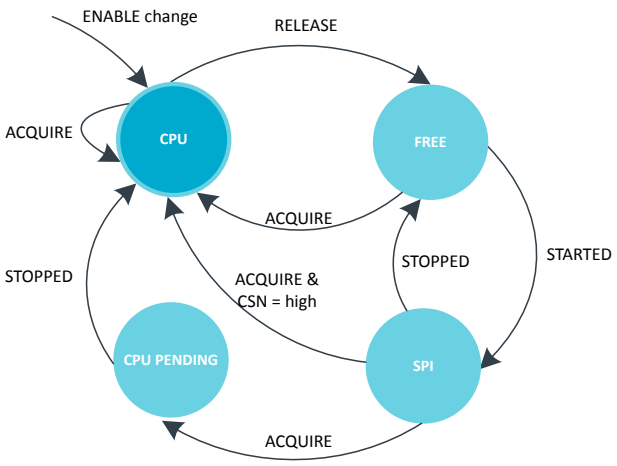


Figure 202: SPI semaphore FSM

Note: The semaphore mechanism does not, at any time, prevent the CPU from performing read or write access to the RXD.PTR register, the TXD.PTR registers, or the RAM that these pointers are pointing to. The semaphore is only telling when these can be updated by the CPU so that safe sharing is achieved.

The SPI slave will try to acquire the semaphore when STARTED event is detected, the even also indicates that CSN is currently low. If the SPI slave does not manage to acquire the semaphore at this point (i.e., it is under CPU's control), the transaction will be ignored. This means that all incoming data on MOSI will be discarded, and the DEF (default) character will be clocked out on the MISO line throughout the whole transaction. This will also be the case even if the semaphore is released by the CPU during the transaction. In case of a race condition where the CPU and the SPI slave try to acquire the semaphore at the same time, as illustrated in lifeline item 2 in figure SPI transaction when shortcut between END and ACQUIRE is enabled on page 569, the semaphore will be granted to the CPU.

If the SPI slave acquires the semaphore, the transaction will be granted. The incoming data on MOSI will be stored in the RXD buffer and the data in the TXD buffer will be clocked out on MISO.

When a granted transaction is completed and CSN goes high, the SPI slave will automatically release the semaphore and generate the END event.

As long as the semaphore is available, the SPI slave can be granted multiple transactions one after the other.

If the CPU tries to acquire the semaphore while it is assigned to the SPI slave, an immediate handover will not be granted. However, the semaphore will be handed over to the CPU as soon as the SPI slave has released the semaphore after the granted transaction is completed. If the END_ACQUIRE shortcut is enabled and the CPU has triggered the ACQUIRE task during a granted transaction, only one ACQUIRE request will be served following the END event.

7.31.5 Pin configuration

The CSN, SCK, MOSI, and MISO signals associated with the SPI slave are mapped to physical pins according to the configuration specified in the PSEL.CSN, PSEL.SCK, PSEL.MOSI, and PSEL.MISO registers respectively. If the CONNECT field of any of these registers is set to Disconnected, the associated SPI slave signal will not be connected to any physical pins.

The PSEL.CSN, PSEL.SCK, PSEL.MOSI, and PSEL.MISO registers and their configurations are only used as long as the SPI slave is enabled, and retained only as long as the device is in System ON mode, see POWER — Power control on page 43 chapter for more information about power modes. When the peripheral is disabled, the pins will behave as regular GPIOs, and use the configuration in their respective OUT bit field

and PIN_CNF[n] register. PSEL.CSN, PSEL.SCK, PSEL.MOSI, and PSEL.MISO must only be configured when the SPI slave is disabled.

To secure correct behavior in the SPI slave, the pins used by the SPI slave must be configured in the GPIO peripheral as described in GPIO configuration before enabling peripheral on page 571 before enabling the SPI slave. This is to secure that the pins used by the SPI slave are driven correctly if the SPI slave itself is temporarily disabled, or if the device temporarily enters System OFF. This configuration must be retained in the GPIO for the selected I/Os as long as the SPI slave is to be recognized by an external SPI master.

The MISO line is set in high impedance as long as the SPI slave is not selected with CSN.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

SPI signal	SPI pin	Direction	Output value	Comment
CSN	As specified in PSEL.CSN	Input	Not applicable	
SCK	As specified in PSEL.SCK	Input	Not applicable	
MOSI	As specified in PSEL.MOSI	Input	Not applicable	
MISO	As specified in PSEL.MISO	Input	Not applicable	Emulates that the SPI slave is not selected.

Table 150: GPIO configuration before enabling peripheral

7.31.6 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50008000 0x40008000	APPLICATION	SPIS	SPIS0 : S SPIS0 : NS	US	SA	SPI slave 0	
0x50009000 0x40009000	APPLICATION	SPIS	SPIS1 : S SPIS1 : NS	US	SA	SPI slave 1	
0x5000B000 0x4000B000	APPLICATION	SPIS	SPIS2 : S SPIS2 : NS	US	SA	SPI slave 2	
0x5000C000 0x4000C000	APPLICATION	SPIS	SPIS3 : S SPIS3 : NS	US	SA	SPI slave 3	
0x41013000	NETWORK	SPIS	SPIS0	NS	NA	SPI slave 0	

Table 151: Instances

Register	Offset	Security	Description
TASKS_ACQUIRE	0x024		Acquire SPI semaphore
TASKS_RELEASE	0x028		Release SPI semaphore, enabling the SPI slave to acquire it
SUBSCRIBE_ACQUIRE	0x0A4		Subscribe configuration for task ACQUIRE
SUBSCRIBE_RELEASE	0x0A8		Subscribe configuration for task RELEASE
EVENTS_END	0x104		Granted transaction completed
EVENTS_ENDRX	0x110		End of RXD buffer reached
EVENTS_ACQUIRED	0x128		Semaphore acquired
PUBLISH_END	0x184		Publish configuration for event END
PUBLISH_ENDRX	0x190		Publish configuration for event ENDRX
PUBLISH_ACQUIRED	0x1A8		Publish configuration for event ACQUIRED
SHORTS	0x200		Shortcuts between local events and tasks
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
SEMSTAT	0x400		Semaphore status register
STATUS	0x440		Status from last transaction
ENABLE	0x500		Enable SPI slave

Register	Offset	Security	Description	
PSEL.SCK	0x508		Pin select for SCK	
PSEL.MISO	0x50C		Pin select for MISO signal	
PSEL.MOSI	0x510		Pin select for MOSI signal	
PSEL.CSN	0x514		Pin select for CSN signal	
PSELSCK	0x508		Pin select for SCK	Deprecated
PSELMISO	0x50C		Pin select for MISO	Deprecated
PSELMOSI	0x510		Pin select for MOSI	Deprecated
PSELCSN	0x514		Pin select for CSN	Deprecated
RXD.PTR	0x534		RXD data pointer	
RXD.MAXCNT	0x538		Maximum number of bytes in receive buffer	
RXD.AMOUNT	0x53C		Number of bytes received in last granted transaction	
RXD.LIST	0x540		EasyDMA list type	
RXDPTR	0x534		RXD data pointer	Deprecated
MAXRX	0x538		Maximum number of bytes in receive buffer	Deprecated
AMOUNTRX	0x53C		Number of bytes received in last granted transaction	Deprecated
TXD.PTR	0x544		TXD data pointer	
TXD.MAXCNT	0x548		Maximum number of bytes in transmit buffer	
TXD.AMOUNT	0x54C		Number of bytes transmitted in last granted transaction	
TXD.LIST	0x550		EasyDMA list type	
TXDPTR	0x544		TXD data pointer	Deprecated
MAXTX	0x548		Maximum number of bytes in transmit buffer	Deprecated
AMOUNTTX	0x54C		Number of bytes transmitted in last granted transaction	Deprecated
CONFIG	0x554		Configuration register	
DEF	0x55C		Default character. Character clocked out in case of an ignored transaction.	
ORC	0x5C0		Over-read character	

Table 152: Register overview

7.31.6.1 TASKS_ACQUIRE

Address offset: 0x024

Acquire SPI semaphore

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_ACQUIRE			Acquire SPI semaphore
			Trigger	1	Trigger task

7.31.6.2 TASKS_RELEASE

Address offset: 0x028

Release SPI semaphore, enabling the SPI slave to acquire it

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_RELEASE			Release SPI semaphore, enabling the SPI slave to acquire it
			Trigger	1	Trigger task

7.31.6.3 SUBSCRIBE_ACQUIRE

Address offset: 0x0A4

Subscribe configuration for task ACQUIRE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task ACQUIRE will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.31.6.4 SUBSCRIBE_RELEASE

Address offset: 0x0A8

Subscribe configuration for task RELEASE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task RELEASE will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.31.6.5 EVENTS_END

Address offset: 0x104

Granted transaction completed

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_END			Granted transaction completed
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.31.6.6 EVENTS_ENDRX

Address offset: 0x110

End of RXD buffer reached

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDRX			End of RXD buffer reached
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.31.6.7 EVENTS_ACQUIRED

Address offset: 0x128

Semaphore acquired

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ACQUIRED			Semaphore acquired
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.31.6.8 PUBLISH_END

Address offset: 0x184

Publish configuration for event END

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event END will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.31.6.9 PUBLISH_ENDRX

Address offset: 0x190

Publish configuration for event ENDRX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDRX will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.31.6.10 PUBLISH_ACQUIRED

Address offset: 0x1A8

Publish configuration for event ACQUIRED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ACQUIRED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.31.6.11 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	END_ACQUIRE			Shortcut between event END and task ACQUIRE
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.31.6.12 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	END			Write '1' to enable interrupt for event END
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	ENDRX			Write '1' to enable interrupt for event ENDRX
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	ACQUIRED			Write '1' to enable interrupt for event ACQUIRED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.31.6.13 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	END			Write '1' to disable interrupt for event END
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	ENDRX			Write '1' to disable interrupt for event ENDRX
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	ACQUIRED			Write '1' to disable interrupt for event ACQUIRED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.31.6.14 SEMSTAT

Address offset: 0x400

Semaphore status register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**

ID	R/W	Field	Value ID	Value	Description
A	R	SEMSTAT			Semaphore status
			Free	0	Semaphore is free
			CPU	1	Semaphore is assigned to CPU
			SPIS	2	Semaphore is assigned to SPI slave
			CPUPending	3	Semaphore is assigned to SPI but a handover to the CPU is pending

7.31.6.15 STATUS

Address offset: 0x440

Status from last transaction

Note: 1

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	OVERREAD			TX buffer over-read detected, and prevented
			NotPresent	0	Read: error not present
			Present	1	Read: error present
			Clear	1	Write: clear error on writing '1'
B	RW	OVERFLOW			RX buffer overflow detected, and prevented
			NotPresent	0	Read: error not present
			Present	1	Read: error present
			Clear	1	Write: clear error on writing '1'

7.31.6.16 ENABLE

Address offset: 0x500

Enable SPI slave

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ENABLE			Enable or disable SPI slave
			Disabled	0	Disable SPI slave
			Enabled	2	Enable SPI slave

7.31.6.17 PSEL.SCK

Address offset: 0x508

Pin select for SCK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**

ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.31.6.18 PSEL.MISO

Address offset: 0x50C

Pin select for MISO signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**

ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.31.6.19 PSEL.MOSI

Address offset: 0x510

Pin select for MOSI signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.31.6.20 PSEL.CSN

Address offset: 0x514

Pin select for CSN signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.31.6.21 PSELSCK (Deprecated)

Address offset: 0x508

Pin select for SCK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PSELSCK		[0..31]	Pin number configuration for SPI SCK signal
			Disconnected	0xFFFFFFFF	Disconnect

7.31.6.22 PSELMISO (Deprecated)

Address offset: 0x50C

Pin select for MISO

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PSELMISO		[0..31]	Pin number configuration for SPI MISO signal
			Disconnected	0xFFFFFFFF	Disconnect

7.31.6.23 PSELMOSI (Deprecated)

Address offset: 0x510

Pin select for MOSI

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PSELMOSI		[0..31]	Pin number configuration for SPI MOSI signal
			Disconnected	0xFFFFFFFF	Disconnect

7.31.6.24 PSELCSN (Deprecated)

Address offset: 0x514

Pin select for CSN

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PSELCSN		[0..31]	Pin number configuration for SPI CSN signal
			Disconnected	0xFFFFFFFF	Disconnect

7.31.6.25 RXD.PTR

Address offset: 0x534

RXD data pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			RXD data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.31.6.26 RXD.MAXCNT

Address offset: 0x538

Maximum number of bytes in receive buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in receive buffer

7.31.6.27 RXD.AMOUNT

Address offset: 0x53C

Number of bytes received in last granted transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes received in the last granted transaction

7.31.6.28 RXD.LIST

Address offset: 0x540

EasyDMA list type

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LIST			List type
			Disabled	0	Disable EasyDMA list
			ArrayList	1	Use array list

7.31.6.29 RXDPTR (Deprecated)

Address offset: 0x534

RXD data pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	RXDPTR			RXD data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.31.6.30 MAXRX (Deprecated)

Address offset: 0x538

Maximum number of bytes in receive buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MAXRX		[1..0xFFFF]	Maximum number of bytes in receive buffer

7.31.6.31 AMOUNTRX (Deprecated)

Address offset: 0x53C

Number of bytes received in last granted transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNTRX		[1..0xFFFF]	Number of bytes received in the last granted transaction

7.31.6.32 TXD.PTR

Address offset: 0x544

TXD data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			TXD data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.31.6.33 TXD.MAXCNT

Address offset: 0x548

Maximum number of bytes in transmit buffer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in transmit buffer

7.31.6.34 TXD.AMOUNT

Address offset: 0x54C

Number of bytes transmitted in last granted transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transmitted in last granted transaction

7.31.6.35 TXD.LIST

Address offset: 0x550

EasyDMA list type

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LIST			List type
			Disabled	0	Disable EasyDMA list
			ArrayList	1	Use array list

7.31.6.36 TXDPTR (Deprecated)

Address offset: 0x544

TXD data pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	TXDPTR			TXD data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.31.6.37 MAXTX (Deprecated)

Address offset: 0x548

Maximum number of bytes in transmit buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXTX		[1..0xFFFF]	Maximum number of bytes in transmit buffer

7.31.6.38 AMOUNTTX (Deprecated)

Address offset: 0x54C

Number of bytes transmitted in last granted transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNTTX		[1..0xFFFF]	Number of bytes transmitted in last granted transaction

7.31.6.39 CONFIG

Address offset: 0x554

Configuration register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ORDER			Bit order
			MsbFirst	0	Most significant bit shifted out first
			LsbFirst	1	Least significant bit shifted out first
B	RW	CPHA			Serial clock (SCK) phase
			Leading	0	Sample on leading edge of clock, shift serial data on trailing edge
			Trailing	1	Sample on trailing edge of clock, shift serial data on leading edge
C	RW	CPOL			Serial clock (SCK) polarity
			ActiveHigh	0	Active high
			ActiveLow	1	Active low

7.31.6.40 DEF

Address offset: 0x55C

Default character. Character clocked out in case of an ignored transaction.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	DEF			Default character. Character clocked out in case of an ignored transaction.

7.31.6.41 ORC

Address offset: 0x5C0

Over-read character

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ORC			Over-read character. Character clocked out after an over-read of the transmit buffer.

7.31.7 Electrical specification

7.31.7.1 SPIS slave interface electrical specifications

Symbol	Description	Min.	Typ.	Max.	Units
f_{SPIS}	Bit rates for SPIS[23]			8[24]	Mbps
$t_{SPIS,START}$	Time from RELEASE task to receive/transmit (CSN active)	..	..	..	µs

[23] High bit rates may require GPIOs to be set as High Drive, see GPIO chapter for more details.

[24] The actual maximum data rate depends on the master's CLK to MISO and MOSI setup and hold timings.

7.31.7.2 Serial Peripheral Interface Slave (SPIS) timing specifications

Symbol	Description	Min.	Typ.	Max.	Units
$t_{SPIS,CSCKIN}$	SCK input period	125			ns
$t_{SPIS,RFSCKIN}$	SCK input rise/fall time			30	ns
$t_{SPIS,WHSCKIN}$	SCK input high time	30			ns
$t_{SPIS,WLSCKIN}$	SCK input low time	30			ns
$t_{SPIS,SUCSN}$	CSN to CLK setup time	1000[25]			ns
$t_{SPIS,HCSN}$	CLK to CSN hold time	1000			ns
$t_{SPIS,ASA}$	CSN to MISO driven			68	ns
$t_{SPIS,ASO}$	CSN to MISO valid[26]	1000			ns
$t_{SPIS,DISSO}$	CSN to MISO disabled[26]			68	ns
$t_{SPIS,CWH}$	CSN inactive time	300			ns
$t_{SPIS,VSO}$	CLK edge to MISO valid			31	ns
$t_{SPIS,HSO}$	MISO hold time after CLK edge	13			ns
$t_{SPIS,SUSI}$	MOSI to CLK edge setup time	19			ns
$t_{SPIS,HSI}$	CLK edge to MOSI hold time	10			ns

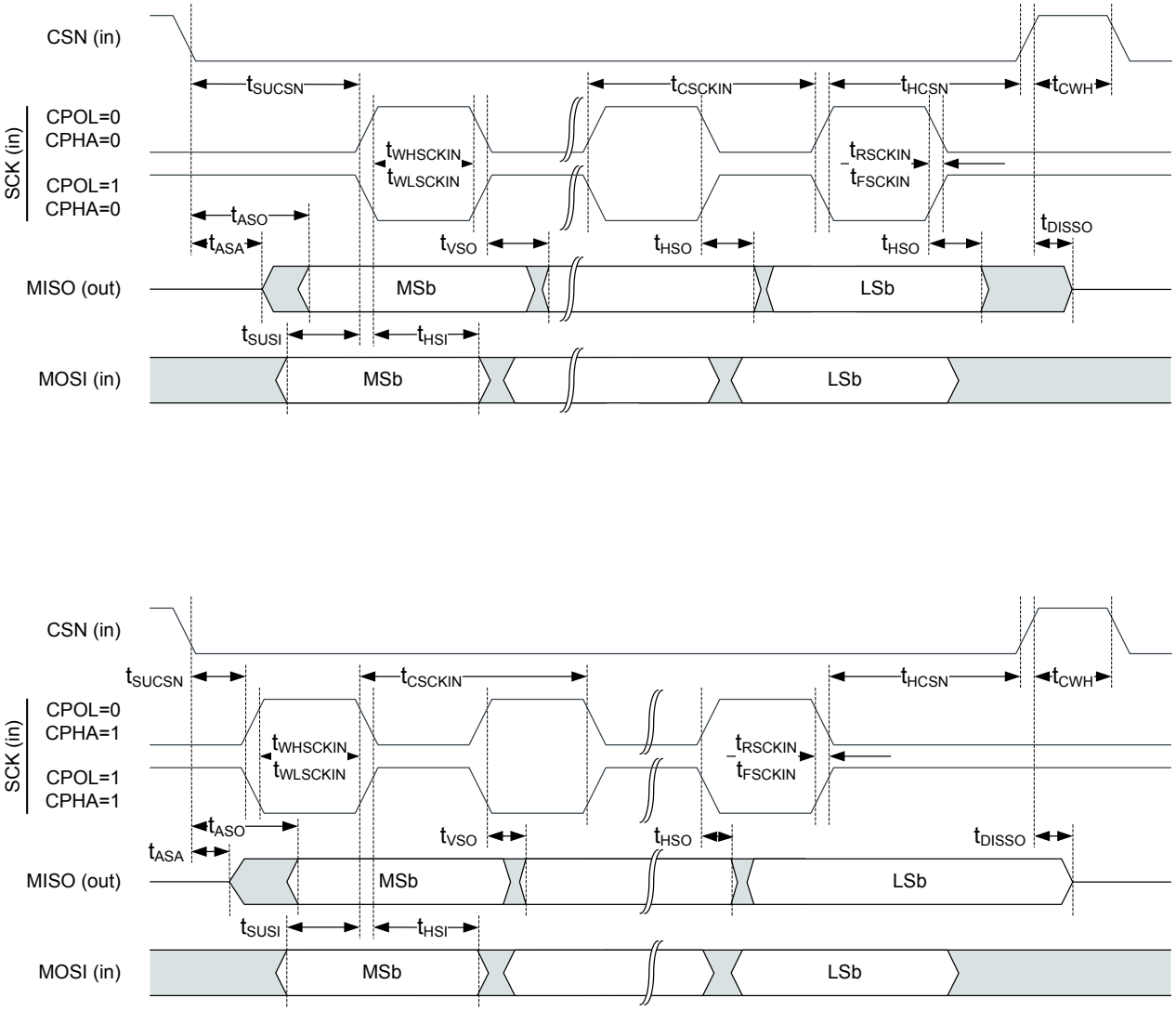


Figure 203: SPIS timing diagram

[25] Excluding any start-up delay for the high frequency clock in low power mode.
[26] At 25pF load, including GPIO capacitance, see GPIO spec.

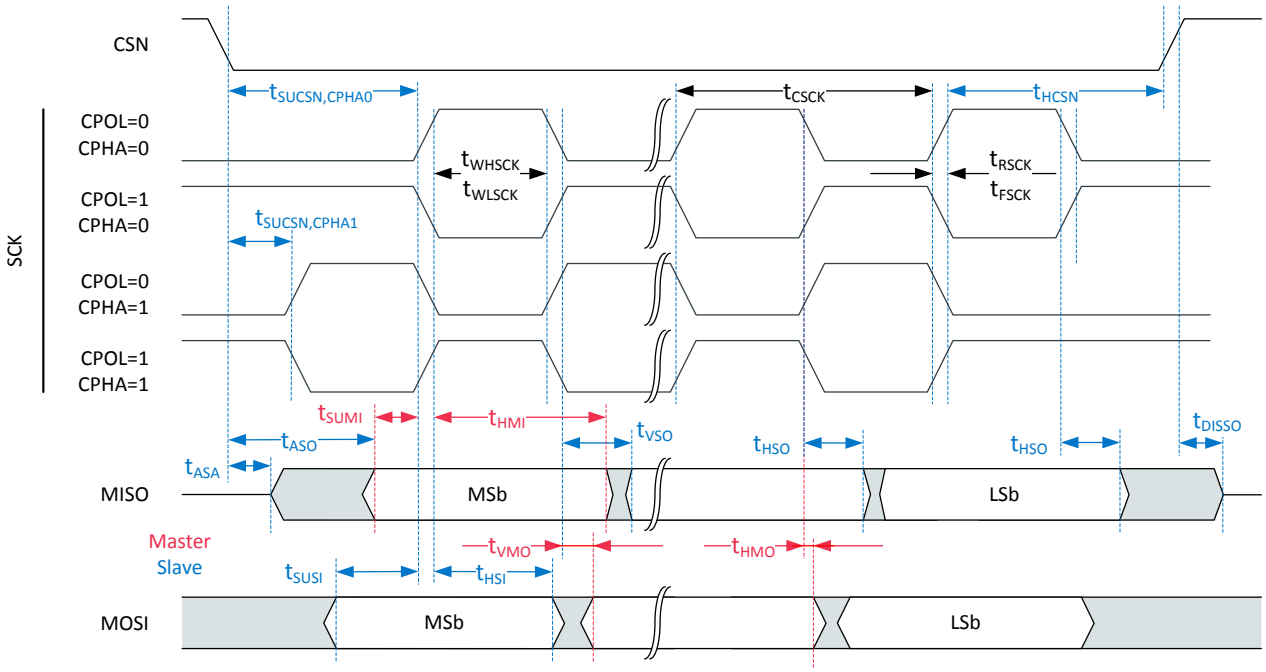


Figure 204: Common SPIM and SPIS timing diagram

7.32 SPU — System protection unit

SPU is the central point in the system to control access to memories, peripherals and other resources.

The main features of SPU are the following:

- Arm TrustZone support, allowing definition of secure, non-secure and non-secure callable memory regions
- Extended Arm TrustZone, protecting memory regions and peripherals from non-CPU devices like EasyDMA transfer
- Pin access protection, preventing non-secure code and peripherals from accessing secure pin resources
- DPPI access protection, realized by preventing non-secure code and peripherals to publish from or subscribe to secured DPPI channels
- External domain access protection, controlling access rights from other MCUs

7.32.1 General concepts

SPU provides a register interface to control the various internal logic blocks that monitor access to memory-mapped slave devices (RAM, flash, peripherals, etc) and other resources (device pins, DPPI channels, etc).

For memory-mapped devices like RAM, flash, and peripherals, the internal logic checks the address and attributes (e.g. read, write, execute, secure) of the incoming transfer to block it if necessary. A secure resource can be accessed by a given master based on the following factors:

- *CPU-type master* – By the security state of the CPU and the security state reported by SPU, for the address in the bus transfer.
- *Non-CPU master* – By the security attribute of the master that initiates the transfer, defined by a SPU register.

The Simplified view of SPU protection on page 586 shows a simplified view of the SPU registers controlling several internal modules.

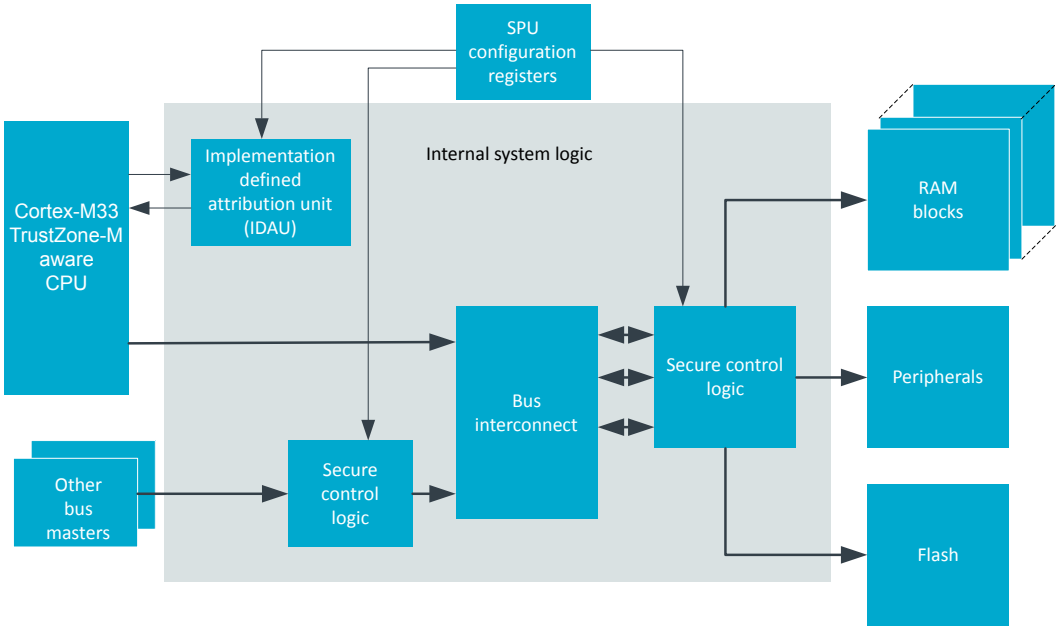


Figure 205: Simplified view of SPU protection

The protection logic implements a read-as-zero/write-ignore (RAZ/WI) policy where the following are true:

- A blocked read operation will always return a zero value on the bus, preventing information leak.
- A write operation to a forbidden region or peripheral will be ignored.

An error is reported through dedicated error signals. For security state violations from an M33 master this will be a SecureFault exception, for other violations this will be an SPU event. The SPU event can be configured to generate an interrupt towards the CPU.

Other resources like pins and DPPI channels are protected by comparing the security attributes of the protected resource with the security attribute of the peripheral that wants to access it. SPU is the only place where those security attributes can be configured.

7.32.1.1 Special considerations for Arm TrustZone for Cortex-M enabled system

SPU also controls custom logic for an Arm TrustZone for Cortex-M enabled CPU.

Custom logic is shown as the implementation defined attribution unit (IDAU) in figure Simplified view of SPU protection on page 586. Full support is provided for the following:

- Arm TrustZone for Cortex-M related instructions, like test target (TT) for reporting the security attributes of a region
- Non-secure callable (NSC) regions, to implement secure entry points from non-secure code

SPU provides the necessary registers to configure the security attributes for memory regions and peripherals. However, as a requirement to use SPU, the secure attribution unit (SAU) needs to be disabled and all memory set as non-secure in the Arm core. This will allow SPU to control the IDAU and set the security attribution of all addresses as originally intended.

7.32.2 Flash access control

The flash memory space is divided into 64 regions of 16 KiB, each with configurable permissions settings.

For each region, the following types of permissions can be configured:

Read

Allows data read access to the region. The code fetch from this region is not controlled by the read permission but by the execute permission described below.

Write
Allows write or page erase access to the region.

Execute
Allows code fetch from this region, even if data read is disabled.

Secure
Allows only bus accesses with the security attribute set to access the region.

Permissions can be set independently. For example, it is possible to configure a flash region to be accessible only through secure transfer, being read-only (no write allowed) and non-executable (no code fetch allowed). For each region, permissions can be set and then locked by using the FLASHREGION[n].PERM.LOCK bit, to prevent subsequent modifications.

The debugger can step through execute-protected memory regions.

The following figure shows the N=64 flash regions, each of size 16 KiB.

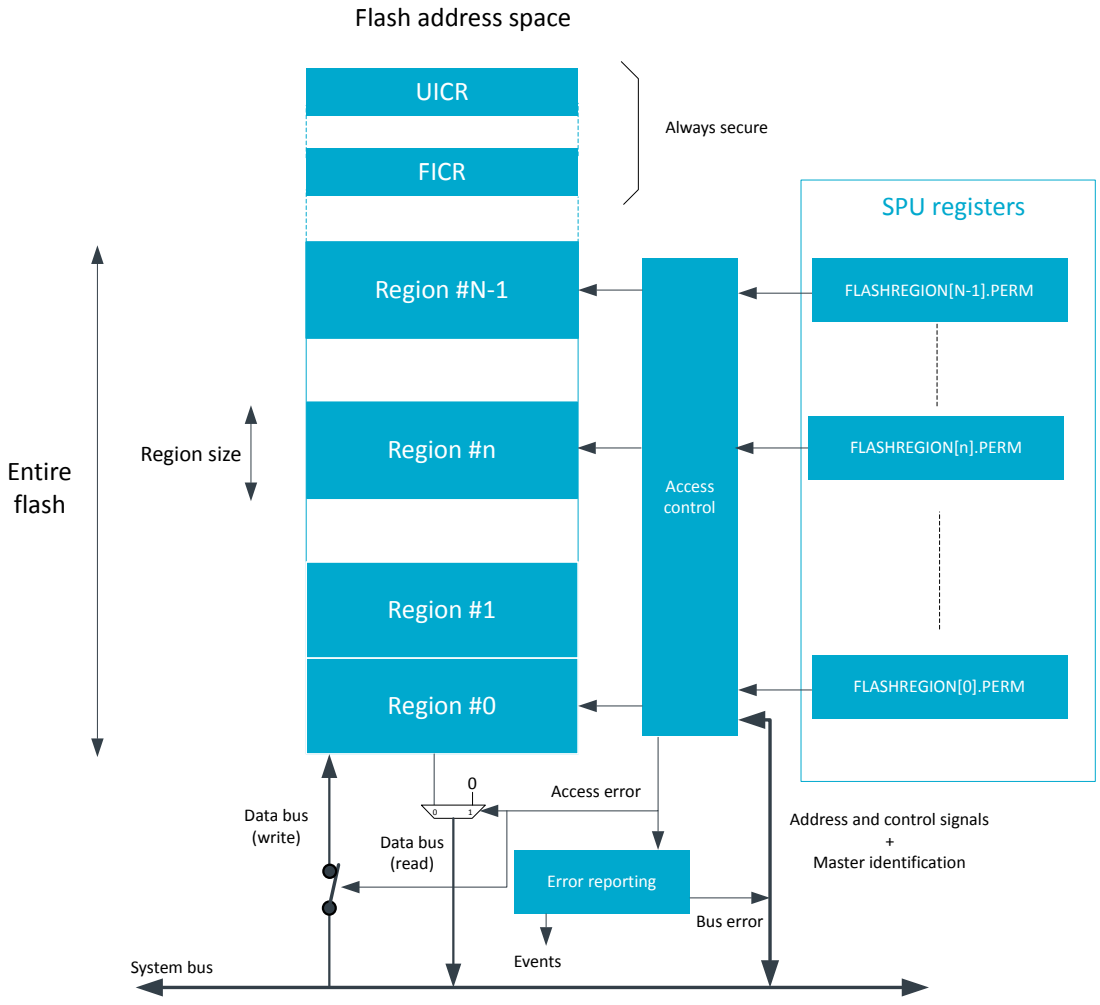


Figure 206: Flash memory regions

7.32.2.1 Non-secure callable (NSC) region definition in flash

SPU provides support for the definition of non-secure callable (NSC) sub-regions to allow non-secure to secure function calls.

A non-secure callable sub-region can only exist within an existing secure region and its definition is done using the following registers:

- FLASHNSC[n].REGION, used to select the secure region that will contain the NSC sub-region
- FLASHNSC[n].SIZE, used to define the size of the NSC sub-region within the secure region

The NSC sub-region will be defined starting with the highest address in that region and descending. The following figure illustrates the NSC sub-regions and the registers used for their definition.

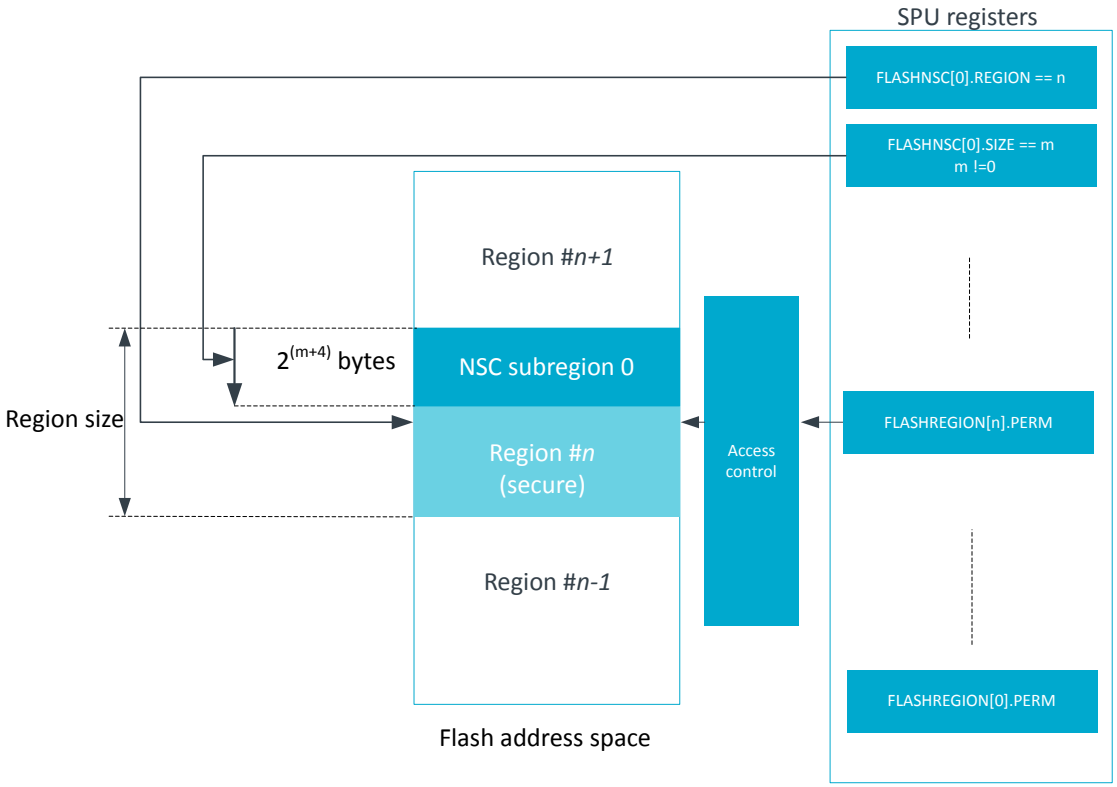


Figure 207: Non-secure callable region definition in the flash memory space

The NSC sub-region will only be defined when the following are true:

- FLASHNSC[i].SIZE value is non zero
- FLASHNSC[i].REGION defines a secure region

If FLASHNSC[i].REGION and FLASHNSCj].REGION have the same value, there is only one sub-region defined as NSC, with the size given by the maximum of FLASHNSC[i].SIZE and FLASHNSC[j].SIZE.

If FLASHNSC[i].REGION defines a non-secure region, then there is no non-secure callable region defined and the selected region stays non-secure.

7.32.2.2 Flash access error reporting

SPU and the logic controlled by it will respond with a certain behavior once an access violation is detected.

The following actions will happen once the logic controlled by SPU detects an access violation on one of the flash ports:

- The faulty transfer will be blocked.
- In case of a read transfer, the bus will be driven to zero.
- If supported by the master, feedback will be sent to the master through specific bus error signals. At the same time, SPU will receive an event that can optionally trigger a CPU interrupt.
- A SecureFault exception will be triggered if a security violation is detected for access from the CPU.
- A BusFault exception will be triggered when a read/write/execute protection violation is detected from the CPU.
- The FLASHACCERR event will be triggered if any access violations are detected for all master types except for the CPU security violation.

The following table summarizes the SPU behavior based on the type of initiator and access violation.

Master type	Security violation	Read/Write/Execute protection violation
Arm Cortex-M33	SecureFault exception	BusFault exception, FLASHACCERR event
EasyDMA	RAZ/WI, FLASHACCERR event	RAZ/WI, FLASHACCERR event
Other masters	RAZ/WI, FLASHACCERR event	RAZ/WI, FLASHACCERR event

Table 153: Error reporting for flash access errors

For the Arm Cortex-M33 master, the SecureFault exception will take precedence over the BusFault exception if a security violation occurs simultaneously with another type of violation.

7.32.2.3 UICR and FICR protections

The user information configuration registers (UICR) and factory information configuration registers (FICR) are always considered as secure. FICR registers are read-only. UICR registers can be read and written by secure code only.

Writing new values to user information configuration registers must follow the procedure described in NVMC — Non-volatile memory controller on page 330. Code execution from FICR and UICR address spaces will always be reported as an access violation except during a debug session.

7.32.3 RAM access control

The RAM memory space is divided into 64 regions of 8 KiB, each with configurable permissions settings.

For each region, the following types of permissions can be configured:

Read
Allows data read access to the region. Code fetch from this region is not controlled by the read permission but by the execute permission described below.

Write
Allows write access to the region.

Execute
Allows code fetch from this region.

Secure
Allows only bus accesses with the security attribute set to access the region.

Permissions can be set independently. For example, it is possible to configure a RAM region to be accessible only through secure transfer, being read-only (no write allowed) and non-executable (no code fetch allowed). For each region, permissions can be set and then locked to prevent subsequent modifications by using the RAMREGION[n].PERM.LOCK bit.

The following figure shows the RAM memory space divided into N=64 regions, each of 8 KiB.

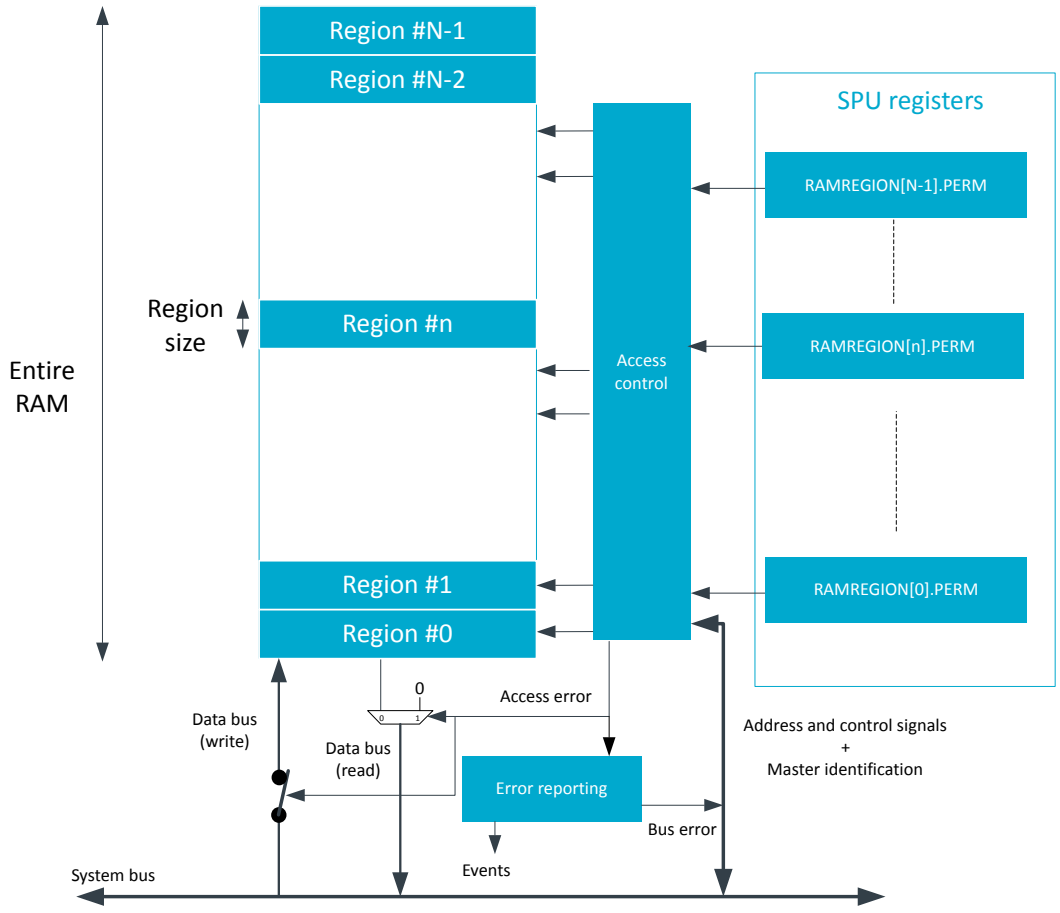


Figure 208: RAM memory regions

7.32.3.1 Non-secure callable (NSC) region definition in RAM

SPU provides support for the definition of non-secure callable (NSC) sub-regions to allow non-secure to secure function calls.

A non-secure callable sub-region can only exist within an existing secure region. It is defined by the following registers:

- RAMNSC[n].REGION, used to select the secure region that will contain the NSC sub-region
- RAMNSC[n].SIZE, used to define the size of the NSC sub-region within the secure region

The NSC sub-region will be defined starting with the highest address in that region and descending. The following figure shows the NSC sub-regions and the registers used for their definition.

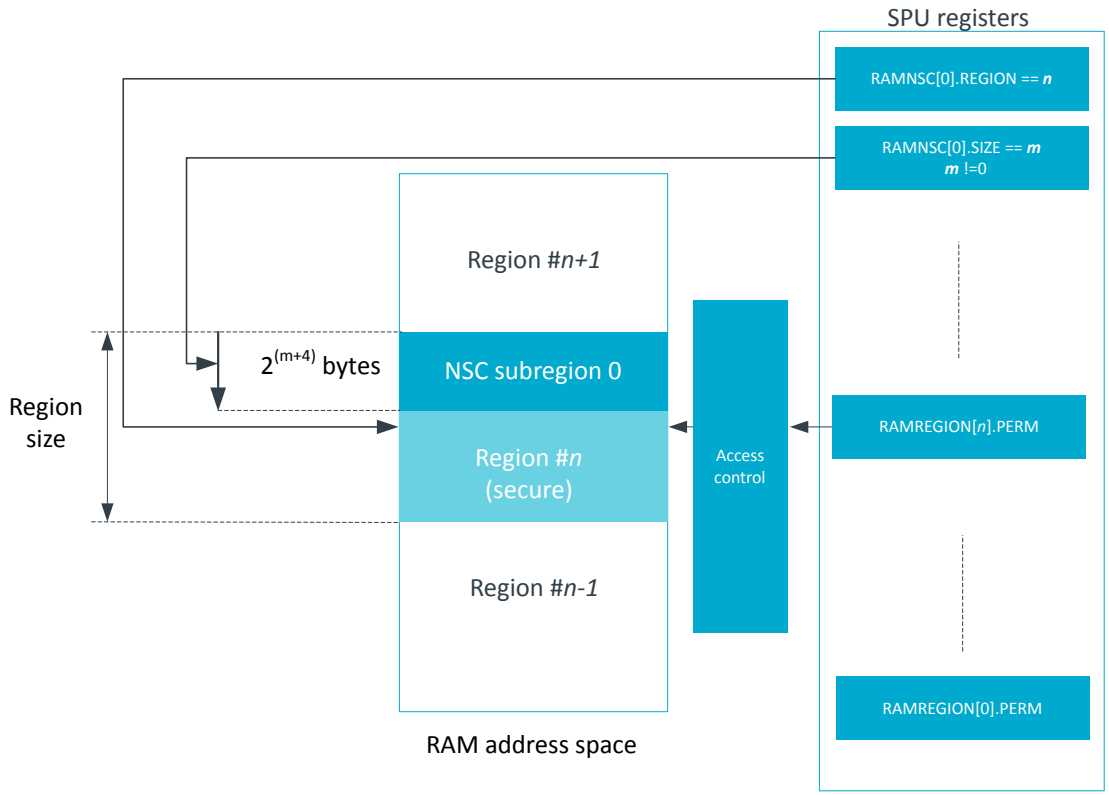


Figure 209: Non-secure callable region definition in the RAM memory space

The NSC sub-region will only be defined when the following are true:

- RAMNSC[*i*].SIZE value is non zero
- RAMNSC[*i*].REGION defines a secure region

If RAMNSC[*i*].REGION and RAMNSC[*j*].REGION have the same value, there is only one sub-region defined as NSC, with the size given by the maximum of RAMNSC[*i*].SIZE and RAMNSC[*j*].SIZE.

If RAMNSC[*i*].REGION defines a non-secure region, then there is no non-secure callable region defined and the selected region stays non-secure.

7.32.3.2 RAM access error reporting

SPU and the logic it controls will respond with a certain behavior once an access violation is detected.

The following actions will happen once the logic controlled by the SPU detects an access violation on one of the RAM ports:

- The faulty transfer will be blocked.
- In case of a read transfer, the bus will be driven to zero.
- If supported by the master, feedback will be sent to the master through specific bus error signals.
- A SecureFault exception will be triggered if security violation is detected for access from Arm Cortex-M33
- A BusFault exception will be triggered when read/write/execute protection violation is detected for Arm Cortex-M33. The SPU will also generate an event that can optionally trigger an interrupt towards the CPU.
- The RAMACCERR event will be triggered if any access violations are detected for all master types but for Arm Cortex-M33 security violation

The following table summarizes the SPU behavior based on the type of initiator and access violation.

Master type	Security violation	Read/Write/Execute protection violation
Arm Cortex-M33	SecureFault exception	BusFault exception, RAMACCERR event
EasyDMA	RAZ/WI, RAMACCERR event	RAZ/WI, RAMACCERR event
Other masters	RAZ/WI, RAMACCERR event	RAZ/WI, RAMACCERR event

Table 154: Error reporting for RAM access errors

For the Arm Cortex-M33 master, the SecureFault exception will take precedence over the BusFault exception if a security violation occurs simultaneously with another type of violation.

7.32.4 Peripheral access control

Access controls are defined by the characteristics of the peripheral.

Peripherals can have their security attribute set as one of the following:

Always secure
For a peripheral related to system control.

Always non-secure
For some general-purpose peripherals.

Configurable
For general-purpose peripherals that may be configured for secure only access.

The full list of peripherals and their corresponding security attributes can be found in Memory on page 18. For each peripheral with ID *n*, PERIPHID[*n*]. PERM will show whether the security attribute for this peripheral is configurable or not.

If not hardcoded, the security attribute can configured using the PERIPHID[*id*].PERM.

At reset, all user-selectable and split security peripherals are set to be secure with secure DMA where present.

Secure code can access both secure peripherals and non-secure peripherals.

7.32.4.1 Peripherals with split security

Peripherals with split security are defined to handle use-cases when both secure and non-secure code needs to control the same resource.

When peripherals with split security have their security attribute set to non-secure, access to specific registers and bitfields within some registers is dependent on the security attribute of the bus transfer. For example, some registers will not be accessible for a non-secure transfer.

When peripherals with split security have their security attribute set to secure, then only secure transfers can access their registers.

See Peripherals on page 146 for an overview of split security peripherals. Respective peripheral chapters explain the specific security behavior of each peripheral.

7.32.4.2 Peripheral address mapping

Peripherals that have non-secure security mapping have their address starting with 0x4XXXXXXX. Peripherals that have secure security mapping have their address starting with 0x5XXXXXXX.

Peripherals with a user-selectable security mapping are available at an address starting with the following:

- 0x4XXXXXXX, if the peripheral security attribute is set to non-secure
- 0x5XXXXXXX, if the peripheral security attribute is set to secure

Note:

Access to a secure peripheral using the 0x4XXXXXXX address range will result in bus error, regardless if the CPU is executing secure or non-secure code.

Similarly, a CPU running secure code attempting to access a non-secure peripheral using the 0x5XXXXXXX address range will result in bus error.

Peripherals with a split security mapping are available at an address starting with the following:

- 0x4XXXXXXX for non-secure access and 0x5XXXXXXX for secure access, if the peripheral security attribute is set to non-secure
  - Secure registers in the 0x4XXXXXXX range are not visible for secure or non-secure code, and an attempt to access such a register will result in write-ignore, read-as-zero behavior
  - Secure code can access both non-secure and secure registers in the 0x5XXXXXXX range
- 0x5XXXXXXX, if the peripheral security attribute is set to secure

Any attempt to access the 0x50000000 to 0x5FFFFFFF address range from non-secure code will be ignored and generate a SecureFault exception.

The following table contains the address mapping for the three peripheral types in each configuration.

Security-features and configuration	Mapped at 0x4XXXXXXX?	Mapped at 0x5XXXXXXX?
Secure peripheral	No	Yes
Non-secure peripheral	Yes	No
Split-security peripheral, with attribute=secure	No	Yes
Split-security peripheral, with attribute=non-secure	Yes, restricted functionality	Yes

Table 155: Peripheral's address mapping in relation to its security-features and configuration

7.32.4.3 Special considerations for peripherals with DMA master

Peripherals containing a DMA master can be configured so the security attribute of the DMA transfers is different from the security attribute of the peripheral itself. This allows a secure peripheral to do non-secure data transfers to or from the system memories.

The following conditions must be met:

- The DMA field of PERIPHID[n].PERM.SECURITYMAPPING should read as "SeparateAttribute"
- The peripheral itself should be secure (PERIPHID[n].PERM.SECATTR == 1)

Then it is possible to select the security attribute of the DMA transfers using the field DMASEC (PERIPHID[n].PERM.DMASEC == Secure and PERIPHID[n].PERM.DMASEC == NonSecure) in PERIPHID[n].PERM.

7.32.4.4 Peripheral access error reporting

Peripherals send error reports once access violation is detected.

The following actions will happen if the logic controlled by the SPU detects an access violation on one of the peripherals:

- The faulty transfer will be blocked.
- In case of a read transfer, the bus will be driven to zero.
- If supported by the master, feedback is sent to the master through specific bus error signals. If the master is a processor supporting Arm TrustZone for Cortex-M, a SecureFault exception will be generated for security related errors.
- The PERIPHACCERR event will be triggered.

7.32.5 Pin access control

Access to device pins can be controlled by SPU. A pin can be declared as secure so that only secure peripherals or secure code can access it. Pins declared as non-secure can be accessed by both secure and non-secure peripherals or code.

The security attribute of each pin can be individually configured in SPU's GPIOPORT[n].PERM register. When the secure attribute is set for a pin, only peripherals that have the secure attribute set will be able to read the value of the pin or change it.

Peripherals can select the pins they need access to through their PSEL registers. If a peripheral has its attribute set to non-secure, but one of its PSEL registers selects a pin with the attribute set to secure, the SPU controlled logic will ensure that the pin selection is not propagated. In addition, the pin value will always be read as zero, to prevent a non-secure peripheral from obtaining a value from a secure pin. Access to other pins with attribute set as non-secure will not be blocked.

Pins can be assigned to other domains than the application domain by changing the MCUSEL value in the GPIO PIN_CNF[n] register. Domains that do not have a pin assigned to them cannot control a pin or read its status. Any pin configuration set in a domain that doesn't have ownership of that pin will not take effect until the MCUSEL is updated to assign that pin to the domain. Within each domain, pin access is controlled by that domain's local security configuration and peripheral PSEL registers. This is illustrated in the following figure:

The SPU setting will still count when the APP domain accesses its local GPIO peripheral, as local registers are still writable even though MCUSEL is set to a different domain. Any changes in the APP GPIO peripheral done to a GPIO controlled by another domain will not affect the GPIO pad until MCUSEL is changed to APP.

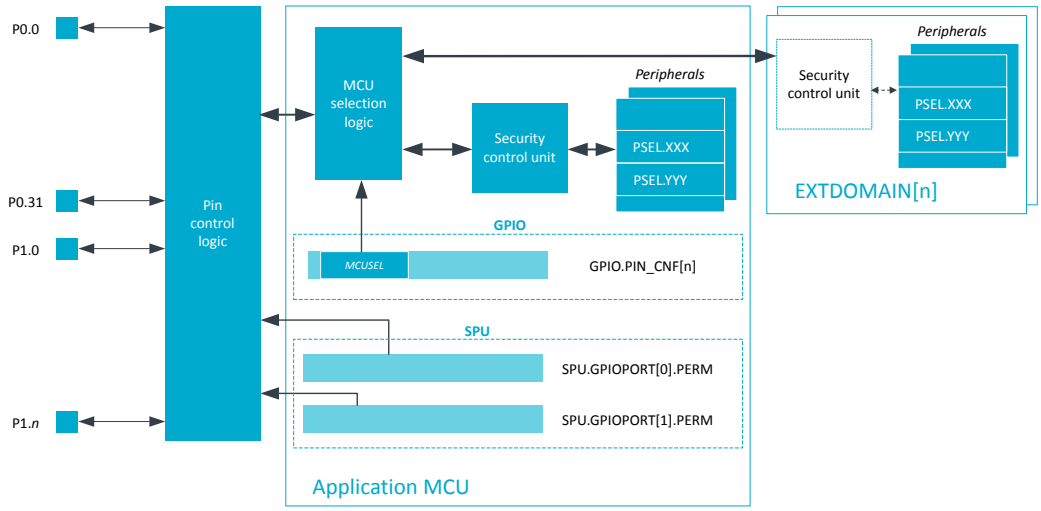


Figure 210: Pin access for domains other than the application domain

7.32.6 DPPI access control

Access to DPPI channels can be restricted. A channel can be declared as secure so that only secure peripherals can access it.

The security attribute of a DPPI channel is configured in DPPI[n].PERM (n=0..0) on page 604. When the secure attribute is set for a channel, only peripherals that have the secure attribute set will be able to publish events to this channel or subscribe to this channel to receive tasks.

The DPPI controller peripheral (DPPIC) is a split security peripheral, i.e., its security behavior depends on the security attributes of both the DPPIC and the accessing party. See Special considerations regarding the DPPIC configuration registers on page 595 for more information about the DPPIC security behavior.

If a non-secure peripheral wants to publish an event on a secure DPPI channel, the channel will ignore the event. If a non-secure peripheral subscribes to a secure DPPI channel, it will not receive any events from

this channel. The following figure illustrates the principle of operation of the security logic for a subscribed channel:

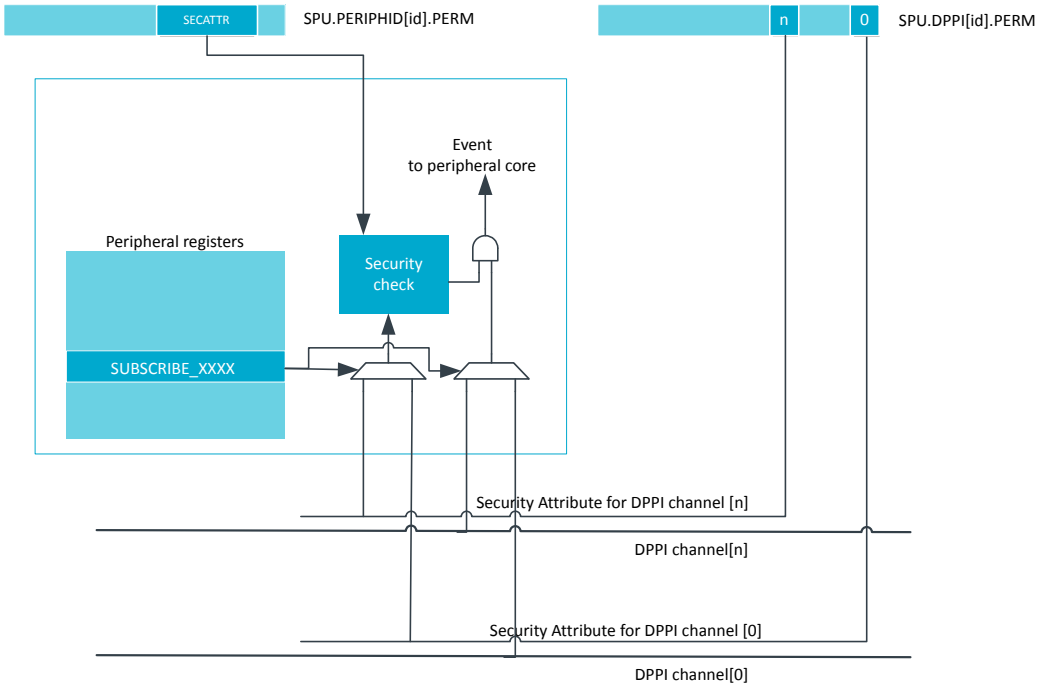


Figure 211: Subscribed channel security concept

No error reporting mechanism is associated with the DPPI access control logic.

7.32.6.1 Special considerations regarding the DPPIC configuration registers

DPPI channels can be enabled, disabled and grouped through the DPPI controller (DPPIC). The DPPIC is a split-security peripheral, and handles both secure and non-secure accesses.

A non-secure peripheral access will only be able to configure and control DPPI channels defined as non-secure in SPU's DPPI[n].PERM register(s). A secure peripheral access can control all DPPI channels, independently of the configuration in the DPPI[n].PERM register(s).

The DPPIC allows the creation of group of channels to be able to enable or disable all channels within a group simultaneously. The security attribute of a group of channels (secure or non-secure) is defined as follows:

- If all channels (enabled or not) in the group are non-secure, then the group is considered non-secure
- If at least one of the channels (enabled or not) in the group is secure, then the group is considered secure

A non-secure access to a DPPIC register, or a bitfield controlling a channel marked as secure in DPPI[n].PERM register(s), will be ignored:

- Write accesses will have no effect
- Read will always return a zero value

No exceptions are thrown when a non-secure access targets a register or bitfield controlling a secure channel. For example, if the bit *i* is set in the DPPI[n].PERM register (declaring the DPPI channel *i* as secure), then:

- Non-secure write accesses to registers CHEN, CHENSET and CHENCLR will not be able to write to bit *i* of those registers

- Non-secure write accesses to registers TASK_CHG[*j*].EN and TASK_CHG[*j*].DIS will be ignored if the channel group *j* contains at least one channel defined as secure (it can be the channel *i* itself or any channel declared as secured)
- Non-secure read accesses to registers CHEN, CHENSET and CHENCLR will always read zero for the bit at position *i*

For the channel configuration registers (DPPIC.CHG[n]), access from non-secure code is only possible if the included channels are all non-secure, whether the channels are enabled or not. If a DPPIC.CHG[*g*] register included one or more secure channels, then the group *g* is considered as secure and only a secure transfer can read or write DPPIC.CHG[*g*]. A non-secure write will be ignored and a non-secure read will return zero.

The DPPIC can subscribe to secure or non-secure channels through SUBSCRIBE_CHG[n] registers in order to trigger task for enabling or disabling groups of channels. But an event from a non-secure channel will be ignored if the group subscribing to this channel is secure. An event from a secure channel can trigger both secure and non-secure tasks.

7.32.7 External domain access control

Other domains with their own CPUs can access peripherals, flash and RAM memories. The SPU allows controlling accesses from those bus masters.

The external domains can access application MCU memories and peripherals. External domains are assigned security attributes as described in register EXTDOMAIN[n].PERM.

Domain	Capability register	Permission register
Network MCU	EXTDOMAIN[n].PERM (n=0..0) on page 603, SECUREMAPPING field	EXTDOMAIN[n].PERM (n=0..0) on page 603, SECATTR field

Table 156: Register mapping for external domains

The figure below illustrates how the security control units are used to assign security attributes to transfers initiated by the external domains:

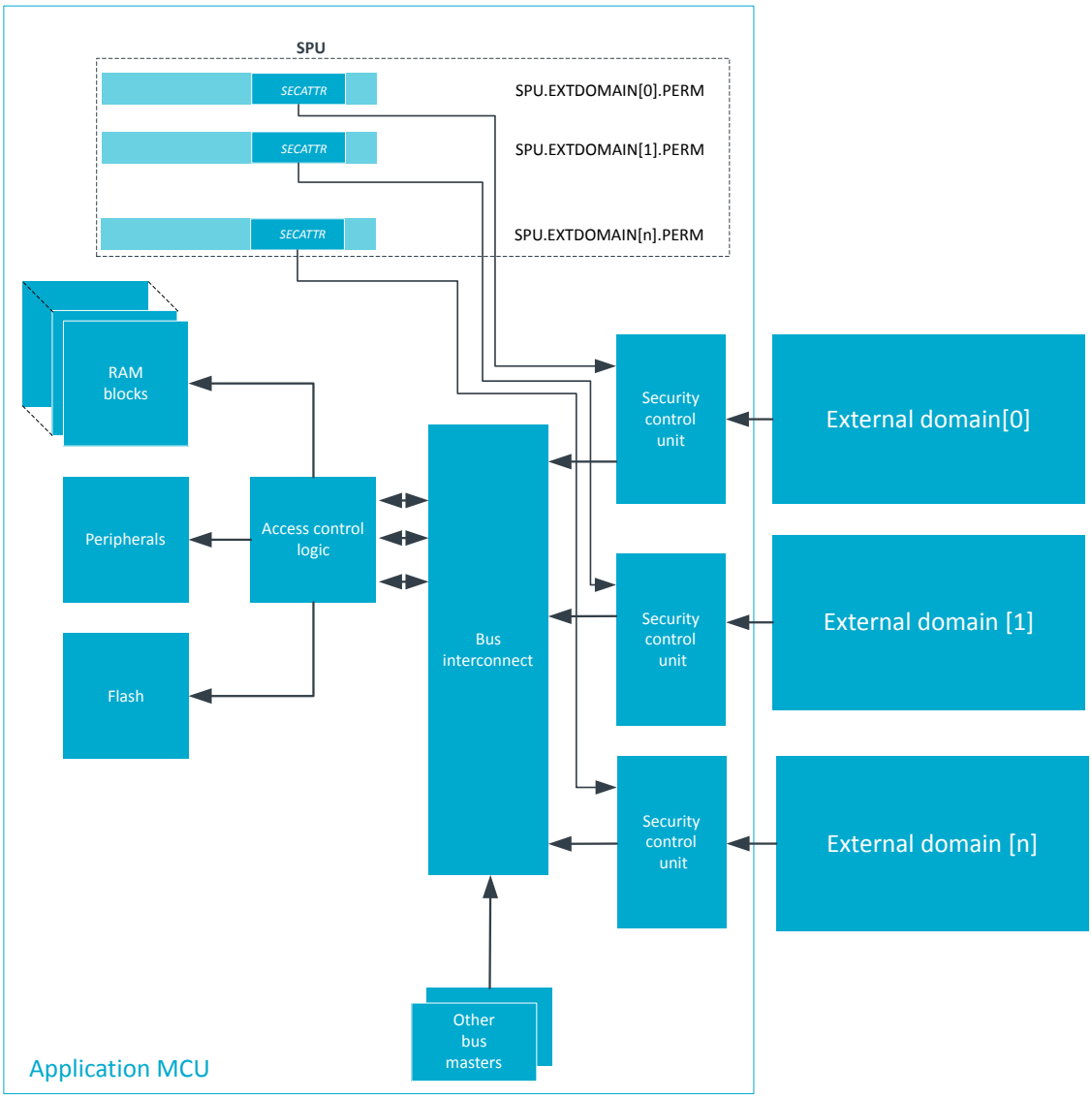


Figure 212: Access control from external domains

7.32.8 Arm TrustZone for Cortex-M ID allocation

Flash and RAM regions, as well as non-secure and secure peripherals, are assigned unique Arm TrustZone IDs.

The Arm TrustZone ID should not be mistaken for the peripheral ID used to identify peripherals.

The following table lists the Arm TrustZone ID allocation.

Regions	Arm TrustZone Cortex-M ID
Flash regions 0..63	0..63
RAM regions 0..63	64..127
UICR	252
FICR	252
CACHEDATA	252
CACHEINFO	252
Non-secure peripherals	253
Secure peripherals	254

Table 157: Arm TrustZone ID allocation

7.32.9 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50003000	APPLICATION	SPU	SPU	S	NA	System protection unit	

Table 158: Instances

Register	Offset	Security	Description	
EVENTS_RAMACCERR	0x100		A security violation has been detected for the RAM memory space	
EVENTS_FLASHACCERR	0x104		A security violation has been detected for the flash memory space	
EVENTS_PERIPHACCERR	0x108		A security violation has been detected on one or several peripherals	
PUBLISH_RAMACCERR	0x180		Publish configuration for event RAMACCERR	
PUBLISH_FLASHACCERR	0x184		Publish configuration for event FLASHACCERR	
PUBLISH_PERIPHACCERR	0x188		Publish configuration for event PERIPHACCERR	
INTEN	0x300		Enable or disable interrupt	
INTENSET	0x304		Enable interrupt	
INTENCLR	0x308		Disable interrupt	
CAP	0x400		Show implemented features for the current device	
CPULOCK	0x404		Configure bits to lock down CPU features at runtime	
EXTDOMAIN[n].PERM	0x440		Access for bus access generated from the external domain n List capabilities of the external domain n	
DPPI[n].PERM	0x480		Select between secure and non-secure attribute for the DPPI channels	
DPPI[n].LOCK	0x484		Prevent further modification of the corresponding PERM register	
GPIOPORT[n].PERM	0x4C0		Select between secure and non-secure attribute for pins 0 to 31 of port n	Retained
GPIOPORT[n].LOCK	0x4C4		Prevent further modification of the corresponding PERM register	
FLASHNSC[n].REGION	0x500		Define which flash region can contain the non-secure callable (NSC) region n	
FLASHNSC[n].SIZE	0x504		Define the size of the non-secure callable (NSC) region n	
RAMNSC[n].REGION	0x540		Define which RAM region can contain the non-secure callable (NSC) region n	
RAMNSC[n].SIZE	0x544		Define the size of the non-secure callable (NSC) region n	
FLASHREGION[n].PERM	0x600		Access permissions for flash region n	
RAMREGION[n].PERM	0x700		Access permissions for RAM region n	
PERIPHID[n].PERM	0x800		List capabilities and access permissions for the peripheral with ID n	

Table 159: Register overview

7.32.9.1 EVENTS_RAMACCERR

Address offset: 0x100

A security violation has been detected for the RAM memory space

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RAMACCERR			A security violation has been detected for the RAM memory space
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.32.9.2 EVENTS_FLASHACCERR

Address offset: 0x104

A security violation has been detected for the flash memory space

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_FLASHACCERR			A security violation has been detected for the flash memory space
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.32.9.3 EVENTS_PERIPHACCERR

Address offset: 0x108

A security violation has been detected on one or several peripherals

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_PERIPHACCERR			A security violation has been detected on one or several peripherals
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.32.9.4 PUBLISH_RAMACCERR

Address offset: 0x180

Publish configuration for event RAMACCERR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RAMACCERR will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.32.9.5 PUBLISH_FLASHACCERR

Address offset: 0x184

Publish configuration for event FLASHACCERR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event FLASHACCERR will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.32.9.6 PUBLISH_PERIPHACCERR

Address offset: 0x188

Publish configuration for event PERIPHACCERR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event PERIPHACCERR will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.32.9.7 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	RAMACCERR			Enable or disable interrupt for event RAMACCERR
			Disabled	0	Disable
			Enabled	1	Enable
B	RW	FLASHACCERR			Enable or disable interrupt for event FLASHACCERR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Disabled	0	Disable
			Enabled	1	Enable
C	RW	PERIPHACCERR			Enable or disable interrupt for event PERIPHACCERR
			Disabled	0	Disable
			Enabled	1	Enable

7.32.9.8 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	RAMACCERR			Write '1' to enable interrupt for event RAMACCERR
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	FLASHACCERR			Write '1' to enable interrupt for event FLASHACCERR
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	PERIPHACCERR			Write '1' to enable interrupt for event PERIPHACCERR
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.32.9.9 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	RAMACCERR			Write '1' to disable interrupt for event RAMACCERR
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	FLASHACCERR			Write '1' to disable interrupt for event FLASHACCERR
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	PERIPHACCERR			Write '1' to disable interrupt for event PERIPHACCERR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.32.9.10 CAP

Address offset: 0x400

Show implemented features for the current device

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**

ID	R/W	Field	Value ID	Value	Description
A	R	TZM			Show Arm TrustZone status
			NotAvailable	0	Arm TrustZone support not available
			Enabled	1	Arm TrustZone support is available

7.32.9.11 CPULOCK

Address offset: 0x404

Configure bits to lock down CPU features at runtime

Write '1' to any position to set the corresponding lock bit, which will remain set until the next reset

Any '0' writes to this register will be ignored

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LOCKSVTAIRCR			Write '1' to prevent updating the secure interrupt configuration until the next reset When set to '1', this lock bit prevents changes to: • The Secure vector table base address • Handling of Secure interrupt priority • BusFault, HardFault, and NMI security target
			Locked	1	Disables writes to the VTOR_S, AIRCR.PRIS, and AIRCR.BFHFNMINS registers
			Unlocked	0	These registers can be updated
B	RW	LOCKNSVTOR			Write '1' to prevent updating the non-secure vector table base address until the next reset When set to '1', this lock bit prevents changes to the Non-secure interrupt vector table base address register VTOR_NS
			Locked	1	The address of the non-secure vector table is locked
			Unlocked	0	The address of the non-secure vector table can be updated

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
C	RW	LOCKSMPU			Write '1' to prevent updating the secure MPU regions until the next reset When set to '1', this lock bit prevents changes to programmed Secure MPU memory regions and all writes to the registers are ignored
			Locked	1	Disables writes to the MPU_CTRL, MPU_RNR, MPU_RBAR, MPU_RLAR, MPU_RBAR_An and MPU_RLAR_An from software or from a debug agent connected to the processor in Secure state
			Unlocked	0	These registers can be updated
D	RW	LOCKNSMPU			Write '1' to prevent updating the Non-secure MPU regions until the next reset When set to '1', this lock bit prevents changes to programmed Non-secure MPU memory regions already programmed. All writes to the registers are ignored.
			Locked	1	Disables writes to the MPU_CTRL_NS, MPU_RNR_NS, MPU_RBAR_NS, MPU_RLAR_NS, MPU_RBAR_A_NSn and MPU_RLAR_A_NSn from software or from a debug agent connected to the processor
			Unlocked	0	These registers can be updated
E	RW	LOCKSAU			Write '1' to prevent updating the secure SAU regions until the next reset When set to '1', this lock bit prevents changes to Secure SAU memory regions already programmed. All writes to the registers are ignored.
			Locked	1	Disables writes to the SAU_CTRL, SAU_RNR, SAU_RBAR and SAU_RLAR registers from software or from a debug agent connected to the processor
			Unlocked	0	These registers can be updated

7.32.9.12 EXTDOMAIN[n].PERM (n=0..0)

Address offset: 0x440 + (n × 0x4)

Access for bus access generated from the external domain n

List capabilities of the external domain n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A
Reset 0x00000002				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0**

ID	R/W	Field	Value ID	Value	Description
A	R	SECUREMAPPING			Define configuration capabilities for TrustZone Cortex-M secure attribute
					This does not affect DPPI in the external domain
			NonSecure	0	The bus access from this external domain always have the non-secure attribute set
			Secure	1	The bus access from this external domain always have the secure attribute set
			UserSelectable	2	Non-secure or secure attribute for bus access from this domain is defined by the EXTDOMAIN[n].PERM register
B	RW	SECATTR			Peripheral security mapping
					This bit has effect only if EXTDOMAIN[n].PERM.SECUREMAPPING reads as UserSelectable
			NonSecure	0	Bus accesses from this domain have the non-secure attribute set
			Secure	1	Bus accesses from this domain have secure attribute set
C	RW	LOCK			
			Unlocked	0	This register can be updated
			Locked	1	The content of this register can't be changed until the next reset

7.32.9.13 DPPI[n].PERM (n=0..0)

Address offset: 0x480 + (n × 0x8)

Select between secure and non-secure attribute for the DPPI channels

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**

ID	R/W	Field	Value ID	Value	Description
A-f	RW	CHANNEL[i] (i=0..31)			Select secure attribute
			Secure	1	Channel i has its secure attribute set
			NonSecure	0	Channel i has its non-secure attribute set

7.32.9.14 DPPI[n].LOCK (n=0..0)

Address offset: 0x484 + (n × 0x8)

Prevent further modification of the corresponding PERM register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LOCK			
			Locked	1	DPPI[n].PERM register can't be changed until next reset
			Unlocked	0	DPPI[n].PERM register content can be changed

7.32.9.15 GPIOPORT[n].PERM (n=0..1) (Retained)

Address offset: 0x4C0 + (n × 0x8)

This register is a retained register

Select between secure and non-secure attribute for pins 0 to 31 of port n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A-f	RW	PIN[i] (i=0..31)			Select secure attribute attribute for PIN i.
			Secure	1	Pin i has its secure attribute set
			NonSecure	0	Pin i has its non-secure attribute set

7.32.9.16 GPIOPORT[n].LOCK (n=0..1)

Address offset: 0x4C4 + (n × 0x8)

Prevent further modification of the corresponding PERM register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LOCK			
			Locked	1	GPIOPORT[n].PERM register can't be changed until next reset
			Unlocked	0	GPIOPORT[n].PERM register content can be changed

7.32.9.17 FLASHNSC[n].REGION (n=0..1)

Address offset: 0x500 + (n × 0x8)

Define which flash region can contain the non-secure callable (NSC) region n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	REGION			Region number
B	RW	LOCK			
			Unlocked	0	This register can be updated
			Locked	1	The content of this register can't be changed until the next reset

7.32.9.18 FLASHNSC[n].SIZE (n=0..1)

Address offset: 0x504 + (n × 0x8)

Define the size of the non-secure callable (NSC) region n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	SIZE			Size of the non-secure callable (NSC) region n
			Disabled	0	The region n is not defined as a non-secure callable region. Normal security attributes (secure or non-secure) are enforced.
			32	1	The region n is defined as non-secure callable with size 32 bytes
			64	2	The region n is defined as non-secure callable with size 64 bytes
			128	3	The region n is defined as non-secure callable with size 128 bytes
			256	4	The region n is defined as non-secure callable with size 256 bytes
			512	5	The region n is defined as non-secure callable with size 512 bytes
			1024	6	The region n is defined as non-secure callable with size 1024 bytes
			2048	7	The region n is defined as non-secure callable with size 2048 bytes
			4096	8	The region n is defined as non-secure callable with size 4096 bytes
B	RW	LOCK			
			Unlocked	0	This register can be updated
			Locked	1	The content of this register can't be changed until the next reset

7.32.9.19 RAMNSC[n].REGION (n=0..1)

Address offset: 0x540 + (n × 0x8)

Define which RAM region can contain the non-secure callable (NSC) region n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	REGION			Region number
B	RW	LOCK			
			Unlocked	0	This register can be updated
			Locked	1	The content of this register can't be changed until the next reset

7.32.9.20 RAMNSC[n].SIZE (n=0..1)

Address offset: 0x544 + (n × 0x8)

Define the size of the non-secure callable (NSC) region n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	SIZE			Size of the non-secure callable (NSC) region n
			Disabled	0	The region n is not defined as a non-secure callable region. Normal security attributes (secure or non-secure) are enforced.
			32	1	The region n is defined as non-secure callable with size 32 bytes
			64	2	The region n is defined as non-secure callable with size 64 bytes
			128	3	The region n is defined as non-secure callable with size 128 bytes
			256	4	The region n is defined as non-secure callable with size 256 bytes
			512	5	The region n is defined as non-secure callable with size 512 bytes
			1024	6	The region n is defined as non-secure callable with size 1024 bytes
			2048	7	The region n is defined as non-secure callable with size 2048 bytes
			4096	8	The region n is defined as non-secure callable with size 4096 bytes
B	RW	LOCK			
			Unlocked	0	This register can be updated
			Locked	1	The content of this register can't be changed until the next reset

7.32.9.21 FLASHREGION[n].PERM (n=0..63)

Address offset: 0x600 + (n × 0x4)

Access permissions for flash region n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000017				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1**

ID	R/W	Field	Value ID	Value	Description
A	RW	EXECUTE			Configure instruction fetch permissions from flash region n
			Enable	1	Allow instruction fetches from flash region n
			Disable	0	Block instruction fetches from flash region n
B	RW	WRITE			Configure write permission for flash region n
			Enable	1	Allow write operation to region n
			Disable	0	Block write operation to region n
C	RW	READ			Configure read permissions for flash region n
			Enable	1	Allow read operation from flash region n
			Disable	0	Block read operation from flash region n
D	RW	SECATTR			Security attribute for flash region n
			Non_Secure	0	Flash region n security attribute is non-secure
			Secure	1	Flash region n security attribute is secure
E	RW	LOCK			
			Unlocked	0	This register can be updated

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000017				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1**
ID	R/W	Field	Value ID	Value	Description
			Locked	1	The content of this register can't be changed until the next reset

7.32.9.22 RAMREGION[n].PERM (n=0..63)

Address offset: 0x700 + (n × 0x4)

Access permissions for RAM region n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000017				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	EXECUTE			Configure instruction fetch permissions from RAM region n
			Enable	1	Allow instruction fetches from RAM region n
			Disable	0	Block instruction fetches from RAM region n
B	RW	WRITE			Configure write permission for RAM region n
			Enable	1	Allow write operation to RAM region n
			Disable	0	Block write operation to RAM region n
C	RW	READ			Configure read permissions for RAM region n
			Enable	1	Allow read operation from RAM region n
			Disable	0	Block read operation from RAM region n
D	RW	SECATTR			Security attribute for RAM region n
			Non_Secure	0	RAM region n security attribute is non-secure
			Secure	1	RAM region n security attribute is secure
E	RW	LOCK			
			Unlocked	0	This register can be updated
			Locked	1	The content of this register can't be changed until the next reset

7.32.9.23 PERIPHID[n].PERM (n=0..255)

Address offset: 0x800 + (n × 0x4)

List capabilities and access permissions for the peripheral with ID n

Note: Reset values are unique per peripheral instantation. Please refer to the peripheral instantiation table. Entries not listed in the instantiation table are undefined.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F E D C B B A A
Reset 0x00000012				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0**
ID	R/W	Field	Value ID	Value	Description
A	R	SECUREMAPPING			Define configuration capabilities for Arm TrustZone Cortex-M secure attribute
			NonSecure	0	This peripheral is always accessible as a non-secure peripheral
			Secure	1	This peripheral is always accessible as a secure peripheral
			UserSelectable	2	Non-secure or secure attribute for this peripheral is defined by the PERIPHID[n].PERM register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F E D C B B A A
Reset 0x00000012				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0**
ID	R/W	Field	Value ID	Value	Description
			Split	3	This peripheral implements the split security mechanism. Non-secure or secure attribute for this peripheral is defined by the PERIPHID[n].PERM register.
B	R	DMA			Indicates if the peripheral has DMA capabilities and if DMA transfer can be assigned to a different security attribute than the peripheral itself
			NoDMA	0	Peripheral has no DMA capability
			NoSeparateAttribute	1	Peripheral has DMA and DMA transfers always have the same security attribute as assigned to the peripheral
			SeparateAttribute	2	Peripheral has DMA and DMA transfers can have a different security attribute than the one assigned to the peripheral
C	RW	SECATTR			Peripheral security mapping This bit has effect only if PERIPHID[n].PERM.SECUREMAPPING reads as UserSelectable or Split
			Secure	1	Peripheral is mapped in secure peripheral address space
			NonSecure	0	If SECUREMAPPING == UserSelectable: Peripheral is mapped in non-secure peripheral address space. If SECUREMAPPING == Split: Peripheral is mapped in non-secure and secure peripheral address space.
D	RW	DMASEC			Security attribution for the DMA transfer This bit has effect only if PERIPHID[n].PERM.SECATTR is set to secure
			Secure	1	DMA transfers initiated by this peripheral have the secure attribute set
			NonSecure	0	DMA transfers initiated by this peripheral have the non-secure attribute set
E	RW	LOCK			
			Unlocked	0	This register can be updated
			Locked	1	The content of this register can't be changed until the next reset
F	R	PRESENT			Indicate if a peripheral is present with ID n
			NotPresent	0	Peripheral is not present
			IsPresent	1	Peripheral is present

7.33 SWI — Software interrupts

A set of interrupts have been reserved for use as software interrupts.

7.33.1 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x4101A000	NETWORK	SWI	SWI0	NS	NA	Software interrupt 0	
0x4101B000	NETWORK	SWI	SWI1	NS	NA	Software interrupt 1	
0x4101C000	NETWORK	SWI	SWI2	NS	NA	Software interrupt 2	
0x4101D000	NETWORK	SWI	SWI3	NS	NA	Software interrupt 3	

Table 160: Instances

7.34 TEMP — Temperature sensor

The temperature sensor measures die temperature over the temperature range of the device. Linearity compensation can be implemented if required by the application.

The main features of TEMP are the following:

- Temperature range is greater than or equal to operating temperature of the device
- Resolution is 0.25 °C

TEMP is started by triggering the START task.

When the temperature measurement is completed, a DATARDY event will be generated and the result of the measurement can be read from the TEMP register.

To achieve the measurement accuracy stated in the electrical specification, the crystal oscillator must be selected as the HFCLK source, see CLOCK — Clock control on page 69 for more information.

When the temperature measurement is completed, TEMP analog electronics power down to save power.

TEMP only supports one-shot operation, meaning that every TEMP measurement has to be explicitly started using the START task.

7.34.1 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x41010000	NETWORK	TEMP	TEMP	NS	NA	Temperature sensor	

Table 161: Instances

Register	Offset	Security	Description
TASKS_START	0x000		Start temperature measurement
TASKS_STOP	0x004		Stop temperature measurement
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
EVENTS_DATARDY	0x100		Temperature measurement complete, data ready
PUBLISH_DATARDY	0x180		Publish configuration for event DATARDY
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
TEMP	0x508		Temperature in °C (0.25° steps)
A0	0x520		Slope of first piecewise linear function
A1	0x524		Slope of second piecewise linear function
A2	0x528		Slope of third piecewise linear function
A3	0x52C		Slope of fourth piecewise linear function
A4	0x530		Slope of fifth piecewise linear function

Register	Offset	Security	Description
A5	0x534		Slope of sixth piecewise linear function
B0	0x540		y-intercept of first piecewise linear function
B1	0x544		y-intercept of second piecewise linear function
B2	0x548		y-intercept of third piecewise linear function
B3	0x54C		y-intercept of fourth piecewise linear function
B4	0x550		y-intercept of fifth piecewise linear function
B5	0x554		y-intercept of sixth piecewise linear function
T0	0x560		Endpoint of first piecewise linear function
T1	0x564		Endpoint of second piecewise linear function
T2	0x568		Endpoint of third piecewise linear function
T3	0x56C		Endpoint of fourth piecewise linear function
T4	0x570		Endpoint of fifth piecewise linear function

Table 162: Register overview

7.34.1.1 TASKS_START

Address offset: 0x000

Start temperature measurement

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_START			Start temperature measurement
			Trigger	1	Trigger task

7.34.1.2 TASKS_STOP

Address offset: 0x004

Stop temperature measurement

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Stop temperature measurement
			Trigger	1	Trigger task

7.34.1.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task START

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task START will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.34.1.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.34.1.5 EVENTS_DATARDY

Address offset: 0x100

Temperature measurement complete, data ready

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_DATARDY			Temperature measurement complete, data ready
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.34.1.6 PUBLISH_DATARDY

Address offset: 0x180

Publish configuration for event DATARDY

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event DATARDY will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.34.1.7 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	DATARDY			Write '1' to enable interrupt for event DATARDY
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.34.1.8 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	DATARDY			Write '1' to disable interrupt for event DATARDY
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.34.1.9 TEMP

Address offset: 0x508

Temperature in °C (0.25° steps)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	TEMP			Temperature in °C (0.25° steps) Result of temperature measurement. Die temperature in °C, 2's complement format, 0.25 °C steps. Decision point: DATARDY

7.34.1.10 A0

Address offset: 0x520

Slope of first piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x000002D9				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 1 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	A0			Slope of first piecewise linear function

7.34.1.11 A1

Address offset: 0x524

Slope of second piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000322				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 0 1 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	A1			Slope of second piecewise linear function

7.34.1.12 A2

Address offset: 0x528

Slope of third piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000355				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 1 0 1 0 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	A2			Slope of third piecewise linear function

7.34.1.13 A3

Address offset: 0x52C

Slope of fourth piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x000003DF				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	A3			Slope of fourth piecewise linear function

7.34.1.14 A4

Address offset: 0x530

Slope of fifth piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x0000044E				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 1 1 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	A4			Slope of fifth piecewise linear function

7.34.1.15 A5

Address offset: 0x534

Slope of sixth piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A
Reset 0x000004B7				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 1 0 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	A5			Slope of sixth piecewise linear function

7.34.1.16 B0

Address offset: 0x540

y-intercept of first piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A
Reset 0x00000FC7				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	B0			y-intercept of first piecewise linear function

7.34.1.17 B1

Address offset: 0x544

y-intercept of second piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A
Reset 0x00000F71				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 1 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	B1			y-intercept of second piecewise linear function

7.34.1.18 B2

Address offset: 0x548

y-intercept of third piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A
Reset 0x00000F6C				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 0 1 1 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	B2			y-intercept of third piecewise linear function

7.34.1.19 B3

Address offset: 0x54C

y-intercept of fourth piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A
Reset 0x00000FCB				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 1 0 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	B3			y-intercept of fourth piecewise linear function

7.34.1.20 B4

Address offset: 0x550

y-intercept of fifth piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A
Reset 0x0000004B				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	B4			y-intercept of fifth piecewise linear function

7.34.1.21 B5

Address offset: 0x554

y-intercept of sixth piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A
Reset 0x000000F6				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	B5			y-intercept of sixth piecewise linear function

7.34.1.22 T0

Address offset: 0x560

Endpoint of first piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A
Reset 0x000000E1				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	T0			Endpoint of first piecewise linear function

7.34.1.23 T1

Address offset: 0x564

Endpoint of second piecewise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A
Reset 0x000000F9				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	T1			Endpoint of second piecewise linear function

7.34.1.24 T2

Address offset: 0x568

Endpoint of third piecewise linear function

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A A A A A A A A
Reset 0x00000010						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description	
A	RW	T2			Endpoint of third piecewise linear function	

7.34.1.25 T3

Address offset: 0x56C

Endpoint of fourth piecewise linear function

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A A A A A A A A
Reset 0x00000026						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0**
ID	R/W	Field	Value ID	Value	Description	
A	RW	T3			Endpoint of fourth piecewise linear function	

7.34.1.26 T4

Address offset: 0x570

Endpoint of fifth piecewise linear function

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A A A A A A A A
Reset 0x0000003F						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description	
A	RW	T4			Endpoint of fifth piecewise linear function	

7.34.2 Electrical specification

7.34.2.1 Temperature Sensor Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
t_{TEMP}	Time required for temperature measurement		36		µs
$T_{TEMP,RANGE}$	Temperature sensor range	-20		70	°C
$T_{TEMP,ACC}$	Temperature sensor accuracy	-5		5	°C
$T_{TEMP,RANGE,EXT}$	Temperature sensor range, extended temperature range	-40		105	°C
$T_{TEMP,ACC,EXT}$	Temperature sensor accuracy, extended temperature range	-7		7	°C
$T_{TEMP,RES}$	Temperature sensor resolution		0.25		°C
$T_{TEMP,STB}$	Sample to sample stability at constant device temperature			±0.25	°C
$T_{TEMP,OFFST}$	Sample offset at 25°C	-2.5		2.5	°C

7.35 TIMER — Timer/counter

This peripheral is a general purpose timer allowing time intervals to be defined by user input. It can operate in two modes: Timer mode and Counter mode.

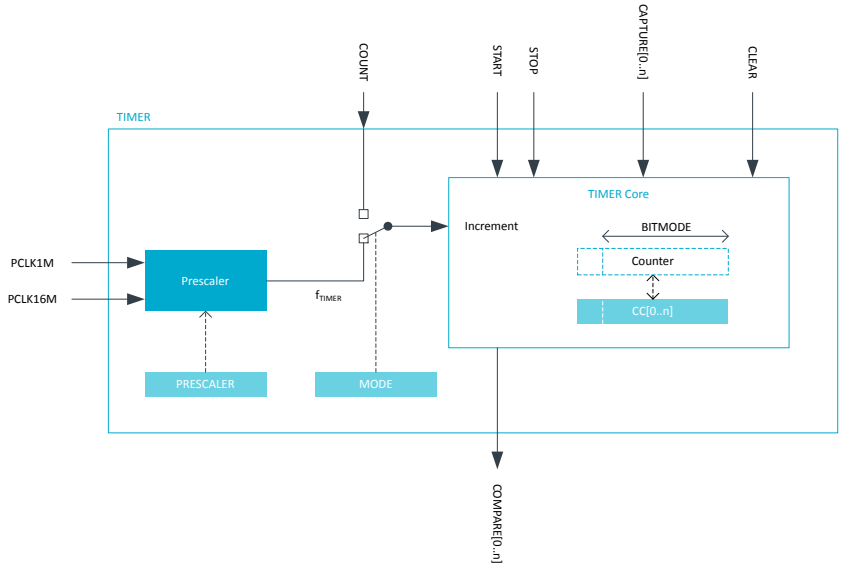


Figure 213: Block schematic for timer/counter

TIMER runs on the high-frequency clock source (HFCLK) and includes a four-bit (1/2X) prescaler that can divide the timer input clock from the HFCLK controller. Clock source selection between PCLK16M and PCLK1M is automatic according to the TIMER base frequency set by the prescaler. The TIMER base frequency is always given as 16 MHz divided by the prescaler value.

The PPI system allows a TIMER event to trigger a task on another system peripheral on the device. The PPI system also enables the TIMER task/event feature to generate periodic output and PWM signals to any GPIO. The number of GPIO inputs or outputs used at the same time is limited by the number of GPIOTE channels.

TIMER can operate in two modes: Timer mode and Counter mode. In both modes, TIMER is started by triggering the START task, and stopped by triggering the STOP task. After TIMER stops, it can resume timing/counting by triggering the START task again. When timing/counting resumes, TIMER continues from the value it was on prior to stopping.

In Timer mode, TIMER's internal Counter register is incremented by one for every tick of the timer frequency f_{TIMER}, as illustrated in Block schematic for timer/counter on page 618. The timer frequency is derived from PCLK16M as shown in the following example, using the values specified in the PRESCALER register.

$$f_{TIMER} = 16\ MHz\ /\ (2^{PRESCALER})$$

When $f_{TIMER} \leq 1$ MHz, TIMER uses PCLK1M instead of PCLK16M for reduced power consumption.

In Counter mode, the TIMER's internal Counter register is incremented by one each time the COUNT task is triggered, meaning the timer frequency and the prescaler are not utilized in Counter mode. Similarly, the COUNT task has no effect in Timer mode.

TIMER's maximum value is configured by changing the bit-width of the timer in register BITMODE on page 626.

PRESCALER on page 626 and BITMODE on page 626 must only be updated when TIMER is stopped. If these registers are updated while TIMER is started, unpredictable behavior may occur.

When TIMER is incremented beyond its maximum value, the Counter register will overflow and TIMER will automatically start over from zero.

The Counter register can be cleared by triggering the CLEAR task. This will explicitly set the internal value to zero.

TIMER implements multiple capture/compare registers.

Independent of prescaler settings, the accuracy of TIMER is equivalent to one tick of the timer frequency f_{TIMER} as illustrated in Block schematic for timer/counter on page 618.

7.35.1 Capture

TIMER implements one capture task for every available capture/compare register.

Every time the CAPTURE[n] task is triggered, the counter value is copied to the CC[n] register.

7.35.2 Compare

TIMER implements one COMPARE event for every available capture/compare register.

When the counter value becomes equal to the value specified in a capture compare register CC[n], the corresponding compare event COMPARE[n] is generated.

BITMODE on page 626 specifies how many Counter and capture/compare register bits are used when the comparison is performed. Other bits are ignored.

The COMPARE event can be configured to operate in one-shot mode by configuring the corresponding ONESHOTEN[n] register. After writing CC[n], a COMPARE[n] event is generated the first time the Counter matches CC[n].

7.35.3 Task delays

After TIMER is started, the CLEAR, COUNT, and STOP tasks are guaranteed to take effect within one clock cycle of the PCLK16M.

7.35.4 Task priority

If the START task and the STOP task are triggered at the same time, meaning within the same period of PCLK16M, the STOP task is prioritized.

If one or more of the CAPTURE tasks and the CLEAR task are triggered at the same time, that is, within the same period of PCLK16M, the CAPTURE tasks are prioritized. This means that the CC registers will capture the counter value before the CLEAR tasks are triggered.

7.35.5 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x5000F000 0x4000F000	APPLICATION	TIMER	TIMER0 : S TIMER0 : NS	US	NA	Timer 0	6 capture compare channels implemented
0x50010000 0x40010000	APPLICATION	TIMER	TIMER1 : S TIMER1 : NS	US	NA	Timer 1	6 capture compare channels implemented
0x50011000 0x40011000	APPLICATION	TIMER	TIMER2 : S TIMER2 : NS	US	NA	Timer 2	6 capture compare channels implemented
0x4100C000	NETWORK	TIMER	TIMER0	NS	NA	Timer 0	
0x41018000	NETWORK	TIMER	TIMER1	NS	NA	Timer 1	
0x41019000	NETWORK	TIMER	TIMER2	NS	NA	Timer 2	

Table 163: Instances

Register	Offset	Security	Description	
TASKS_START	0x000		Start Timer	
TASKS_STOP	0x004		Stop Timer	
TASKS_COUNT	0x008		Increment Timer (Counter mode only)	
TASKS_CLEAR	0x00C		Clear time	
TASKS_SHUTDOWN	0x010		Shut down timer	Deprecated
TASKS_CAPTURE[n]	0x040		Capture Timer value to CC[n] register	
SUBSCRIBE_START	0x080		Subscribe configuration for task START	
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP	
SUBSCRIBE_COUNT	0x088		Subscribe configuration for task COUNT	
SUBSCRIBE_CLEAR	0x08C		Subscribe configuration for task CLEAR	
SUBSCRIBE_SHUTDOWN	0x090		Subscribe configuration for task SHUTDOWN	Deprecated
SUBSCRIBE_CAPTURE[n]	0x0C0		Subscribe configuration for task CAPTURE[n]	
EVENTS_COMPARE[n]	0x140		Compare event on CC[n] match	
PUBLISH_COMPARE[n]	0x1C0		Publish configuration for event COMPARE[n]	
SHORTS	0x200		Shortcuts between local events and tasks	
INTEN	0x300		Enable or disable interrupt	
INTENSET	0x304		Enable interrupt	
INTENCLR	0x308		Disable interrupt	
MODE	0x504		Timer mode selection	
BITMODE	0x508		Configure the number of bits used by the TIMER	
PRESCALER	0x510		Timer prescaler register	
CC[n]	0x540		Capture/Compare register n	
ONESHOTEN[n]	0x580		Enable one-shot operation for Capture/Compare channel n	

Table 164: Register overview

7.35.5.1 TASKS_START

Address offset: 0x000

Start Timer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_START			Start Timer
			Trigger	1	Trigger task

7.35.5.2 TASKS_STOP

Address offset: 0x004

Stop Timer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Stop Timer
			Trigger	1	Trigger task

7.35.5.3 TASKS_COUNT

Address offset: 0x008

Increment Timer (Counter mode only)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_COUNT			Increment Timer (Counter mode only)
			Trigger	1	Trigger task

7.35.5.4 TASKS_CLEAR

Address offset: 0x00C

Clear time

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_CLEAR			Clear time
			Trigger	1	Trigger task

7.35.5.5 TASKS_SHUTDOWN (Deprecated)

Address offset: 0x010

Shut down timer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_SHUTDOWN			Shut down timer Deprecated
			Trigger	1	Trigger task

7.35.5.6 TASKS_CAPTURE[n] (n=0..7)

Address offset: 0x040 + (n × 0x4)

Capture Timer value to CC[n] register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_CAPTURE			Capture Timer value to CC[n] register
			Trigger	1	Trigger task

7.35.5.7 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task START

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task START will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.35.5.8 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.35.5.9 SUBSCRIBE_COUNT

Address offset: 0x088

Subscribe configuration for task COUNT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task COUNT will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.35.5.10 SUBSCRIBE_CLEAR

Address offset: 0x08C

Subscribe configuration for task CLEAR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task CLEAR will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.35.5.11 SUBSCRIBE_SHUTDOWN (Deprecated)

Address offset: 0x090

Subscribe configuration for task SHUTDOWN

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task SHUTDOWN will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.35.5.12 SUBSCRIBE_CAPTURE[n] (n=0..7)

Address offset: 0x0C0 + (n × 0x4)

Subscribe configuration for task CAPTURE[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task CAPTURE[n] will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.35.5.13 EVENTS_COMPARE[n] (n=0..7)

Address offset: 0x140 + (n × 0x4)

Compare event on CC[n] match

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_COMPARE			Compare event on CC[n] match
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.35.5.14 PUBLISH_COMPARE[n] (n=0..7)

Address offset: 0x1C0 + (n × 0x4)

Publish configuration for event COMPARE[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event COMPARE[n] will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.35.5.15 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A-H	RW	COMPARE[i]_CLEAR (i=0..7)			Shortcut between event COMPARE[i] and task CLEAR
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
I-P	RW	COMPARE[i]_STOP (i=0..7)			Shortcut between event COMPARE[i] and task STOP
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.35.5.16 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-H	RW	COMPARE[i] (i=0..7)			Enable or disable interrupt for event COMPARE[i]
			Disabled	0	Disable
			Enabled	1	Enable

7.35.5.17 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-H	RW	COMPARE[i] (i=0..7)			Write '1' to enable interrupt for event COMPARE[i]
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.35.5.18 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-H	RW	COMPARE[i] (i=0..7)			Write '1' to disable interrupt for event COMPARE[i]
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.35.5.19 MODE

Address offset: 0x504

Timer mode selection

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MODE			Timer mode
			Timer	0	Select Timer mode
			Counter	1	Select Counter mode
			LowPowerCounter	2	Select Low Power Counter mode

Deprecated

7.35.5.20 BITMODE

Address offset: 0x508

Configure the number of bits used by the TIMER

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	BITMODE			Timer bit width
			16Bit	0	16 bit timer bit width
			08Bit	1	8 bit timer bit width
			24Bit	2	24 bit timer bit width
			32Bit	3	32 bit timer bit width

7.35.5.21 PRESCALER

Address offset: 0x510

Timer prescaler register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000004				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PRESCALER		[0..9]	Prescaler value

7.35.5.22 CC[n] (n=0..7)

Address offset: 0x540 + (n × 0x4)

Capture/Compare register n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CC			Capture/Compare value Only the number of bits indicated by BITMODE will be used by the TIMER.

7.35.5.23 ONESHOTEN[n] (n=0..7)

Address offset: 0x580 + (n × 0x4)

Enable one-shot operation for Capture/Compare channel n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ONESHOTEN			Enable one-shot operation
					Configures the corresponding compare-channel for one-shot operation
			Disable	0	Disable one-shot operation
					Compare event is generated every time the Counter matches CC[n]
			Enable	1	Enable one-shot operation
					Compare event is generated the first time the Counter matches CC[n] after CC[n] has been written

7.35.6 Electrical specification

7.36 TWIM — I^2C compatible two-wire interface master with EasyDMA

TWI master with EasyDMA (TWIM) is a two-wire half-duplex master which can communicate with multiple slave devices connected to the same bus.

Listed here are the main features for TWIM:

- I^2C compatible
- Supported baud rates: 100, 250, 400 and 1000 kbps
- Support for clock stretching (non I^2C compliant)
- EasyDMA

The two-wire interface can communicate with a bi-directional wired-AND bus with two lines (SCL, SDA). The protocol makes it possible to interconnect up to 127 individually addressable devices. TWIM is not compatible with CBUS.

The GPIOs used for each two-wire interface line can be chosen from any GPIO on the device and are independently configurable. This enables great flexibility in device pinout and efficient use of board space and signal routing.

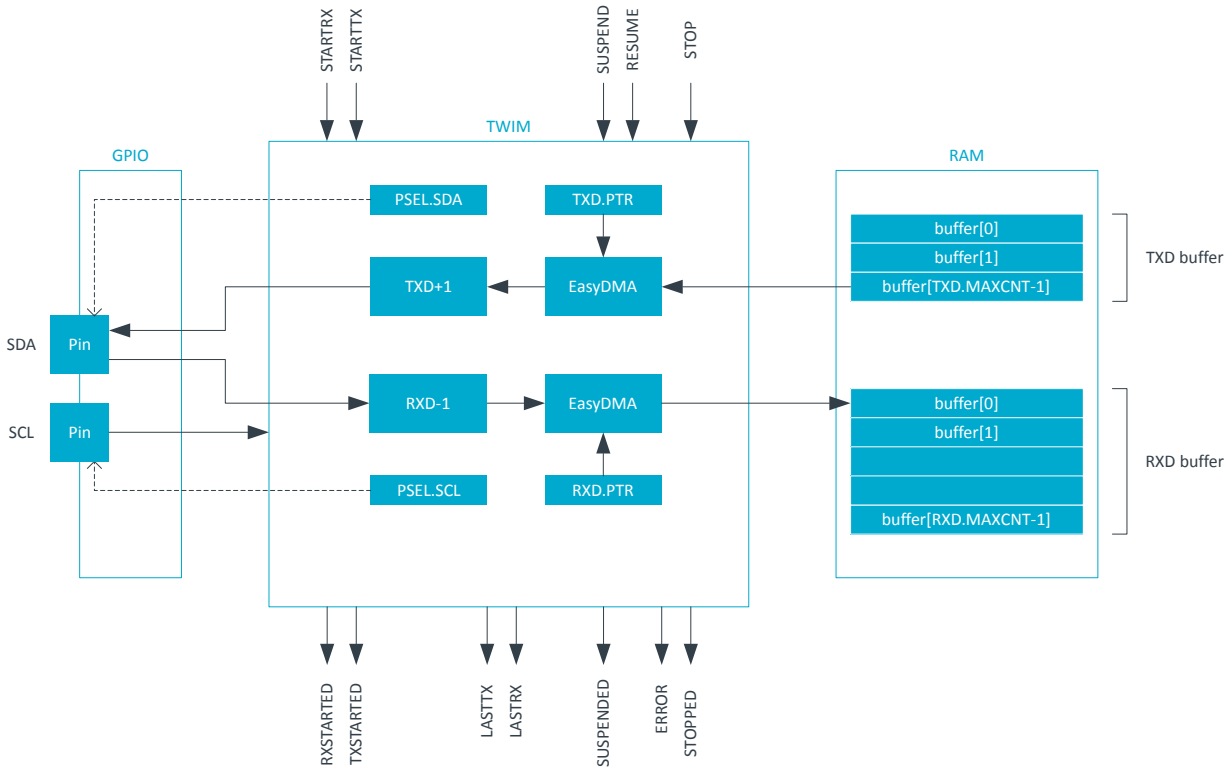


Figure 214: TWI master with EasyDMA

A typical TWI setup consists of one master and one or more slaves, as illustrated in the following figure. TWIM is only able to operate as a single master on the TWI bus. Multi-master bus configuration is not supported.

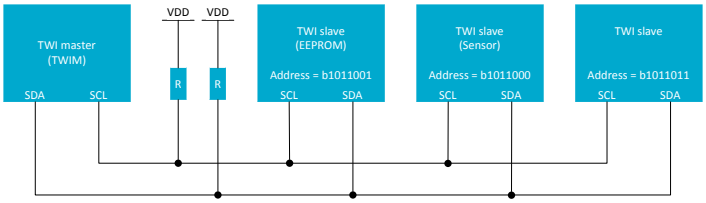


Figure 215: A typical TWI setup comprising one master and three slaves

TWIM supports clock stretching performed by the slaves. The SCK pulse following a stretched clock cycle may be shorter than specified by the I^2C specification.

TWIM is started by triggering the STARTTX or STARTRX tasks, and stopped by triggering the STOP task. After a STOP task, TWIM generates a STOPPED event when it has stopped.

After TWIM has been started, the STARTTX or STARTRX tasks should not be triggered again until TWIM has issued a LASTRX, LASTTX, or STOPPED event.

TWIM can be suspended using the SUSPEND task, such as when using the TWI master in a low priority interrupt context. When TWIM enters suspend state, it will automatically issue a SUSPENDED event while performing a continuous clock stretching until it is instructed to resume operation via a RESUME task. TWIM cannot be stopped while it is suspended, thus the STOP task has to be issued after the TWI master has been resumed.

Note: Any ongoing byte transfer will be allowed to complete before the suspend is enforced. A SUSPEND task has no effect unless TWIM is actively involved in a transfer.

If a NACK is clocked in from the slave, TWIM generates an ERROR event.

7.36.1 Shared resources

TWIM shares registers and other resources with other peripherals that have the same ID as the TWI master. Therefore, you must disable all peripherals that have the same ID as TWIM before it can be configured and used.

Disabling a peripheral that has the same ID as TWIM will not reset any of the registers that are shared with TWIM. It is therefore important to configure all relevant registers explicitly to secure that TWIM operates correctly.

The Instantiation table in Peripherals on page 146 shows which peripherals have the same ID as the TWI.

7.36.2 EasyDMA

TWIM implements EasyDMA for accessing RAM without CPU involvement.

TWIM implements the EasyDMA channels found in the following table.

Channel	Type	Register Cluster
TXD	READER	TXD
RXD	WRITER	RXD

Table 165: TWIM EasyDMA Channels

For detailed information regarding the use of EasyDMA, see EasyDMA on page 150.

The RXD.PTR, TXD.PTR, RXD.MAXCNT, and TXD.MAXCNT registers are double-buffered. They can be updated and prepared for the next RX or TX transmission immediately after having received the RXSTARTED or TXSTARTED event.

The STOPPED event indicates that EasyDMA has finished accessing the buffer in RAM.

7.36.3 Master write sequence

A TWIM write sequence is started by triggering the STARTTX task. After the STARTTX task has been triggered, TWIM generates a start condition on the TWI bus, followed by clocking out the address and the READ/WRITE bit set to 0 (WRITE=0, READ=1).

The address must match the address of the slave device that the master wants to write to. The READ/WRITE bit is followed by an ACK/NACK bit (ACK=0 or NACK=1) generated by the slave.

After receiving the ACK bit, TWIM clocks out the data bytes found in the transmit buffer located in RAM at the address specified in the TXD.PTR register. Each byte clocked out from TWIM will be followed by an ACK/NACK bit clocked in from the slave.

A typical TWIM write sequence is shown in the following figure, including clock stretching performed by TWIM following a SUSPEND task.

A SUSPENDED event indicates that the SUSPEND task has taken effect. This event can be used to synchronize the software.

TWIM will generate a LASTTX event when it starts to transmit the last byte.

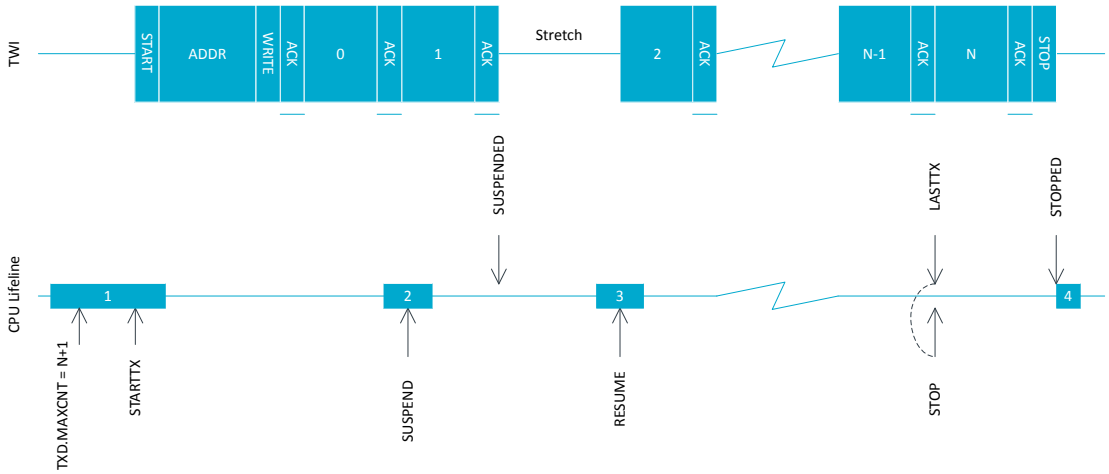


Figure 216: TWIM writing data to a slave

TWIM is stopped by triggering the STOP task. This task should be triggered during the transmission of the last byte to secure that TWIM will stop as fast as possible after sending the last byte. The shortcut between LASTTX and STOP can alternatively be used to accomplish this.

Note: TWIM does not stop by itself when the entire RAM buffer has been sent, or when an error occurs. The STOP task must be issued, through the use of a local or PPI shortcut, or in software as part of the error handler.

7.36.4 Master read sequence

A TWIM read sequence is started by triggering the STARTRX task. After the STARTRX task has been triggered, TWIM generates a start condition on the TWI bus, followed by clocking out the address and the READ/WRITE bit set to 1 (WRITE = 0, READ = 1). The address must match the address of the slave device that the master wants to read from. The READ/WRITE bit is followed by an ACK/NACK bit (ACK=0 or NACK = 1) generated by the slave.

After sending the ACK bit, the TWI slave sends data to the master using the clock generated by TWIM.

Data received will be stored in RAM at the address specified in the RXD.PTR register. The TWI master will generate an ACK after all but the last byte have been received from the slave. TWIM generates a NACK after the last byte received to indicate that the read sequence shall stop.

A typical TWIM read sequence is illustrated in the following figure, including clock stretching performed by TWIM following a SUSPEND task.

A SUSPENDED event indicates that the SUSPEND task has taken effect. This event can be used to synchronize the software.

TWIM generates a LASTRX event when it is ready to receive the last byte. If RXD.MAXCNT > 1, the LASTRX event is generated after sending the ACK of the previously received byte. If RXD.MAXCNT = 1, the LASTRX event is generated after receiving the ACK following the address and READ bit.

TWIM is stopped by triggering the STOP task. This task must be triggered before the NACK bit is supposed to be transmitted. The STOP task can be triggered at any time during the reception of the last byte. It is recommended to use the shortcut between LASTRX and STOP to accomplish this.

TWIM does not stop by itself when the RAM buffer is full, or when an error occurs. The STOP task must be issued, through the use of a local or PPI shortcut, or in software as part of the error handler.

TWIM cannot be stopped while suspended, so the STOP task must be issued after TWIM has been resumed.

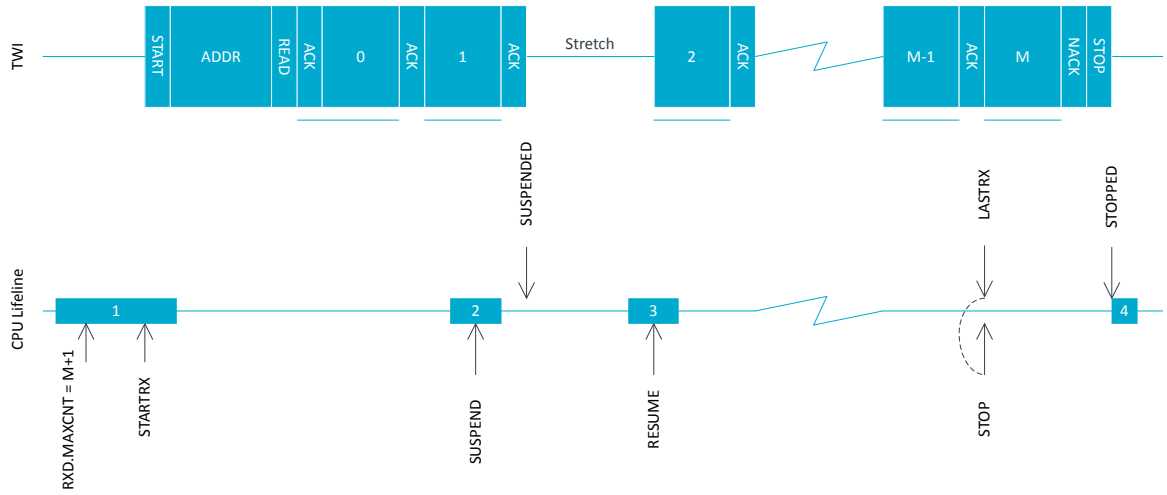


Figure 217: TWIM reading data from a slave

7.36.5 Master repeated start sequence

A typical repeated start sequence is when TWIM writes two bytes to the slave followed by reading four bytes from the slave. This example uses shortcuts to perform the simplest type of repeated start sequence, i.e. one write followed by one read. The same approach can be used to perform a repeated start sequence where the sequence is read followed by write.

The following figure shows an example of a repeated start sequence where TWIM writes two bytes followed by reading four bytes from the slave.

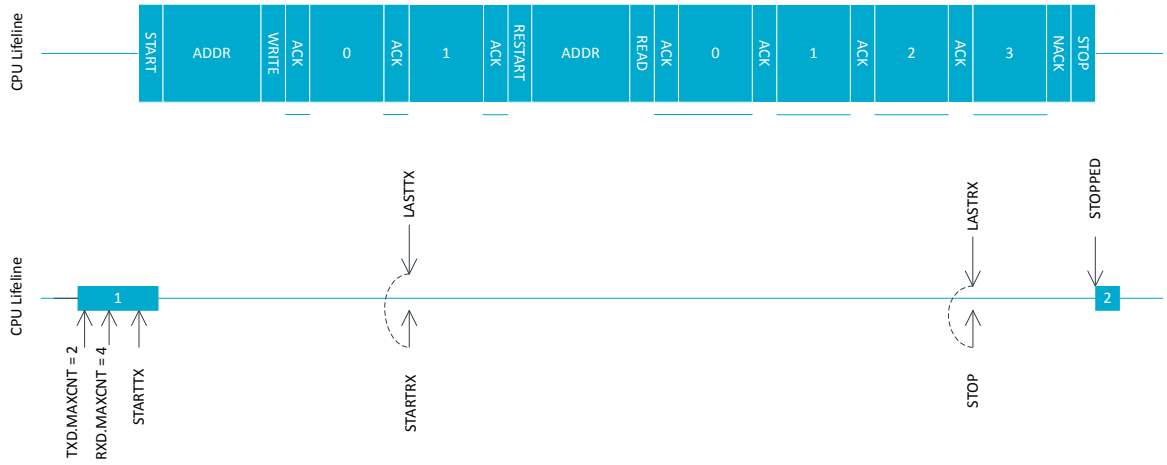


Figure 218: Master repeated start sequence

If a more complex repeated start sequence is needed, and the TWI firmware drive is serviced in a low priority interrupt, it may be necessary to use the SUSPEND task and SUSPENDED event to guarantee that the correct tasks are generated at the correct time. A double repeated start sequence using the SUSPEND task to secure safe operation in low priority interrupts is shown in the following figure.

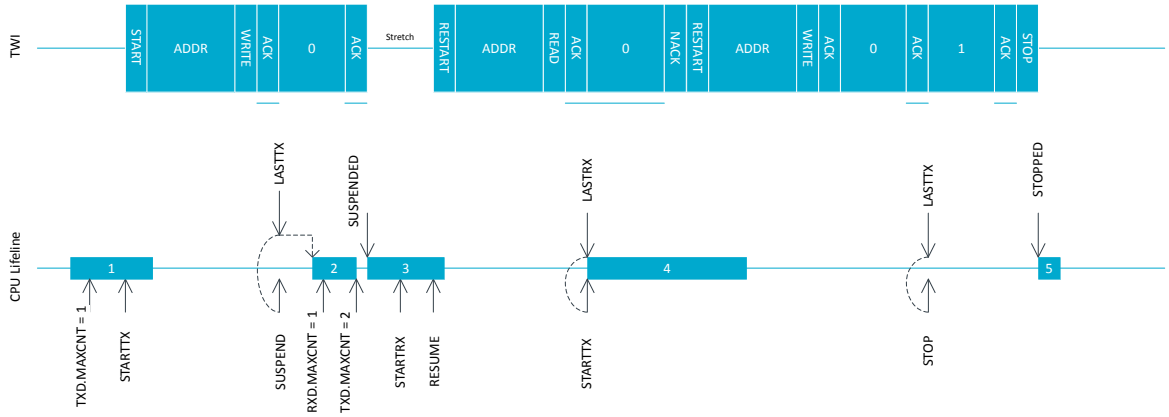


Figure 219: Double repeated start sequence

7.36.6 Low power

To ensure lowest possible power consumption when the peripheral is not needed stop and disable TWIM.

When the STOP task is sent, the software shall wait until the STOPPED event is received as a response before disabling the peripheral through the ENABLE register. If the peripheral is already stopped, the STOP task is not required.

7.36.7 Master mode pin configuration

The SCL and SDA signals are mapped to physical pins using the PSEL.SCL and PSEL.SDA registers.

The PSEL.SCL and PSEL.SDA registers and their configurations are only used as long as the TWI master is enabled, and retained only as long as the device is in System ON mode. When the peripheral is disabled, the pins will behave as regular GPIOs, and use the configuration in their respective OUT bit field and PIN_CNF[n] register. PSEL.SCL and PSEL.SDA must only be configured when the TWI master is disabled.

To secure correct signal levels on the pins used by TWIM while in System OFF mode, and when TWIM is disabled, these pins must be configured in the GPIO peripheral as described in the following table.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

TWI master signal	TWI master pin	Direction	Output value	Drive strength
SCL	As specified in PSEL.SCL[27]	Input	Not applicable	S0D1[27]
SDA	As specified in PSEL.SDA[27]	Input	Not applicable	S0D1[27]

Table 166: GPIO configuration before enabling peripheral

[27] Special pin and drive strength considerations applies when using the 1000 kbps baud rate. For pin recommendations, see Pin assignments on page 783.

7.36.8 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50008000 0x40008000	APPLICATION	TWIM	TWIM0 : S TWIM0 : NS	US	SA	Two-wire interface master 0	
0x50009000 0x40009000	APPLICATION	TWIM	TWIM1 : S TWIM1 : NS	US	SA	Two-wire interface master 1	
0x5000B000 0x4000B000	APPLICATION	TWIM	TWIM2 : S TWIM2 : NS	US	SA	Two-wire interface master 2	
0x5000C000 0x4000C000	APPLICATION	TWIM	TWIM3 : S TWIM3 : NS	US	SA	Two-wire interface master 3	
0x41013000	NETWORK	TWIM	TWIM0	NS	NA	Two-wire interface master 0	

Table 167: Instances

Register	Offset	Security	Description
TASKS_STARTRX	0x000		Start TWI receive sequence
TASKS_STARTTX	0x008		Start TWI transmit sequence
TASKS_STOP	0x014		Stop TWI transaction. Must be issued while the TWI master is not suspended.
TASKS_SUSPEND	0x01C		Suspend TWI transaction
TASKS_RESUME	0x020		Resume TWI transaction
SUBSCRIBE_STARTRX	0x080		Subscribe configuration for task STARTRX
SUBSCRIBE_STARTTX	0x088		Subscribe configuration for task STARTTX
SUBSCRIBE_STOP	0x094		Subscribe configuration for task STOP
SUBSCRIBE_SUSPEND	0x09C		Subscribe configuration for task SUSPEND
SUBSCRIBE_RESUME	0x0A0		Subscribe configuration for task RESUME
EVENTS_STOPPED	0x104		TWI stopped
EVENTS_ERROR	0x124		TWI error
EVENTS_SUSPENDED	0x148		SUSPEND task has been issued, TWI traffic is now suspended.
EVENTS_RXSTARTED	0x14C		Receive sequence started
EVENTS_TXSTARTED	0x150		Transmit sequence started
EVENTS_LASTRX	0x15C		Byte boundary, starting to receive the last byte
EVENTS_LASTTX	0x160		Byte boundary, starting to transmit the last byte
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
PUBLISH_ERROR	0x1A4		Publish configuration for event ERROR
PUBLISH_SUSPENDED	0x1C8		Publish configuration for event SUSPENDED
PUBLISH_RXSTARTED	0x1CC		Publish configuration for event RXSTARTED
PUBLISH_TXSTARTED	0x1D0		Publish configuration for event TXSTARTED
PUBLISH_LASTRX	0x1DC		Publish configuration for event LASTRX
PUBLISH_LASTTX	0x1E0		Publish configuration for event LASTTX
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ERRORSRC	0x4C4		Error source
ENABLE	0x500		Enable TWIM
PSEL.SCL	0x508		Pin select for SCL signal
PSEL.SDA	0x50C		Pin select for SDA signal
FREQUENCY	0x524		TWI frequency. Accuracy depends on the HFCLK source selected.
RXD.PTR	0x534		Data pointer
RXD.MAXCNT	0x538		Maximum number of bytes in receive buffer
RXD.AMOUNT	0x53C		Number of bytes transferred in the last transaction

Register	Offset	Security	Description
RXD.LIST	0x540		EasyDMA list type
TXD.PTR	0x544		Data pointer
TXD.MAXCNT	0x548		Maximum number of bytes in transmit buffer
TXD.AMOUNT	0x54C		Number of bytes transferred in the last transaction
TXD.LIST	0x550		EasyDMA list type
ADDRESS	0x588		Address used in the TWI transfer

Table 168: Register overview

7.36.8.1 TASKS_STARTRX

Address offset: 0x000

Start TWI receive sequence

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STARTRX			Start TWI receive sequence
			Trigger	1	Trigger task

7.36.8.2 TASKS_STARTTX

Address offset: 0x008

Start TWI transmit sequence

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STARTTX			Start TWI transmit sequence
			Trigger	1	Trigger task

7.36.8.3 TASKS_STOP

Address offset: 0x014

Stop TWI transaction. Must be issued while the TWI master is not suspended.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Stop TWI transaction. Must be issued while the TWI master is not suspended.
			Trigger	1	Trigger task

7.36.8.4 TASKS_SUSPEND

Address offset: 0x01C

Suspend TWI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_SUSPEND			Suspend TWI transaction
			Trigger	1	Trigger task

7.36.8.5 TASKS_RESUME

Address offset: 0x020

Resume TWI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_RESUME			Resume TWI transaction
			Trigger	1	Trigger task

7.36.8.6 SUBSCRIBE_STARTRX

Address offset: 0x080

Subscribe configuration for task STARTRX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STARTRX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.36.8.7 SUBSCRIBE_STARTTX

Address offset: 0x088

Subscribe configuration for task STARTTX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STARTTX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.36.8.8 SUBSCRIBE_STOP

Address offset: 0x094

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.36.8.9 SUBSCRIBE_SUSPEND

Address offset: 0x09C

Subscribe configuration for task SUSPEND

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task SUSPEND will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.36.8.10 SUBSCRIBE_RESUME

Address offset: 0x0A0

Subscribe configuration for task RESUME

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task RESUME will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.36.8.11 EVENTS_STOPPED

Address offset: 0x104

TWI stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_STOPPED			TWI stopped
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.36.8.12 EVENTS_ERROR

Address offset: 0x124

TWI error

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ERROR			TWI error
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.36.8.13 EVENTS_SUSPENDED

Address offset: 0x148

SUSPEND task has been issued, TWI traffic is now suspended.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_SUSPENDED			SUSPEND task has been issued, TWI traffic is now suspended.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.36.8.14 EVENTS_RXSTARTED

Address offset: 0x14C

Receive sequence started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RXSTARTED			Receive sequence started
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.36.8.15 EVENTS_TXSTARTED

Address offset: 0x150

Transmit sequence started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_TXSTARTED			Transmit sequence started
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.36.8.16 EVENTS_LASTRX

Address offset: 0x15C

Byte boundary, starting to receive the last byte

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_LASTRX			Byte boundary, starting to receive the last byte
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.36.8.17 EVENTS_LASTTX

Address offset: 0x160

Byte boundary, starting to transmit the last byte

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_LASTTX			Byte boundary, starting to transmit the last byte
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.36.8.18 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event STOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event STOPPED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.36.8.19 PUBLISH_ERROR

Address offset: 0x1A4

Publish configuration for event ERROR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ERROR will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.36.8.20 PUBLISH_SUSPENDED

Address offset: 0x1C8

Publish configuration for event SUSPENDED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event SUSPENDED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.36.8.21 PUBLISH_RXSTARTED

Address offset: 0x1CC

Publish configuration for event RXSTARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RXSTARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.36.8.22 PUBLISH_TXSTARTED

Address offset: 0x1D0

Publish configuration for event TXSTARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event TXSTARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.36.8.23 PUBLISH_LASTRX

Address offset: 0x1DC

Publish configuration for event LASTRX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event LASTRX will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.36.8.24 PUBLISH_LASTTX

Address offset: 0x1E0

Publish configuration for event LASTTX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event LASTTX will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.36.8.25 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LASTTX_STARTRX			Shortcut between event LASTTX and task STARTRX
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
B	RW	LASTTX_SUSPEND			Shortcut between event LASTTX and task SUSPEND
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
C	RW	LASTTX_STOP			Shortcut between event LASTTX and task STOP
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
D	RW	LASTRX_STARTTX			Shortcut between event LASTRX and task STARTTX
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
F	RW	LASTRX_STOP			Shortcut between event LASTRX and task STOP
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.36.8.26 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					J I H G F D A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STOPPED			Enable or disable interrupt for event STOPPED
			Disabled	0	Disable
			Enabled	1	Enable
D	RW	ERROR			Enable or disable interrupt for event ERROR
			Disabled	0	Disable
			Enabled	1	Enable
F	RW	SUSPENDED			Enable or disable interrupt for event SUSPENDED
			Disabled	0	Disable
			Enabled	1	Enable
G	RW	RXSTARTED			Enable or disable interrupt for event RXSTARTED
			Disabled	0	Disable
			Enabled	1	Enable
H	RW	TXSTARTED			Enable or disable interrupt for event TXSTARTED
			Disabled	0	Disable
			Enabled	1	Enable
I	RW	LASTRX			Enable or disable interrupt for event LASTRX
			Disabled	0	Disable
			Enabled	1	Enable
J	RW	LASTTX			Enable or disable interrupt for event LASTTX
			Disabled	0	Disable
			Enabled	1	Enable

7.36.8.27 INTENSET

Address offset: 0x304

Enable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					J I H G F D A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STOPPED			Write '1' to enable interrupt for event STOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	ERROR			Write '1' to enable interrupt for event ERROR
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	SUSPENDED			Write '1' to enable interrupt for event SUSPENDED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	RXSTARTED			Write '1' to enable interrupt for event RXSTARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				J I H G F D A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	TXSTARTED			Write '1' to enable interrupt for event TXSTARTED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
I	RW	LASTRX			Write '1' to enable interrupt for event LASTRX
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
J	RW	LASTTX			Write '1' to enable interrupt for event LASTTX
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.36.8.28 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				J I H G F D A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STOPPED			Write '1' to disable interrupt for event STOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	ERROR			Write '1' to disable interrupt for event ERROR
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	SUSPENDED			Write '1' to disable interrupt for event SUSPENDED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	RXSTARTED			Write '1' to disable interrupt for event RXSTARTED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	TXSTARTED			Write '1' to disable interrupt for event TXSTARTED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
I	RW	LASTRX			Write '1' to disable interrupt for event LASTRX
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				J I H G F D A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
J	RW	LASTTX			Write '1' to disable interrupt for event LASTTX
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.36.8.29 ERRORSRC

Address offset: 0x4C4

Error source

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	OVERRUN			Overrun error
					A new byte was received before previous byte got transferred into RXD buffer. (Previous data is lost)
			NotReceived	0	Error did not occur
			Received	1	Error occurred
B	RW	ANACK			NACK received after sending the address (write '1' to clear)
			NotReceived	0	Error did not occur
			Received	1	Error occurred
C	RW	DNACK			NACK received after sending a data byte (write '1' to clear)
			NotReceived	0	Error did not occur
			Received	1	Error occurred

7.36.8.30 ENABLE

Address offset: 0x500

Enable TWIM

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	ENABLE			Enable or disable TWIM
			Disabled	0	Disable TWIM
			Enabled	6	Enable TWIM

7.36.8.31 PSEL.SCL

Address offset: 0x508

Pin select for SCL signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.36.8.32 PSEL.SDA

Address offset: 0x50C

Pin select for SDA signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.36.8.33 FREQUENCY

Address offset: 0x524

TWI frequency. Accuracy depends on the HFCLK source selected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x04000000				**0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	FREQUENCY			TWI master clock frequency
			K100	0x01980000	100 kbps
			K250	0x04000000	250 kbps
			K400	0x06400000	400 kbps
			K1000	0x0FF00000	1000 kbps

7.36.8.34 RXD.PTR

Address offset: 0x534

Data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer
					See the memory chapter for details about which memories are available for EasyDMA.

7.36.8.35 RXD.MAXCNT

Address offset: 0x538

Maximum number of bytes in receive buffer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in receive buffer

7.36.8.36 RXD.AMOUNT

Address offset: 0x53C

Number of bytes transferred in the last transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.

7.36.8.37 RXD.LIST

Address offset: 0x540

EasyDMA list type

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LIST			List type
			Disabled	0	Disable EasyDMA list
			ArrayList	1	Use array list

7.36.8.38 TXD.PTR

Address offset: 0x544

Data pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the memory chapter for details about which memories are available for EasyDMA.

7.36.8.39 TXD.MAXCNT

Address offset: 0x548

Maximum number of bytes in transmit buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in transmit buffer

7.36.8.40 TXD.AMOUNT

Address offset: 0x54C

Number of bytes transferred in the last transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.

7.36.8.41 TXD.LIST

Address offset: 0x550

EasyDMA list type

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LIST			List type
			Disabled	0	Disable EasyDMA list
			ArrayList	1	Use array list

7.36.8.42 ADDRESS

Address offset: 0x588

Address used in the TWI transfer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ADDRESS			Address used in the TWI transfer

7.36.9 Electrical specification

7.36.9.1 TWIM interface electrical specifications

Symbol	Description	Min.	Typ.	Max.	Units
$f_{TWIM,SCL}$	Bit rates for TWIM[28]	100		1000	kbps
$t_{TWIM,START}$	Time from STARTRX/STARTTX task to transmission started		1.5		µs

7.36.9.2 Two Wire Interface Master (TWIM) timing specifications

Symbol	Description	Min.	Typ.	Max.	Units
t_{TWIM,SU_DAT}	Data setup time before positive edge on SCL – all modes	20			ns
t_{TWIM,HD_DAT}	Data hold time after negative edge on SCL – 100, 250 and 400 kbps	500		625	ns
t_{TWIM,HD_DAT}	Data hold time after negative edge on SCL – 1000 kbps	250		315	ns
$t_{TWIM,HD\_STA,100kbps}$	TWIM master hold time for START and repeated START condition, 100 kbps	9900			ns
$t_{TWIM,HD\_STA,250kbps}$	TWIM master hold time for START and repeated START condition, 250 kbps	3900			ns
$t_{TWIM,HD\_STA,400kbps}$	TWIM master hold time for START and repeated START condition, 400 kbps	2400			ns
$t_{TWIM,HD\_STA,1000kbps}$	TWIM master hold time for START and repeated START condition, 1000 kbps	900			ns
$t_{TWIM,SU\_STO,100kbps}$	TWIM master setup time from SCL high to STOP condition, 100 kbps	5000			ns
$t_{TWIM,SU\_STO,250kbps}$	TWIM master setup time from SCL high to STOP condition, 250 kbps	2000			ns
$t_{TWIM,SU\_STO,400kbps}$	TWIM master setup time from SCL high to STOP condition, 400 kbps	1250			ns
$t_{TWIM,SU\_STO,1000kbps}$	TWIM master setup time from SCL high to STOP condition, 1000 kbps	500			ns
$t_{TWIM,BUF,100kbps}$	TWIM master bus free time between STOP and START conditions, 100 kbps	5250			ns
$t_{TWIM,BUF,250kbps}$	TWIM master bus free time between STOP and START conditions, 250 kbps	2250			ns
$t_{TWIM,BUF,400kbps}$	TWIM master bus free time between STOP and START conditions, 400 kbps	1500			ns
$t_{TWIM,BUF,1000kbps}$	TWIM master bus free time between STOP and START conditions, 1000 kbps	750			ns

[28] High bit rates or stronger pull-ups may require GPIOs to be set as High Drive, see GPIO — General purpose input/output on page 220 for more details.

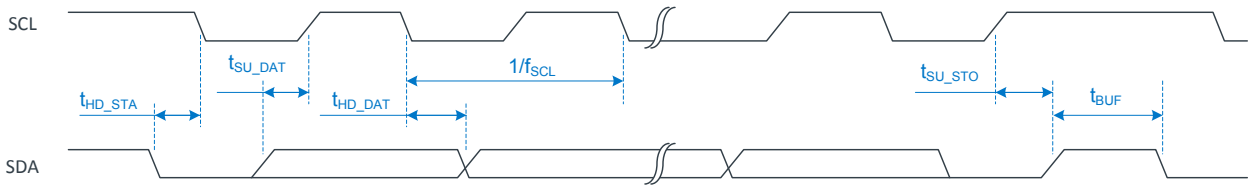


Figure 220: TWIM timing diagram, 1 byte transaction

7.36.10 Pullup resistor

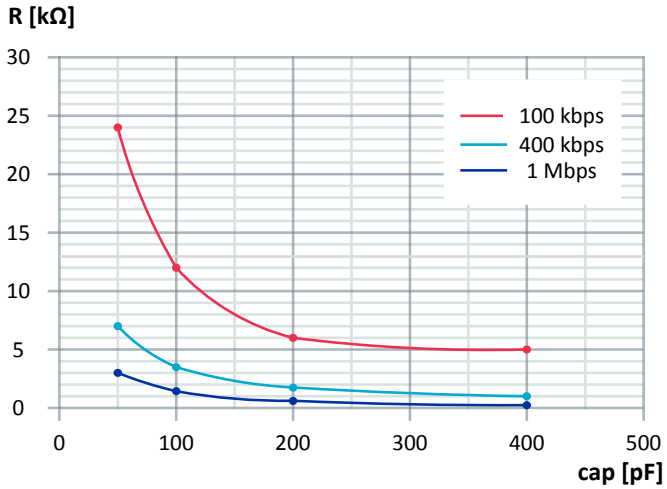


Figure 221: Recommended TWIM pullup value vs. line capacitance

- The I^2C specification allows a line capacitance of 400 pF at most.
- The value of internal pullup resistor (R_{PU}) for nRF5340 can be found in GPIO — General purpose input/output on page 220.

7.37 TWIS — I^2C compatible two-wire interface slave with EasyDMA

TWI slave with EasyDMA (TWIS) is a two-wire half-duplex slave which can communicate with a master device connected to the same bus.

Listed here are the main features for TWIS:

- I^2C compatible
- Supported baud rates: 100 and 400 kbps
- EasyDMA

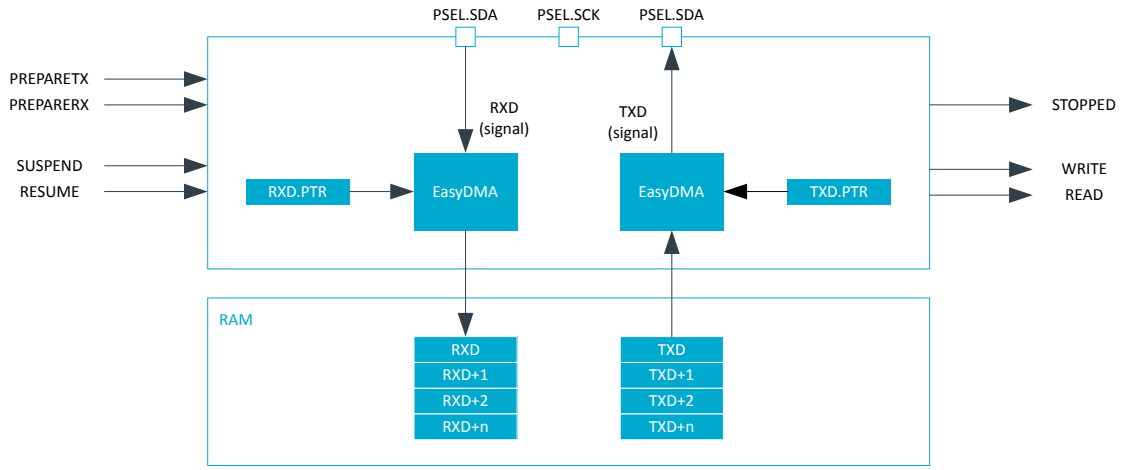


Figure 222: TWI slave with EasyDMA

A typical TWI setup consists of one master and one or more slaves. For an example, see the following figure. TWIS is only able to operate with a single master on the TWI bus.

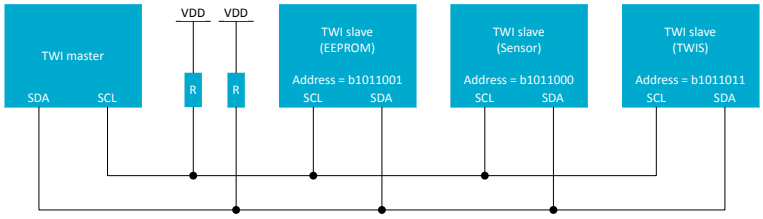


Figure 223: A typical TWI setup comprising one master and three slaves

The following figure shows the TWI slave state machine.

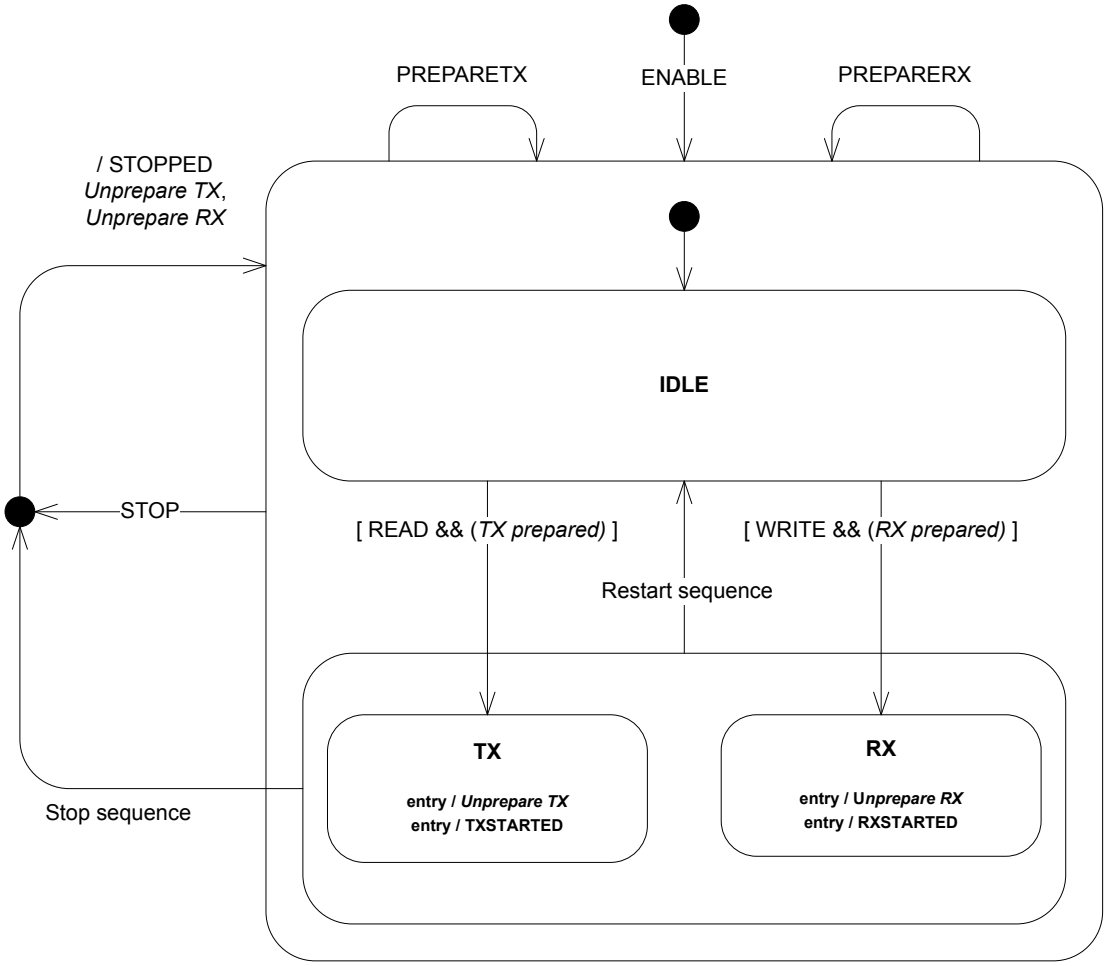


Figure 224: TWI slave state machine

The following table contains descriptions of the symbols used in the state machine.

Symbol	Type	Description
ENABLE	Register	The TWI slave has been enabled via the ENABLE register.
PREPARETX	Task	The TASKS_PREPARETX task has been triggered.
STOP	Task	The TASKS_STOP task has been triggered.
PREPARERX	Task	The TASKS_PREPARERX task has been triggered.
STOPPED	Event	The EVENTS_STOPPED event was generated.
RXSTARTED	Event	The EVENTS_RXSTARTED event was generated.
TXSTARTED	Event	The EVENTS_TXSTARTED event was generated.
TX prepared	Internal	Internal flag indicating that a TASKS_PREPARETX task has been triggered. This flag is not visible to the user.
RX prepared	Internal	Internal flag indicating that a TASKS_PREPARERX task has been triggered. This flag is not visible to the user.
Unprepare TX	Internal	Clears the internal 'TX prepared' flag until next TASKS_PREPARETX task.
Unprepare RX	Internal	Clears the internal 'RX prepared' flag until next TASKS_PREPARERX task.
Stop condition	TWI protocol	A TWI stop condition was detected.
Restart condition	TWI protocol	A TWI restart condition was detected.

Table 169: TWI slave state machine symbols

TWIS can perform clock stretching, with the premise that the master is able to support it.

It operates in a low power mode while waiting for a TWI master to initiate a transfer. As long as TWIS is not addressed, it will remain in this low power mode.

To secure correct behavior of the TWI slave, PSEL.SCL, PSEL.SDA, CONFIG, and the ADDRESS[n] registers must be configured prior to enabling the TWI slave through the ENABLE register. Similarly, changing these settings must be performed while the TWI slave is disabled. Failing to do so may result in unpredictable behavior.

7.37.1 Shared resources

TWIS shares registers and other resources with other peripherals that have the same ID as TWIS.

Therefore, you must disable all peripherals that have the same ID as the TWI slave before TWIS can be configured and used. Disabling a peripheral that has the same ID as the TWI slave will not reset any of the registers that are shared with TWIS. It is therefore important to configure all relevant registers explicitly to secure that TWIS operates correctly.

The Instantiation table in Peripherals on page 146 shows which peripherals have the same ID as TWIS.

7.37.2 EasyDMA

TWIS implements EasyDMA for accessing RAM without CPU involvement.

The following table shows the Easy DMA channels that TWIS implements.

Channel	Type	Register Cluster
TXD	READER	TXD
RXD	WRITER	RXD

Table 170: TWIS EasyDMA Channels

For detailed information regarding the use of EasyDMA, see EasyDMA on page 150.

The STOPPED event indicates that EasyDMA has finished accessing the buffer in RAM.

7.37.3 TWIS responding to a read command

Before TWIS can respond to a read command, it must be configured correctly and enabled via the ENABLE register. When enabled, TWIS will be in its IDLE state.

A read command is started when the TWI master generates a start condition on the TWI bus, followed by clocking out the address and the READ/WRITE bit set to `1` (WRITE=0, READ=1). The READ/WRITE bit is followed by an ACK/NACK bit (ACK=0 or NACK=1) response from the TWI slave.

TWIS is able to listen for up to two addresses at the same time. This is configured in the ADDRESS registers and the CONFIG register.

TWIS will only acknowledge (ACK) the read command if the address presented by the master matches one of the addresses the slave is configured to listen for. TWIS will generate a READ event when it acknowledges the read command.

TWIS is only able to detect a read command from the IDLE state.

TWIS will set an internal 'TX prepared' flag when the PREPARETX task is triggered.

When the read command is received, TWIS will enter the TX state if the internal 'TX prepared' flag is set.

If the internal 'TX prepared' flag is not set when the read command is received, TWIS will stretch the master's clock until the PREPARETX task is triggered and the internal 'TX prepared' flag is set.

TWIS will generate the TXSTARTED event and clear the 'TX prepared' flag ('unprepare TX') when it enters the TX state. In this state TWIS will send the data bytes found in the transmit buffer to the master using the master's clock.

TWIS will go back to the IDLE state if the TWI slave receives a restart command when it is in the TX state.

TWIS is stopped when it receives the stop condition from the TWI master. A STOPPED event will be generated when the transaction has stopped. TWIS will clear the 'TX prepared' flag ('unprepare TX') and go back to the IDLE state when it has stopped.

The transmit buffer is located in RAM at the address specified in the TXD.PTR register. TWIS will only be able to send TXD.MAXCNT bytes from the transmit buffer for each transaction. If the TWI master forces the slave to send more than TXD.MAXCNT bytes, the slave will send the byte specified in the ORC register to the master instead. If this happens, an ERROR event will be generated.

The EasyDMA configuration registers RXD.PTR, TXD.PTR, RXD.AMOUNT, and TXD.AMOUNT, are latched when the TXSTARTED event is generated.

TWIS can be forced to stop by triggering the STOP task. A STOPPED event will be generated when TWIS has stopped. TWIS will clear the 'TX prepared' flag and go back to the IDLE state when it has stopped, see also Terminating an ongoing TWI transaction on page 654.

Each byte sent from the slave will be followed by an ACK/NACK bit sent from the master. The TWI master will generate a NACK following the last byte that it wants to receive to tell the slave to release the bus so that the TWI master can generate the stop condition. The TXD.AMOUNT register can be queried after a transaction to see how many bytes were sent.

A typical TWIS read command response is illustrated in the following figure, including clock stretching following a SUSPEND task.

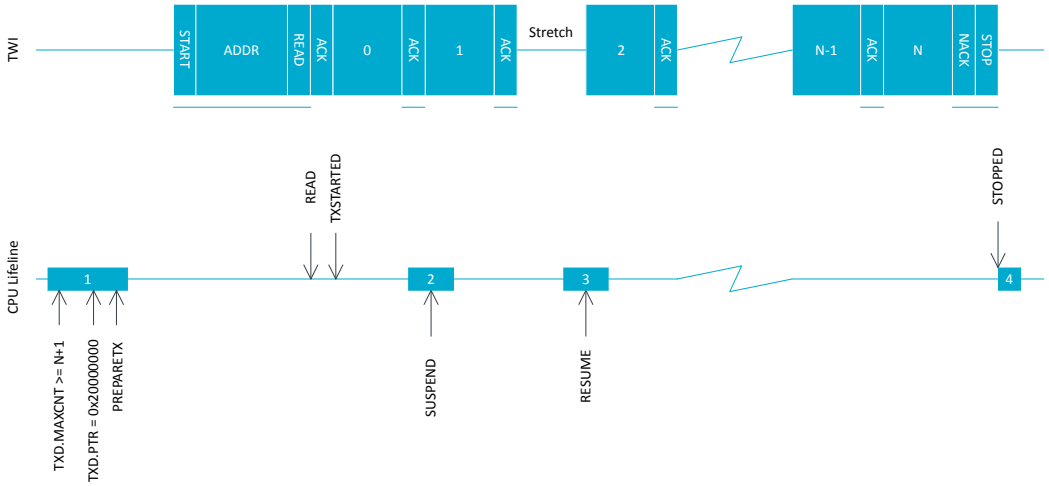


Figure 225: TWIS responding to a read command

7.37.4 TWIS responding to a write command

Before TWIS can respond to a write command, TWIS must be configured correctly and enabled via the ENABLE register. When enabled, TWIS will be in its IDLE state.

A write command is started when the TWI master generates a start condition on the TWI bus, followed by clocking out the address and the READ/WRITE bit set to 0 (WRITE=0, READ=1). The READ/WRITE bit is followed by an ACK/NACK bit (ACK=0 or NACK=1) response from the slave.

TWIS is able to listen for up to two addresses at the same time. This is configured in the ADDRESS registers and the CONFIG register.

TWIS will only acknowledge (ACK) the write command if the address presented by the master matches one of the addresses the slave is configured to listen for. TWIS will generate a WRITE event if it acknowledges the write command.

TWIS is only able to detect a write command from the IDLE state.

TWIS will set an internal 'RX prepared' flag when the PREPARERX task is triggered.

When the write command is received, TWIS will enter the RX state if the internal 'RX prepared' flag is set.

If the internal 'RX prepared' flag is not set when the write command is received, TWIS will stretch the master's clock until the PREPARERX task is triggered and the internal 'RX prepared' flag is set.

TWIS will generate the RXSTARTED event and clear the internal 'RX prepared' flag ('unprepare RX') when it enters the RX state. In this state, TWIS will be able to receive the bytes sent by the TWI master.

TWIS will go back to the IDLE state if TWIS receives a restart command when it is in the RX state.

TWIS is stopped when it receives the stop condition from the TWI master. A STOPPED event will be generated when the transaction has stopped. TWIS will clear the internal 'RX prepared' flag ('unprepare RX') and go back to the IDLE state when it has stopped.

The receive buffer is located in RAM at the address specified in the RXD.PTR register. TWIS will only be able to receive as many bytes as specified in the RXD.MAXCNT register. If the TWI master tries to send more bytes to the slave than it can receive, the extra bytes are discarded and NACKed by the slave. If this happens, an ERROR event will be generated.

The EasyDMA configuration registers, see RXD.PTR etc., are latched when the RXSTARTED event is generated.

TWIS can be forced to stop by triggering the STOP task. A STOPPED event will be generated when TWIS has stopped. TWIS will clear the internal 'RX prepared' flag and go back to the IDLE state when it has stopped, see also Terminating an ongoing TWI transaction on page 654.

TWIS will generate an ACK after every byte received from the master. The RXD.AMOUNT register can be queried after a transaction to see how many bytes were received.

A typical TWIS write command response is illustrated in the following figure, including clock stretching following a SUSPEND task.

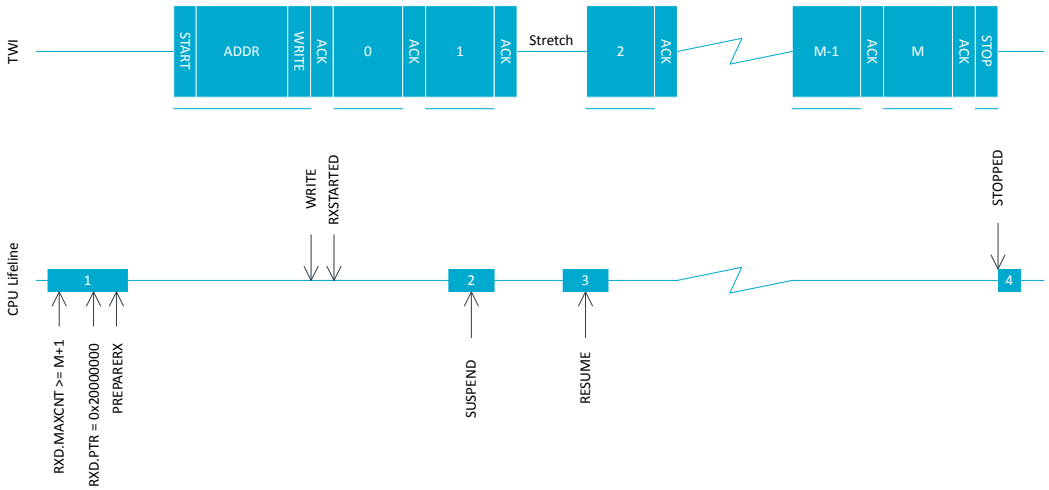


Figure 226: TWIS responding to a write command

7.37.5 Master repeated start sequence

An example of a repeated start sequence is one in which the TWI master writes two bytes to TWIS followed by reading four bytes from the slave.

This is illustrated in the following figure.

In this example, the receiver does not know what the master wants to read in advance. This information is in the first two received bytes of the write in the repeated start sequence. To guarantee that the CPU is able to process the received data before TWIS starts to reply to the read command, the SUSPEND task is triggered via a shortcut from the READ event generated when the read command is received. When the CPU has processed the incoming data and prepared the correct data response, the CPU will resume the transaction by triggering the RESUME task.

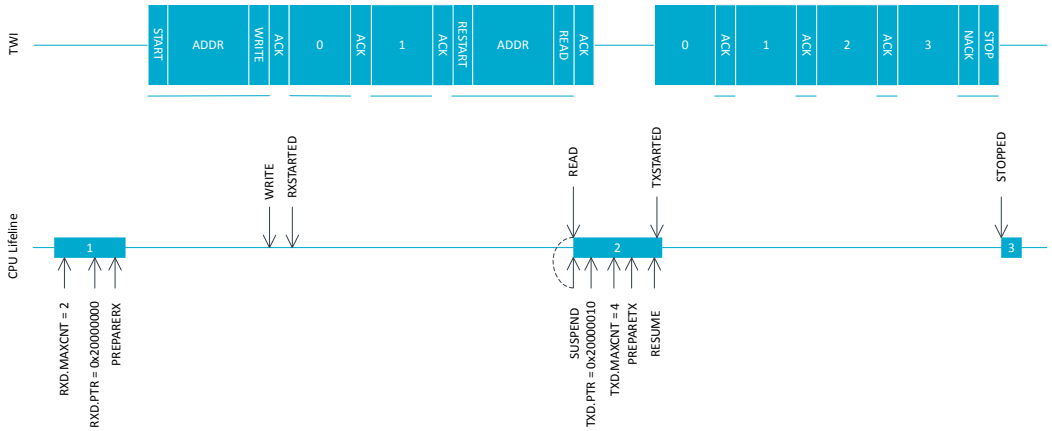


Figure 227: Repeated start sequence

7.37.6 Terminating an ongoing TWI transaction

In some situations, e.g. if the external TWI master is not responding correctly, it may be required to terminate an ongoing transaction.

This can be achieved by triggering the STOP task. In this situation, a STOPPED event will be generated when the TWI has stopped independent of whether or not a STOP condition has been generated on the TWI bus. The TWI slave will release the bus when it has stopped and go back to its IDLE state.

7.37.7 Low power

To ensure lowest possible power consumption when the peripheral is not needed stop and disable TWIS.

The STOP task may not be always needed (the peripheral might already be stopped), but if the task is triggered, software shall wait until the STOPPED event is generated before disabling the peripheral through the ENABLE register.

7.37.8 Slave mode pin configuration

The SCL and SDA signals are mapped to physical pins using the PSEL.SCL and PSEL.SDA registers.

The PSEL.SCL and PSEL.SDA registers and their configurations are only used as long as TWIS is enabled, and retained only as long as the device is in System ON mode. When the peripheral is disabled, the pins will behave as regular GPIOs, and use the configuration in their respective OUT bit field and PIN_CNF[n] register. PSEL.SCL and PSEL.SDA must only be configured when TWIS is disabled.

To secure correct signal levels on the pins used by TWIS while in System OFF mode, and when TWIS is disabled, these pins must be configured in the GPIO peripheral as described in the following table.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

TWI slave signal	TWI slave pin	Direction	Output value	Drive strength
SCL	As specified in PSEL.SCL	Input	Not applicable	S0D1
SDA	As specified in PSEL.SDA	Input	Not applicable	S0D1

Table 171: GPIO configuration before enabling peripheral

7.37.9 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50008000 0x40008000	APPLICATION	TWIS	TWIS0 : S TWIS0 : NS	US	SA	Two-wire interface slave 0	
0x50009000 0x40009000	APPLICATION	TWIS	TWIS1 : S TWIS1 : NS	US	SA	Two-wire interface slave 1	
0x5000B000 0x4000B000	APPLICATION	TWIS	TWIS2 : S TWIS2 : NS	US	SA	Two-wire interface slave 2	
0x5000C000 0x4000C000	APPLICATION	TWIS	TWIS3 : S TWIS3 : NS	US	SA	Two-wire interface slave 3	
0x41013000	NETWORK	TWIS	TWIS0	NS	NA	Two-wire interface slave 0	

Table 172: Instances

Register	Offset	Security	Description
TASKS_STOP	0x014		Stop TWI transaction
TASKS_SUSPEND	0x01C		Suspend TWI transaction
TASKS_RESUME	0x020		Resume TWI transaction
TASKS_PREPARERX	0x030		Prepare the TWI slave to respond to a write command
TASKS_PREPARETX	0x034		Prepare the TWI slave to respond to a read command
SUBSCRIBE_STOP	0x094		Subscribe configuration for task STOP
SUBSCRIBE_SUSPEND	0x09C		Subscribe configuration for task SUSPEND
SUBSCRIBE_RESUME	0x0A0		Subscribe configuration for task RESUME
SUBSCRIBE_PREPARERX	0x0B0		Subscribe configuration for task PREPARERX
SUBSCRIBE_PREPARETX	0x0B4		Subscribe configuration for task PREPARETX
EVENTS_STOPPED	0x104		TWI stopped
EVENTS_ERROR	0x124		TWI error
EVENTS_RXSTARTED	0x14C		Receive sequence started
EVENTS_TXSTARTED	0x150		Transmit sequence started
EVENTS_WRITE	0x164		Write command received
EVENTS_READ	0x168		Read command received
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
PUBLISH_ERROR	0x1A4		Publish configuration for event ERROR
PUBLISH_RXSTARTED	0x1CC		Publish configuration for event RXSTARTED
PUBLISH_TXSTARTED	0x1D0		Publish configuration for event TXSTARTED
PUBLISH_WRITE	0x1E4		Publish configuration for event WRITE
PUBLISH_READ	0x1E8		Publish configuration for event READ
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ERRORSRC	0x4D0		Error source
MATCH	0x4D4		Status register indicating which address had a match
ENABLE	0x500		Enable TWIS
PSEL.SCL	0x508		Pin select for SCL signal
PSEL.SDA	0x50C		Pin select for SDA signal

Register	Offset	Security	Description
RXD.PTR	0x534		RXD Data pointer
RXD.MAXCNT	0x538		Maximum number of bytes in RXD buffer
RXD.AMOUNT	0x53C		Number of bytes transferred in the last RXD transaction
RXD.LIST	0x540		EasyDMA list type
TXD.PTR	0x544		TXD Data pointer
TXD.MAXCNT	0x548		Maximum number of bytes in TXD buffer
TXD.AMOUNT	0x54C		Number of bytes transferred in the last TXD transaction
TXD.LIST	0x550		EasyDMA list type
ADDRESS[n]	0x588		TWI slave address n
CONFIG	0x594		Configuration register for the address match mechanism
ORC	0x5C0		Over-read character. Character sent out in case of an over-read of the transmit buffer.

Table 173: Register overview

7.37.9.1 TASKS_STOP

Address offset: 0x014

Stop TWI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Stop TWI transaction
			Trigger	1	Trigger task

7.37.9.2 TASKS_SUSPEND

Address offset: 0x01C

Suspend TWI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_SUSPEND			Suspend TWI transaction
			Trigger	1	Trigger task

7.37.9.3 TASKS_RESUME

Address offset: 0x020

Resume TWI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_RESUME			Resume TWI transaction
			Trigger	1	Trigger task

7.37.9.4 TASKS_PREPARERX

Address offset: 0x030

Prepare the TWI slave to respond to a write command

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A
Reset 0x00000000						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description	
A	W	TASKS_PREPARERX			Prepare the TWI slave to respond to a write command	
			Trigger	1	Trigger task	

7.37.9.5 TASKS_PREPARETX

Address offset: 0x034

Prepare the TWI slave to respond to a read command

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A
Reset 0x00000000						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description	
A	W	TASKS_PREPARETX			Prepare the TWI slave to respond to a read command	
			Trigger	1	Trigger task	

7.37.9.6 SUBSCRIBE_STOP

Address offset: 0x094

Subscribe configuration for task STOP

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B		A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**		
ID	R/W	Field	Value ID	Value	Description	
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to	
B	RW	EN				
			Disabled	0	Disable subscription	
			Enabled	1	Enable subscription	

7.37.9.7 SUBSCRIBE_SUSPEND

Address offset: 0x09C

Subscribe configuration for task SUSPEND

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B		A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**		
ID	R/W	Field	Value ID	Value	Description	
A	RW	CHIDX		[255..0]	DPPI channel that task SUSPEND will subscribe to	
B	RW	EN				
			Disabled	0	Disable subscription	
			Enabled	1	Enable subscription	

7.37.9.8 SUBSCRIBE_RESUME

Address offset: 0x0A0

Subscribe configuration for task RESUME

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task RESUME will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.37.9.9 SUBSCRIBE_PREPARERX

Address offset: 0x0B0

Subscribe configuration for task PREPARERX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task PREPARERX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.37.9.10 SUBSCRIBE_PREPARETX

Address offset: 0x0B4

Subscribe configuration for task PREPARETX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task PREPARETX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.37.9.11 EVENTS_STOPPED

Address offset: 0x104

TWI stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_STOPPED			TWI stopped
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.37.9.12 EVENTS_ERROR

Address offset: 0x124

TWI error

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ERROR			TWI error
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.37.9.13 EVENTS_RXSTARTED

Address offset: 0x14C

Receive sequence started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RXSTARTED			Receive sequence started
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.37.9.14 EVENTS_TXSTARTED

Address offset: 0x150

Transmit sequence started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_TXSTARTED			Transmit sequence started
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.37.9.15 EVENTS_WRITE

Address offset: 0x164

Write command received

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_WRITE			Write command received
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.37.9.16 EVENTS_READ

Address offset: 0x168

Read command received

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_READ			Read command received
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.37.9.17 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event STOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event STOPPED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.37.9.18 PUBLISH_ERROR

Address offset: 0x1A4

Publish configuration for event ERROR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ERROR will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.37.9.19 PUBLISH_RXSTARTED

Address offset: 0x1CC

Publish configuration for event RXSTARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RXSTARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.37.9.20 PUBLISH_TXSTARTED

Address offset: 0x1D0

Publish configuration for event TXSTARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event TXSTARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.37.9.21 PUBLISH_WRITE

Address offset: 0x1E4

Publish configuration for event WRITE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event WRITE will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.37.9.22 PUBLISH_READ

Address offset: 0x1E8

Publish configuration for event READ

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event READ will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.37.9.23 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	WRITE_SUSPEND			Shortcut between event WRITE and task SUSPEND
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
B	RW	READ_SUSPEND			Shortcut between event READ and task SUSPEND
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.37.9.24 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STOPPED			Enable or disable interrupt for event STOPPED
			Disabled	0	Disable
			Enabled	1	Enable
B	RW	ERROR			Enable or disable interrupt for event ERROR
			Disabled	0	Disable
			Enabled	1	Enable
E	RW	RXSTARTED			Enable or disable interrupt for event RXSTARTED
			Disabled	0	Disable
			Enabled	1	Enable
F	RW	TXSTARTED			Enable or disable interrupt for event TXSTARTED
			Disabled	0	Disable
			Enabled	1	Enable
G	RW	WRITE			Enable or disable interrupt for event WRITE
			Disabled	0	Disable
			Enabled	1	Enable
H	RW	READ			Enable or disable interrupt for event READ
			Disabled	0	Disable
			Enabled	1	Enable

7.37.9.25 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STOPPED			Write '1' to enable interrupt for event STOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	ERROR			Write '1' to enable interrupt for event ERROR
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	RXSTARTED			Write '1' to enable interrupt for event RXSTARTED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	TXSTARTED			Write '1' to enable interrupt for event TXSTARTED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	WRITE			Write '1' to enable interrupt for event WRITE
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	READ			Write '1' to enable interrupt for event READ
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.37.9.26 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STOPPED			Write '1' to disable interrupt for event STOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	ERROR			Write '1' to disable interrupt for event ERROR
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	RXSTARTED			Write '1' to disable interrupt for event RXSTARTED
			Clear	1	Disable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	TXSTARTED			Write '1' to disable interrupt for event TXSTARTED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	WRITE			Write '1' to disable interrupt for event WRITE
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	READ			Write '1' to disable interrupt for event READ
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.37.9.27 ERRORSRC

Address offset: 0x4D0

Error source

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	OVERFLOW			RX buffer overflow detected, and prevented
			NotDetected	0	Error did not occur
			Detected	1	Error occurred
B	RW	DNACK			NACK sent after receiving a data byte
			NotReceived	0	Error did not occur
			Received	1	Error occurred
C	RW	OVERREAD			TX buffer over-read detected, and prevented
			NotDetected	0	Error did not occur
			Detected	1	Error occurred

7.37.9.28 MATCH

Address offset: 0x4D4

Status register indicating which address had a match

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	MATCH		[0..1]	Indication of which address in {ADDRESS} that matched the incoming address

7.37.9.29 ENABLE

Address offset: 0x500

Enable TWIS

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ENABLE			Enable or disable TWIS
			Disabled	0	Disable TWIS
			Enabled	9	Enable TWIS

7.37.9.30 PSEL.SCL

Address offset: 0x508

Pin select for SCL signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**

ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.37.9.31 PSEL.SDA

Address offset: 0x50C

Pin select for SDA signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**

ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.37.9.32 RXD.PTR

Address offset: 0x534

RXD Data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			RXD Data pointer
					See the memory chapter for details about which memories are available for EasyDMA.

7.37.9.33 RXD.MAXCNT

Address offset: 0x538

Maximum number of bytes in RXD buffer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in RXD buffer

7.37.9.34 RXD.AMOUNT

Address offset: 0x53C

Number of bytes transferred in the last RXD transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transferred in the last RXD transaction

7.37.9.35 RXD.LIST

Address offset: 0x540

EasyDMA list type

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LIST			List type
			Disabled	0	Disable EasyDMA list
			ArrayList	1	Use array list

7.37.9.36 TXD.PTR

Address offset: 0x544

TXD Data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			TXD Data pointer See the memory chapter for details about which memories are available for EasyDMA.

7.37.9.37 TXD.MAXCNT

Address offset: 0x548

Maximum number of bytes in TXD buffer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in TXD buffer

7.37.9.38 TXD.AMOUNT

Address offset: 0x54C

Number of bytes transferred in the last TXD transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transferred in the last TXD transaction

7.37.9.39 TXD.LIST

Address offset: 0x550

EasyDMA list type

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	LIST			List type
			Disabled	0	Disable EasyDMA list
			ArrayList	1	Use array list

7.37.9.40 ADDRESS[n] (n=0..1)

Address offset: 0x588 + (n × 0x4)

TWI slave address n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	ADDRESS			TWI slave address

7.37.9.41 CONFIG

Address offset: 0x594

Configuration register for the address match mechanism

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A-B	RW	ADDRESS[i] (i=0..1)			Enable or disable address matching on ADDRESS[i]
			Disabled	0	Disabled
			Enabled	1	Enabled

7.37.9.42 ORC

Address offset: 0x5C0

Over-read character. Character sent out in case of an over-read of the transmit buffer.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	ORC			Over-read character. Character sent out in case of an over-read of the transmit buffer.

7.37.10 Electrical specification

7.37.10.1 TWIS slave timing specifications

Symbol	Description	Min.	Typ.	Max.	Units
$f_{TWIS,SCL}$	Bit rates for TWIS[29]	100		400	kbps
$t_{TWIS,START}$	Time from PREPARERX/PREPARETX task to ready to receive/ transmit		1.5		µs
t_{TWIS,SU_DAT}	Data setup time before positive edge on SCL – all modes	20			ns
t_{TWIS,HD_DAT}	Data hold time after negative edge on SCL – all modes	350		600	ns
$t_{TWIS,HD\_STA,100kbps}$	TWI slave hold time from for START condition (SDA low to SCL low), 100 kbps	500			ns
$t_{TWIS,HD\_STA,400kbps}$	TWI slave hold time from for START condition (SDA low to SCL low), 400 kbps	500			ns
$t_{TWIS,SU\_STO,100kbps}$	TWI slave setup time from SCL high to STOP condition, 100 kbps	500			ns

[29] High bit rates or stronger pull-ups may require GPIOs to be set as High Drive, see GPIO chapter for more details.

Symbol	Description	Min.	Typ.	Max.	Units
$t_{TWIS,SU\_STO,400kbps}$	TWI slave setup time from SCL high to STOP condition, 400 kbps	500			ns
$t_{TWIS,BUF,100kbps}$	TWI slave bus free time between STOP and START conditions, 100 kbps	500			ns
$t_{TWIS,BUF,400kbps}$	TWI slave bus free time between STOP and START conditions, 400 kbps	500			ns

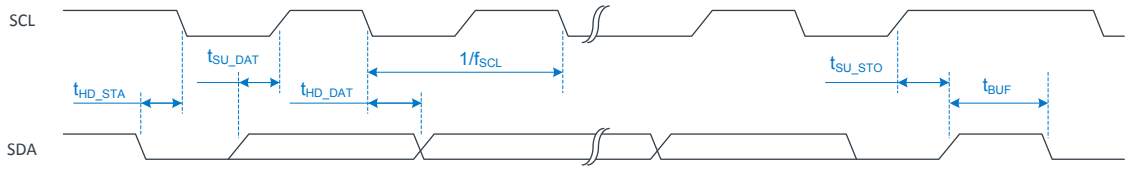


Figure 228: TWIS timing diagram, 1 byte transaction

7.38 UARTE — Universal asynchronous receiver/transmitter with EasyDMA

The Universal asynchronous receiver/transmitter with EasyDMA (UARTE) offers fast, full-duplex, asynchronous serial communication. Built-in flow control (CTS, RTS) is supported in hardware at a rate up to 1 Mbps and EasyDMA data transfer to and from RAM.

The main features of UARTE are the following:

- Full-duplex operation
- Automatic hardware flow control
- Optional even parity bit checking and generation
- EasyDMA
- Up to 1 Mbps baudrate
- Return to IDLE between transactions supported (when using HW flow control)
- One or two stop bit
- Least significant bit (LSB) first

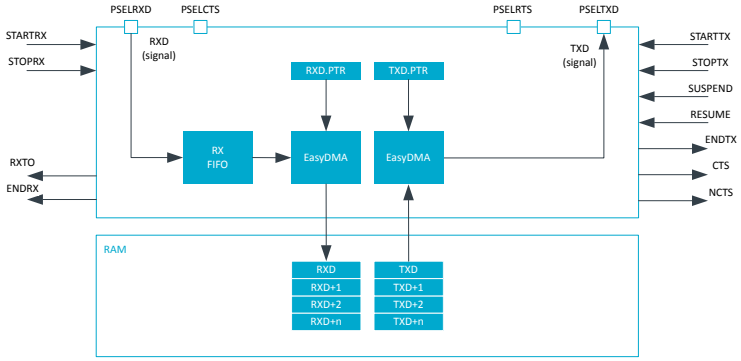


Figure 229: UARTE configuration

The GPIOs used for each UART interface can be chosen from any GPIO on the device and are independently configurable. This enables device pinout flexibility and efficient use of board space and signal routing.

Note: The external crystal oscillator must be enabled to obtain sufficient clock accuracy for stable communication. See CLOCK — Clock control on page 69 for more information.

7.38.1 EasyDMA

UARTE implements EasyDMA for reading and writing to and from the RAM.

If the TXD.PTR and the RXD.PTR are not pointing to the Data RAM region, an EasyDMA transfer may result in a HardFault or RAM corruption. See Memory on page 18 for more information about the different memory regions.

The RXD.PTR, TXD.PTR, RXD.MAXCNT, and TXD.MAXCNT registers are double-buffered. They can be updated and prepared for the next reception or transmission immediately after having received the RXSTARTED or TXSTARTED event.

The ENDRX and ENDTX events indicate that the EasyDMA is finished accessing the RX or TX buffer in RAM.

7.38.2 Transmission

The first step of a DMA transmission is storing bytes in the transmit buffer and configuring EasyDMA. This is achieved by writing the initial address pointer to TXD.PTR, and the number of bytes to transmit from the RAM buffer to TXD.MAXCNT. The UARTE transmission is started by triggering the STARTTX task.

After each byte has been sent over the TXD line, a TXDRDY event will be generated.

When all bytes have been transmitted, the transmission will automatically end and the ENDTX event will be generated.

A UARTE transmission sequence is stopped by triggering the STOPTX task. A TXSTOPPED event will be generated when the UARTE transmitter has stopped.

If the ENDTX event has not already been generated when the UARTE transmitter has come to a stop, UARTE will generate the ENDTX event explicitly even though all bytes specified in the TXD.MAXCNT register have not been transmitted.

If flow control is enabled through the HWFC field in the CONFIG register, a transmission will be automatically suspended when CTS is deactivated and resumed when CTS is activated again, as shown in the following figure. A byte that is in transmission when CTS is deactivated will be fully transmitted before the transmission is suspended.

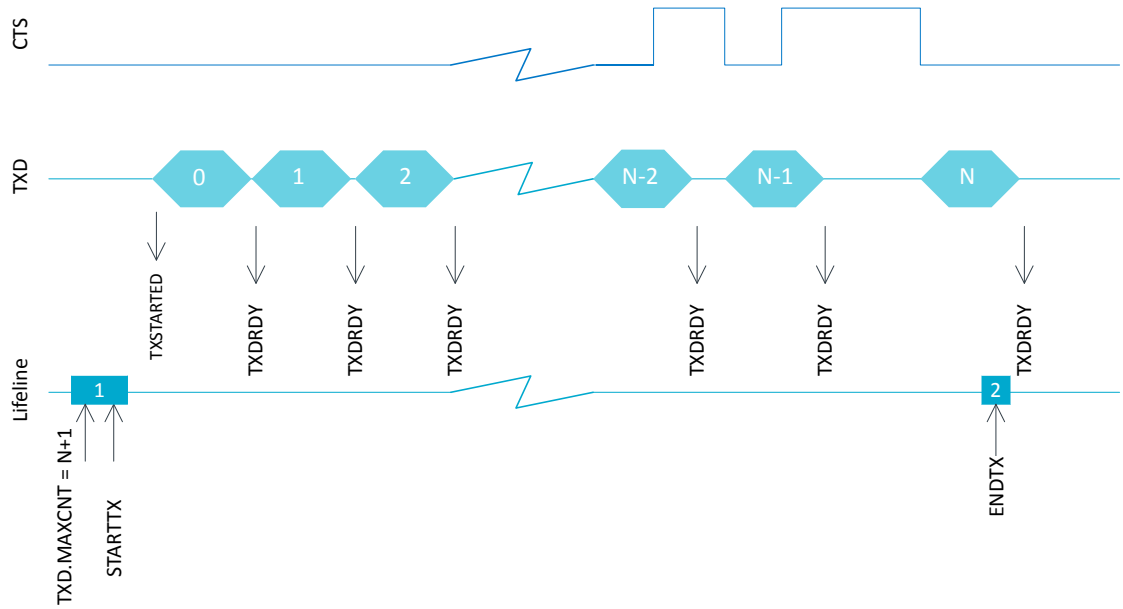


Figure 230: UARTE transmission

The UARTE transmitter is in its lowest activity level consuming the least amount of energy when it is stopped. That is, before it is started via STARTTX or after it has been stopped via STOPTX and the

TXSTOPPED event has been generated. See POWER — Power control on page 43 for more information about power modes.

7.38.3 Reception

The UARTE receiver is started by triggering the STARTRX task. The UARTE receiver uses EasyDMA to store incoming data in an RX buffer in RAM.

The RX buffer is located at the address specified in the RXD.PTR register. The RXD.PTR register is double-buffered and it can be updated and prepared for the next STARTRX task immediately after the RXSTARTED event is generated. The size of the RX buffer is specified in the RXD.MAXCNT register. UARTE generates an ENDRX event when it has filled up the RX buffer, as seen in the following figure.

For each byte received over the RXD line, an RXDRDY event is generated. This event is likely to occur before the corresponding data has been transferred to Data RAM.

The RXD.AMOUNT register can be queried following an ENDRX event to see how many new bytes have been transferred to the RX buffer in RAM since the previous ENDRX event.

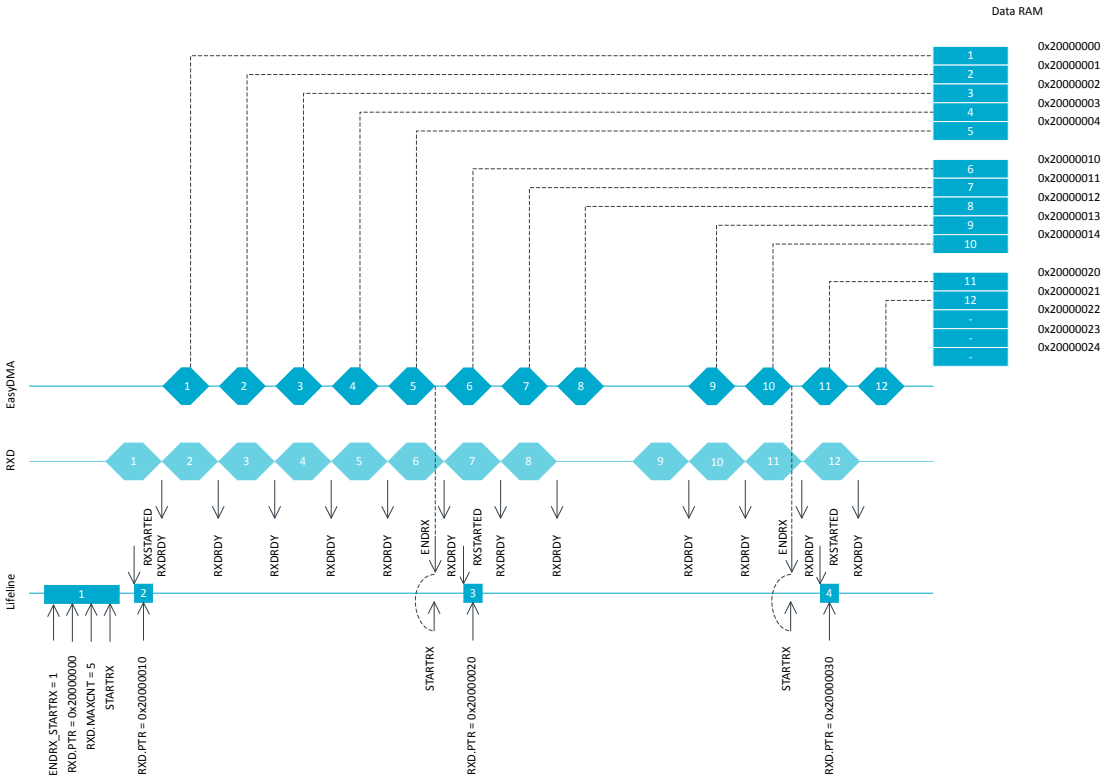


Figure 231: UARTE reception

The UARTE receiver is stopped by triggering the STOPRX task. An RXTO event is generated when the UARTE has stopped. UARTE makes sure that an impending ENDRX event is generated before the RXTO event is generated. This means that UARTE guarantees that no ENDRX event is generated after RXTO, unless UARTE is restarted or a FLUSHRX command is issued after the RXTO event is generated.

Note: If the ENDRX event has not been generated when the UARTE receiver stops, indicating that all pending content in the RX FIFO has been moved to the RX buffer, UARTE generates the ENDRX event explicitly even though the RX buffer is not full. In this scenario the ENDRX event is generated before the RXTO event is generated.

To determine the amount of bytes the RX buffer has received, the CPU can read the RXD.AMOUNT register following the ENDRX event or the RXTO event.

UARTE can receive up to four bytes after the STOPRX task has been triggered, if these are sent in succession immediately after the RTS signal is deactivated.

After the RXTO event is generated, the internal RX FIFO may still contain data. To move this data to RAM, the FLUSHRX task must be triggered. The RX buffer should be emptied, or the RXD.PTR register should be updated before the FLUSHRX task is triggered. This ensures the data in the RX buffer is not overwritten. To make sure that all data in the RX FIFO is moved to the RX buffer, the RXD.MAXCNT register must be set to RXD.MAXCNT > 4, as seen in the following figure. The UARTE will generate the ENDRX event after completing the FLUSHRX task even if the RX FIFO was empty or if the RX buffer does not fill up. After the ENDRX event, the RXD.AMOUNT register holds the actual amount of bytes transferred to the RX buffer.

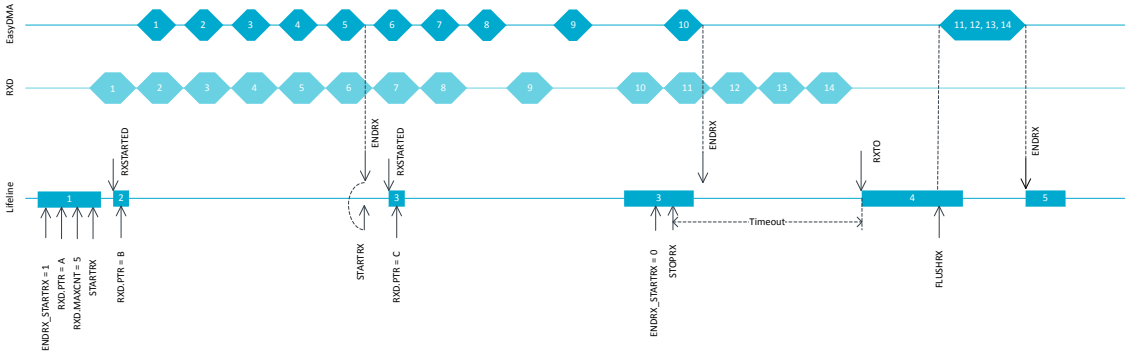


Figure 232: UARTE reception with forced stop via STOPRX

If hardware flow control is enabled through the HWFC field in the CONFIG register, the RTS signal will be deactivated when the receiver is stopped via the STOPRX task or when the UARTE is only able to receive four more bytes in its internal RX FIFO.

With flow control disabled, the UARTE will function in the same way as when the flow control is enabled except that the RTS line will not be used. This means that no signal will be generated when the UARTE has reached the point where it is only able to receive four more bytes in its internal RX FIFO. Data received when the internal RX FIFO is filled up, will be lost.

The UARTE receiver will be in its lowest activity level, and consume the least amount of energy, when it is stopped, i.e. before it is started via STARTRX or after it has been stopped via STOPRX and the RXTO event has been generated. See POWER — Power control on page 43 for more information about power modes.

7.38.4 Error conditions

An ERROR event, in the form of a framing error, will be generated if a valid stop bit is not detected in a frame. Another ERROR event, in the form of a break condition, will be generated if the RXD line is held active low for longer than the length of a data frame. Effectively, a framing error is always generated before a break condition occurs.

An ERROR event will not stop reception. If the error was a parity error, the received byte is still transferred into Data RAM along with any following bytes. If a framing error occurs (wrong stop bit), that byte will not be stored into Data RAM but following incoming bytes will.

7.38.5 Using the UARTE without flow control

If flow control is not enabled, the interface will behave as if the CTS and RTS lines are kept active all the time.

7.38.6 Parity and stop bit configuration

Automatic even parity generation for both transmission and reception can be configured using the register CONFIG on page 691. If odd parity is desired, it can be configured using the register CONFIG on page 691. See the register description for details.

The amount of stop bits can also be configured through the register CONFIG on page 691.

7.38.7 Low power

To ensure lowest possible power consumption when the peripheral is not needed, stop and disable UARTE.

The STOPTX and STOPRX tasks may not be always needed (the peripheral might already be stopped), but if STOPTX and/or STOPRX is sent, software shall wait until the TXSTOPPED and/or RXTO event is received in response, before disabling the peripheral through the ENABLE register.

7.38.8 Pin configuration

The different signals RXD, CTS (Clear To Send, active low), RTS (Request To Send, active low), and TXD associated with UARTE are mapped to physical pins according to the configuration specified in the PSEL.RXD, PSEL.CTS, PSEL.RTS, and PSEL.TXD registers respectively.

The PSEL.RXD, PSEL.CTS, PSEL.RTS, and PSEL.TXD registers and their configurations are only used as long as the UARTE is enabled, and retained only for the duration the device is in System ON mode. PSEL.RXD, PSEL.RTS, PSEL.RTS, and PSEL.TXD must only be configured when the UARTE is disabled.

To secure correct signal levels on the pins by the UARTE when in System OFF mode, the pins must be configured in the GPIO peripheral as described in the following table.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

UARTE signal	UARTE pin	Direction	Output value
RXD	As specified in PSEL.RXD	Input	Not applicable
CTS	As specified in PSEL.CTS	Input	Not applicable
RTS	As specified in PSEL.RTS	Output	1
TXD	As specified in PSEL.TXD	Output	1

Table 174: GPIO configuration before enabling peripheral

7.38.9 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50008000 0x40008000	APPLICATION	UARTE	UARTE0 : S UARTE0 : NS	US	SA	Universal asynchronous receiver/transmitter with EasyDMA 0	
0x50009000 0x40009000	APPLICATION	UARTE	UARTE1 : S UARTE1 : NS	US	SA	Universal asynchronous receiver/transmitter with EasyDMA 1	
0x5000B000 0x4000B000	APPLICATION	UARTE	UARTE2 : S UARTE2 : NS	US	SA	Universal asynchronous receiver/transmitter with EasyDMA 2	
0x5000C000 0x4000C000	APPLICATION	UARTE	UARTE3 : S UARTE3 : NS	US	SA	Universal asynchronous receiver/transmitter with EasyDMA 3	
0x41013000	NETWORK	UARTE	UARTE0	NS	NA	Universal asynchronous receiver/transmitter	

Table 175: Instances

Register	Offset	Security	Description
TASKS_STARTRX	0x000		Start UART receiver
TASKS_STOPRX	0x004		Stop UART receiver
TASKS_STARTTX	0x008		Start UART transmitter
TASKS_STOPTX	0x00C		Stop UART transmitter
TASKS_FLUSHRX	0x02C		Flush RX FIFO into RX buffer
SUBSCRIBE_STARTRX	0x080		Subscribe configuration for task STARTRX
SUBSCRIBE_STOPRX	0x084		Subscribe configuration for task STOPRX
SUBSCRIBE_STARTTX	0x088		Subscribe configuration for task STARTTX
SUBSCRIBE_STOPTX	0x08C		Subscribe configuration for task STOPTX
SUBSCRIBE_FLUSHRX	0x0AC		Subscribe configuration for task FLUSHRX
EVENTS_CTS	0x100		CTS is activated (set low). Clear To Send.
EVENTS_NCTS	0x104		CTS is deactivated (set high). Not Clear To Send.
EVENTS_RXDRDY	0x108		Data received in RXD (but potentially not yet transferred to Data RAM)
EVENTS_ENDRX	0x110		Receive buffer is filled up
EVENTS_TXDRDY	0x11C		Data sent from TXD
EVENTS_ENDTX	0x120		Last TX byte transmitted
EVENTS_ERROR	0x124		Error detected
EVENTS_RXTO	0x144		Receiver timeout
EVENTS_RXSTARTED	0x14C		UART receiver has started
EVENTS_TXSTARTED	0x150		UART transmitter has started
EVENTS_TXSTOPPED	0x158		Transmitter stopped
PUBLISH_CTS	0x180		Publish configuration for event CTS
PUBLISH_NCTS	0x184		Publish configuration for event NCTS
PUBLISH_RXDRDY	0x188		Publish configuration for event RXDRDY
PUBLISH_ENDRX	0x190		Publish configuration for event ENDRX
PUBLISH_TXDRDY	0x19C		Publish configuration for event TXDRDY
PUBLISH_ENDTX	0x1A0		Publish configuration for event ENDTX
PUBLISH_ERROR	0x1A4		Publish configuration for event ERROR
PUBLISH_RXTO	0x1C4		Publish configuration for event RXTO
PUBLISH_RXSTARTED	0x1CC		Publish configuration for event RXSTARTED
PUBLISH_TXSTARTED	0x1D0		Publish configuration for event TXSTARTED
PUBLISH_TXSTOPPED	0x1D8		Publish configuration for event TXSTOPPED
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ERRORSRC	0x480		Error source
ENABLE	0x500		Enable UART
PSEL.RTS	0x508		Pin select for RTS signal
PSEL.TXD	0x50C		Pin select for TXD signal
PSEL.CTS	0x510		Pin select for CTS signal
PSEL.RXD	0x514		Pin select for RXD signal
BAUDRATE	0x524		Baud rate. Accuracy depends on the HFCLK source selected.
RXD.PTR	0x534		Data pointer
RXD.MAXCNT	0x538		Maximum number of bytes in receive buffer
RXD.AMOUNT	0x53C		Number of bytes transferred in the last transaction
TXD.PTR	0x544		Data pointer
TXD.MAXCNT	0x548		Maximum number of bytes in transmit buffer
TXD.AMOUNT	0x54C		Number of bytes transferred in the last transaction
CONFIG	0x56C		Configuration of parity and hardware flow control

Table 176: Register overview

7.38.9.1 TASKS_STARTRX

Address offset: 0x000

Start UART receiver

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A
Reset 0x00000000						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description	
A	W	TASKS_STARTRX			Start UART receiver	
			Trigger	1	Trigger task	

7.38.9.2 TASKS_STOPRX

Address offset: 0x004

Stop UART receiver

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A
Reset 0x00000000						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description	
A	W	TASKS_STOPRX			Stop UART receiver	
			Trigger	1	Trigger task	

7.38.9.3 TASKS_STARTTX

Address offset: 0x008

Start UART transmitter

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A
Reset 0x00000000						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description	
A	W	TASKS_STARTTX			Start UART transmitter	
			Trigger	1	Trigger task	

7.38.9.4 TASKS_STOPTX

Address offset: 0x00C

Stop UART transmitter

Bit number						31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID						A
Reset 0x00000000						**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description	
A	W	TASKS_STOPTX			Stop UART transmitter	
			Trigger	1	Trigger task	

7.38.9.5 TASKS_FLUSHRX

Address offset: 0x02C

Flush RX FIFO into RX buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_FLUSHRX			Flush RX FIFO into RX buffer
			Trigger	1	Trigger task

7.38.9.6 SUBSCRIBE_STARTRX

Address offset: 0x080

Subscribe configuration for task STARTRX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STARTRX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.38.9.7 SUBSCRIBE_STOPRX

Address offset: 0x084

Subscribe configuration for task STOPRX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOPRX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.38.9.8 SUBSCRIBE_STARTTX

Address offset: 0x088

Subscribe configuration for task STARTTX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STARTTX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.38.9.9 SUBSCRIBE_STOPTX

Address offset: 0x08C

Subscribe configuration for task STOPTX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOPTX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.38.9.10 SUBSCRIBE_FLUSHRX

Address offset: 0x0AC

Subscribe configuration for task FLUSHRX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task FLUSHRX will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.38.9.11 EVENTS_CTS

Address offset: 0x100

CTS is activated (set low). Clear To Send.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_CTS			CTS is activated (set low). Clear To Send.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.12 EVENTS_NCTS

Address offset: 0x104

CTS is deactivated (set high). Not Clear To Send.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_NCTS			CTS is deactivated (set high). Not Clear To Send.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.13 EVENTS_RXDRDY

Address offset: 0x108

Data received in RXD (but potentially not yet transferred to Data RAM)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RXDRDY			Data received in RXD (but potentially not yet transferred to Data RAM)
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.14 EVENTS_ENDRX

Address offset: 0x110

Receive buffer is filled up

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDRX			Receive buffer is filled up
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.15 EVENTS_TXDRDY

Address offset: 0x11C

Data sent from TXD

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_TXDRDY			Data sent from TXD
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.16 EVENTS_ENDTX

Address offset: 0x120

Last TX byte transmitted

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDTX			Last TX byte transmitted
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.17 EVENTS_ERROR

Address offset: 0x124

Error detected

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ERROR			Error detected
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.18 EVENTS_RXTO

Address offset: 0x144

Receiver timeout

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RXTO			Receiver timeout
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.19 EVENTS_RXSTARTED

Address offset: 0x14C

UART receiver has started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_RXSTARTED			UART receiver has started
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.20 EVENTS_TXSTARTED

Address offset: 0x150

UART transmitter has started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_TXSTARTED			UART transmitter has started
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.21 EVENTS_TXSTOPPED

Address offset: 0x158

Transmitter stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_TXSTOPPED			Transmitter stopped
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.38.9.22 PUBLISH_CTS

Address offset: 0x180

Publish configuration for event CTS

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event CTS will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.23 PUBLISH_NCTS

Address offset: 0x184

Publish configuration for event NCTS

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event NCTS will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.24 PUBLISH_RXDRDY

Address offset: 0x188

Publish configuration for event RXDRDY

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RXDRDY will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.25 PUBLISH_ENDRX

Address offset: 0x190

Publish configuration for event ENDRX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDRX will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.26 PUBLISH_TXDRDY

Address offset: 0x19C

Publish configuration for event TXDRDY

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event TXDRDY will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.27 PUBLISH_ENDTX

Address offset: 0x1A0

Publish configuration for event ENDTX

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDTX will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.28 PUBLISH_ERROR

Address offset: 0x1A4

Publish configuration for event ERROR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ERROR will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.29 PUBLISH_RXTO

Address offset: 0x1C4

Publish configuration for event RXTO

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RXTO will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.30 PUBLISH_RXSTARTED

Address offset: 0x1CC

Publish configuration for event RXSTARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event RXSTARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.31 PUBLISH_TXSTARTED

Address offset: 0x1D0

Publish configuration for event TXSTARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event TXSTARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.32 PUBLISH_TXSTOPPED

Address offset: 0x1D8

Publish configuration for event TXSTOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event TXSTOPPED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.38.9.33 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				D C
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
C	RW	ENDRX_STARTRX			Shortcut between event ENDRX and task STARTRX
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
D	RW	ENDRX_STOPRX			Shortcut between event ENDRX and task STOPRX
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.38.9.34 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				L J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CTS			Enable or disable interrupt for event CTS
			Disabled	0	Disable
			Enabled	1	Enable
B	RW	NCTS			Enable or disable interrupt for event NCTS
			Disabled	0	Disable
			Enabled	1	Enable
C	RW	RXDRDY			Enable or disable interrupt for event RXDRDY
			Disabled	0	Disable
			Enabled	1	Enable
D	RW	ENDRX			Enable or disable interrupt for event ENDRX
			Disabled	0	Disable
			Enabled	1	Enable
E	RW	TXDRDY			Enable or disable interrupt for event TXDRDY
			Disabled	0	Disable
			Enabled	1	Enable
F	RW	ENDTX			Enable or disable interrupt for event ENDTX
			Disabled	0	Disable
			Enabled	1	Enable
G	RW	ERROR			Enable or disable interrupt for event ERROR
			Disabled	0	Disable
			Enabled	1	Enable
H	RW	RXTO			Enable or disable interrupt for event RXTO
			Disabled	0	Disable
			Enabled	1	Enable
I	RW	RXSTARTED			Enable or disable interrupt for event RXSTARTED
			Disabled	0	Disable
			Enabled	1	Enable
J	RW	TXSTARTED			Enable or disable interrupt for event TXSTARTED
			Disabled	0	Disable
			Enabled	1	Enable
L	RW	TXSTOPPED			Enable or disable interrupt for event TXSTOPPED
			Disabled	0	Disable
			Enabled	1	Enable

7.38.9.35 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				L J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CTS			Write '1' to enable interrupt for event CTS
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	NCTS			Write '1' to enable interrupt for event NCTS
			Set	1	Enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				L J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	RXDRDY			Write '1' to enable interrupt for event RXDRDY
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	ENDRX			Write '1' to enable interrupt for event ENDRX
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	TXDRDY			Write '1' to enable interrupt for event TXDRDY
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	ENDTX			Write '1' to enable interrupt for event ENDTX
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	ERROR			Write '1' to enable interrupt for event ERROR
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	RXTO			Write '1' to enable interrupt for event RXTO
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
I	RW	RXSTARTED			Write '1' to enable interrupt for event RXSTARTED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
J	RW	TXSTARTED			Write '1' to enable interrupt for event TXSTARTED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
L	RW	TXSTOPPED			Write '1' to enable interrupt for event TXSTOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.38.9.36 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				L J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CTS			Write '1' to disable interrupt for event CTS

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				L J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	NCTS			Write '1' to disable interrupt for event NCTS
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C	RW	RXDRDY			Write '1' to disable interrupt for event RXDRDY
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
D	RW	ENDRX			Write '1' to disable interrupt for event ENDRX
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
E	RW	TXDRDY			Write '1' to disable interrupt for event TXDRDY
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
F	RW	ENDTX			Write '1' to disable interrupt for event ENDTX
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
G	RW	ERROR			Write '1' to disable interrupt for event ERROR
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
H	RW	RXTO			Write '1' to disable interrupt for event RXTO
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
I	RW	RXSTARTED			Write '1' to disable interrupt for event RXSTARTED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
J	RW	TXSTARTED			Write '1' to disable interrupt for event TXSTARTED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
L	RW	TXSTOPPED			Write '1' to disable interrupt for event TXSTOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.38.9.37 ERRORSRC

Address offset: 0x480

Error source

This register is read/write one to clear.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	OVERRUN			Overrun error A start bit is received while the previous data still lies in RXD. (Previous data is lost.)
			NotPresent	0	Read: error not present
			Present	1	Read: error present
B	RW	PARITY			Parity error A character with bad parity is received, if HW parity check is enabled.
			NotPresent	0	Read: error not present
			Present	1	Read: error present
C	RW	FRAMING			Framing error occurred A valid stop bit is not detected on the serial data input after all bits in a character have been received.
			NotPresent	0	Read: error not present
			Present	1	Read: error present
D	RW	BREAK			Break condition The serial data input is '0' for longer than the length of a data frame. (The data frame length is 10 bits without parity bit and 11 bits with parity bit.)
			NotPresent	0	Read: error not present
			Present	1	Read: error present

7.38.9.38 ENABLE

Address offset: 0x500

Enable UART

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ENABLE			Enable or disable UARTE
			Disabled	0	Disable UARTE
			Enabled	8	Enable UARTE

7.38.9.39 PSEL.RTS

Address offset: 0x508

Pin select for RTS signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.38.9.40 PSEL.TXD

Address offset: 0x50C

Pin select for TXD signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.38.9.41 PSEL.CTS

Address offset: 0x510

Pin select for CTS signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.38.9.42 PSEL.RXD

Address offset: 0x514

Pin select for RXD signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN		[0..31]	Pin number
B	RW	PORT		[0..1]	Port number
C	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

7.38.9.43 BAUDRATE

Address offset: 0x524

Baud rate. Accuracy depends on the HFCLK source selected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x04000000				**0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	BAUDRATE			Baud rate
			Baud1200	0x0004F000	1200 baud (actual rate: 1205)
			Baud2400	0x0009D000	2400 baud (actual rate: 2396)
			Baud4800	0x0013B000	4800 baud (actual rate: 4808)
			Baud9600	0x00275000	9600 baud (actual rate: 9598)
			Baud14400	0x003AF000	14400 baud (actual rate: 14401)
			Baud19200	0x004EA000	19200 baud (actual rate: 19208)
			Baud28800	0x0075C000	28800 baud (actual rate: 28777)
			Baud31250	0x00800000	31250 baud
			Baud38400	0x009D0000	38400 baud (actual rate: 38369)
			Baud56000	0x00E50000	56000 baud (actual rate: 55944)
			Baud57600	0x00EB0000	57600 baud (actual rate: 57554)
			Baud76800	0x013A9000	76800 baud (actual rate: 76923)
			Baud115200	0x01D60000	115200 baud (actual rate: 115108)
			Baud230400	0x03B00000	230400 baud (actual rate: 231884)
			Baud250000	0x04000000	250000 baud
			Baud460800	0x07400000	460800 baud (actual rate: 457143)
			Baud921600	0x0F000000	921600 baud (actual rate: 941176)
			Baud1M	0x10000000	1 megabaud

7.38.9.44 RXD.PTR

Address offset: 0x534

Data pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.38.9.45 RXD.MAXCNT

Address offset: 0x538

Maximum number of bytes in receive buffer

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in receive buffer

7.38.9.46 RXD.AMOUNT

Address offset: 0x53C

Number of bytes transferred in the last transaction

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transferred in the last transaction

7.38.9.47 TXD.PTR

Address offset: 0x544

Data pointer

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the Memory chapter for details about which memories are available for EasyDMA.

7.38.9.48 TXD.MAXCNT

Address offset: 0x548

Maximum number of bytes in transmit buffer

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1..0xFFFF]	Maximum number of bytes in transmit buffer

7.38.9.49 TXD.AMOUNT

Address offset: 0x54C

Number of bytes transferred in the last transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT		[1..0xFFFF]	Number of bytes transferred in the last transaction

7.38.9.50 CONFIG

Address offset: 0x56C

Configuration of parity and hardware flow control

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				D C B B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	HWFC			Hardware flow control
			Disabled	0	Disabled
			Enabled	1	Enabled
B	RW	PARITY			Parity
			Excluded	0x0	Exclude parity bit
			Included	0x7	Include even parity bit
C	RW	STOP			Stop bits
			One	0	One stop bit
			Two	1	Two stop bits
D	RW	PARITYTYPE			Even or odd parity type
			Even	0	Even parity
			Odd	1	Odd parity

7.38.10 Electrical specification

7.38.10.1 UARTE electrical specification

Symbol	Description	Min.	Typ.	Max.	Units
f_{UARTE}	Baud rate for UARTE[30].			1000	kbps
$t_{UARTE,CTSH}$	CTS high time	0.5			µs
$t_{UARTE,START}$	Time from STARTRX/STARTTX task to transmission started		0.25		µs

7.39 USBD — Universal serial bus device

The USB device (USBD) controller implements a full speed USB device function that meets 2.0 revision of the USB specification.

[30] High baud rates may require GPIOs to be set as High Drive, see GPIO chapter for more details.

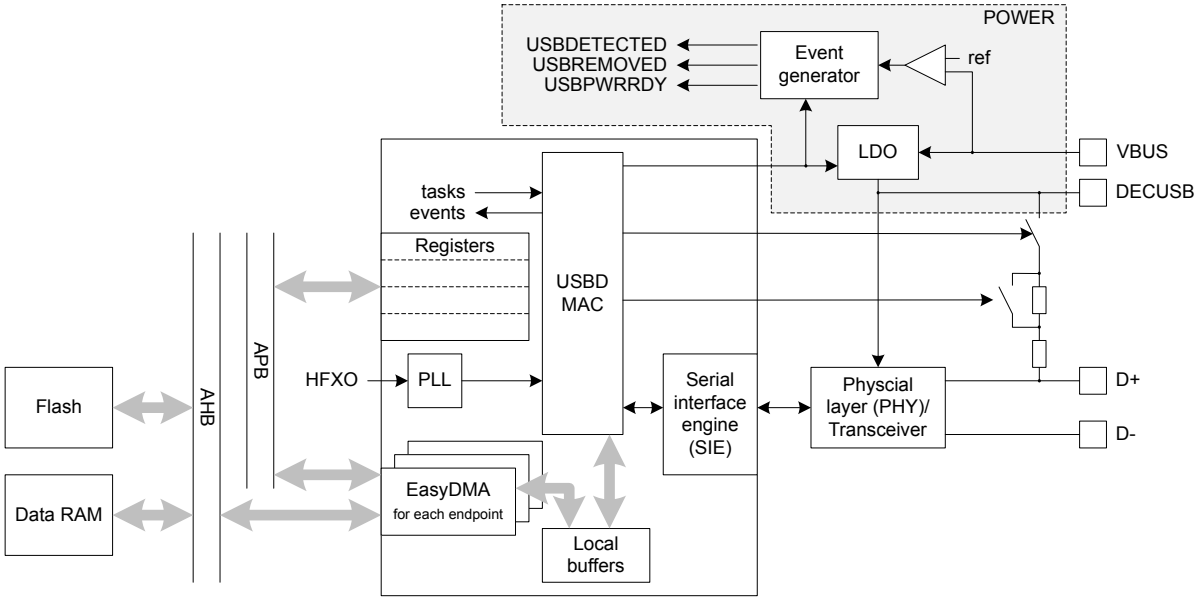


Figure 233: USB device block diagram

Listed here are the main features for USBD:

- Implements full-speed (12 Mbps) device fully compliant to Universal Serial Bus Specification Revision 2.0, including following engineering change notices (ECNs) issued by USB Implementers Forum:
  - *Pull-up/pull-down Resistors ECN*
  - *5V Short Circuit Withstand Requirement Change ECN*
- USB device stack available in the Nordic SDK
- Integrated (on-chip) USB transceiver (PHY)
- Software controlled on-chip pull-up on D+
- Endpoints:
  - 2 control (1 IN, 1 OUT)
  - 14 bulk/interrupt (7 IN, 7 OUT)
  - 2 isochronous (1 IN, 1 OUT)
- Supports double buffering for isochronous (ISO) endpoints (IN/OUT)
- Supports USB suspend, resume, and remote wake-up
- 64 bytes buffer size for each bulk/interrupt endpoint
- Up to 1023 bytes buffer size for ISO endpoints
- EasyDMA for all data transfers

7.39.1 USB device states

The behavior of a USB device can be modelled through a state diagram.

The *USB 2.0 Specification* (see *Chapter 9 USB Device Framework*) defines a number of states for a USB device, as shown in the following figure.

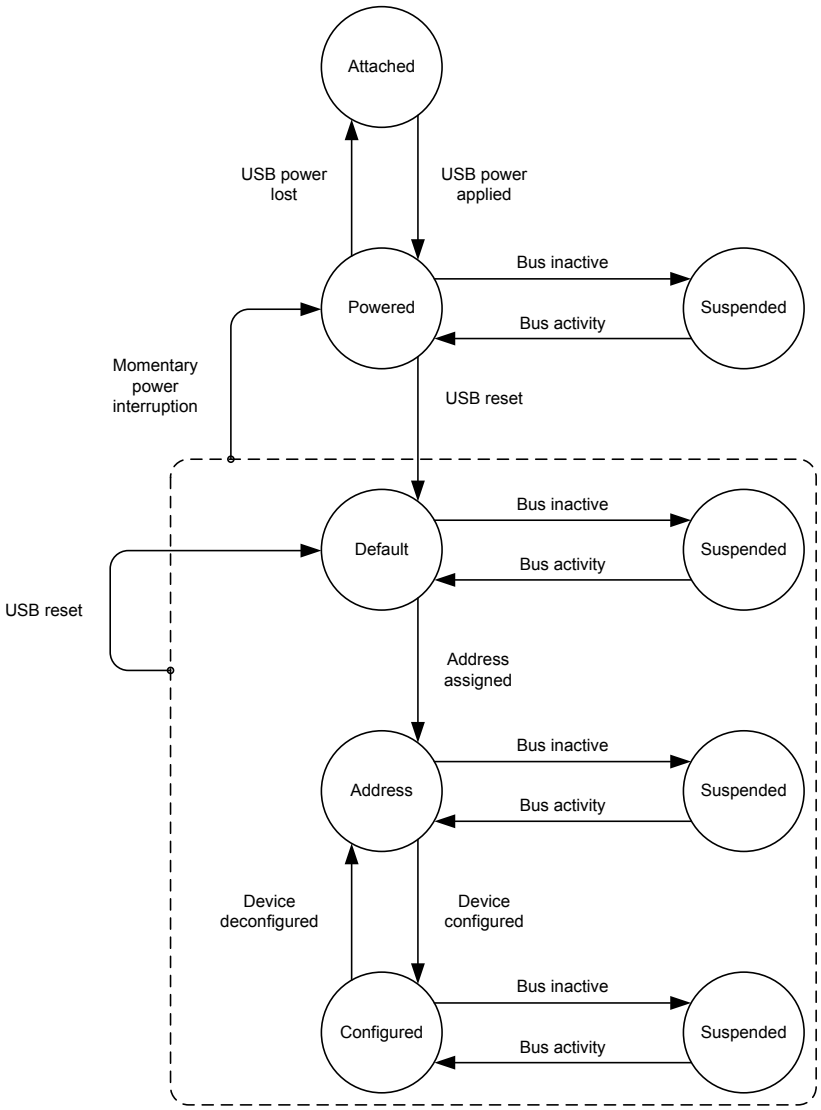


Figure 234: Device state diagram

The device must change state according to host-initiated traffic and USB bus states. It is up to the software to implement a state machine that matches the above definition. To detect the presence or absence of USB supply (VBUS), two events USBDETECTED and USBREMOVED can be used to implement the state machine. For more details on these events, see USBREG — USB regulator control on page 55.

As a general rule when implementing the software, the host behavior shall never be assumed to be predictable. In particular the sequence of commands received during an enumeration. The software shall always react to the current bus conditions or commands sent by the host.

7.39.2 USB terminology

The USB specification defines bus states, rather than logic levels on the D+ and D- lines.

For a full speed device, the bus state where the D+ line is high and the D- line is low is defined as the J state. The bus state where D+ is low and D- high is called the K state.

An idle bus, where D+ and D- lines are only polarized through the pull-up on D+ and pull-downs on the host side, will be in J state.

Both lines low are called SE0 (single-ended 0), and both lines high SE1 (single-ended 1).

7.39.3 USB pins

The USBD peripheral features a number of dedicated pins.

The dedicated USB pins can be grouped in two categories, signal and power. The signal pins consist of the D+ and D- pins, which are to be connected to the USB host. They are dedicated pins, and not available as standard GPIOs. The USBD peripheral is implemented according to the USB specification revision 2.0, *5V Short Circuit Withstand ECN Requirement Change*, meaning these two pins are not 5 V tolerant.

The signal pins and the pull-up will operate only while VBUS is in its valid voltage range, and USBD is enabled through the ENABLE register. For details on the USB power supply and VBUS detection, see USBREG — USB regulator control on page 55.

For more information about the pinout, see Pin assignments on page 783.

7.39.4 USBD power-up sequence

The physical layer interface (PHY)/USB transceiver is powered separately from the rest of the device (VBUS pin), which has some implications on the USBD power-up sequence.

The device is not able to properly signal its presence to the USB host and handle traffic from the host, unless the PHY's power supply is enabled and stable. Turning the PHY's power supply on/off is directly linked to register ENABLE. The device provides events that help synchronizing software to the various steps during the power-up sequence.

To make sure that all resources in USBD are available and the dedicated USB voltage regulator stabilized, the following is recommended:

- Enable USBD only after VBUS has been detected
- Turn the USB pull-up on after the following events have occurred:
  - USBPWRRDY
  - USBEVENT, with the READY condition flagged in EVENTCAUSE

The following sequence chart illustrates a typical handling of VBUS power-up:

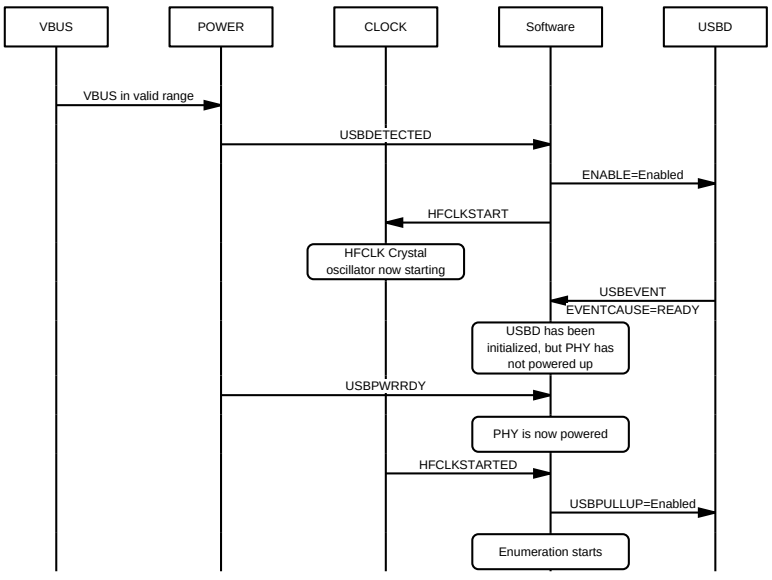


Figure 235: VBUS power-up sequence

Upon detecting VBUS removal, it is recommended to wait for ongoing EasyDMA transfers to finish before disabling USBD (relevant ENDEPIN[n], ENDISOIN, ENDEPOUT[n], or ENDISOOUT events, see EasyDMA on page 697). The USBREMOVED event, described in USBREG — USB regulator control on page 55, signals when the VBUS is removed. Reading the ENABLE register will return Enabled until USBD is completely disabled.

7.39.5 USB pull-up

The USB pull-up serves two purposes: it indicates to the host that the device is connected to the USB bus, and it indicates the device's speed capability.

When no pull-up is connected to the USB bus, the host sees both D+ and D- lines low, as they are pulled down on the host side by 15 kΩ resistors. The device is not detected by the host, putting it in a detached state even if it is physically connected to the host. In this situation, the device is not allowed to draw current from VBUS, according to *USB 2.0 Specification*.

When a full-speed device connects its 1.5 kΩ pull-up to D+, the host sees the corresponding line high. The device is then in the attached state. During the enumeration process, the host attempts to determine if the full-speed device also supports higher speeds and initiates communication with the device to further identify it. The USBD peripheral implemented in this device supports only full-speed operation (12 Mbps), and thus ignores the negotiation for higher speeds in accordance with *USB 2.0 Specification*.

Register USBPULLUP enables software to connect or disconnect the pull-up on D+. This allows the software to control when USB enumeration takes place. It also allows to emulate a physical disconnect from the USB bus, for instance when re-enumeration is required. USBPULLUP has to be enabled to allow the USBD to handle USB traffic and generate appropriate events. This forbids the use of an external pull-up.

Note that disconnecting the pull-up through register USBPULLUP while connected to a host, will result in both D+ and D- lines to be pulled low by the host's pull-down resistors. However, as mentioned above, this will also inhibit the generation of the USBRESET event. The pull-up is disabled by default after a chip reset.

The pull-up shall only get connected after USBD has been enabled through register ENABLE. The USB pull-up value is automatically changed depending on the bus activity, as specified in *Resistor ECN* which amends the original *USB 2.0 Specification*. The user does not have access to this function as it is handled in hardware.

While they should never be used in normal traffic activity, lines D+ and D- may at any time be forced into state specified in register DPDMVALUE by the task DPDMDRIVE. The DPDMNODRIVE task stops driving them, and PHY returns to normal operation.

7.39.6 USB reset

The USB specification defines a USB reset, which is not be confused with a chip reset. The USB reset is a normal USB bus condition, and is used as part of the enumeration sequence, it does not reset the chip.

The USB reset results from a single-ended low state (SE0) on lines D+/D- for a $t_{USB,DETRST}$ amount of time. Only the host is allowed to drive a USB reset condition on the bus. The UBSD peripheral automatically interprets a SE0 longer than $t_{USB,DETRST}$ as a USB reset. When the device detects a USB reset and generates a USBRESET event, the device USB stack and related parts of the application shall re-initialize themselves, and go back to the default state.

Some of the registers in the USBD peripheral get automatically reset to a known state, in particular all data endpoints are disabled and the USBADDR reset to 0.

After the device has connected to the USB bus (i.e. after VBUS is applied), the device shall not respond to any traffic from the time the pull-up is enabled until it has seen a USB reset condition. This is automatically ensured by the USBD.

After a USB reset, the device shall be fully responsive after at most T_{RSTRCY} (according to chapter 7 in the USB specification). Software shall take into account this time that takes the hardware to recover from a USB reset condition.

7.39.7 USB suspend and resume

Normally, the host will maintain activity on the USB at least every millisecond according to USB specification. A USB device will enter suspend when there is no activity on the bus (idle) for a given time. The device will resume operation when it receives any non idle signalling.

To signal that the device shall go into low power mode (suspend), the host stops activity on the USB bus, which becomes idle. Only the device pull-up and host pull-downs act on D+ and D-, and the bus is thus kept at a constant J state. It is up to the device to detect this lack of activity, and enter the low power mode (suspend) within a specified time.

The USB host can decide to suspend or resume USB activity at any time. If remote wake-up is enabled, the device may signal to the host to resume from suspend.

7.39.7.1 Entering suspend

The USBD peripheral automatically detects lack of activity for more than a defined amount of time, and performs steps needed to enter suspend.

When no activity has been detected for longer than $t_{USB,SUSPEND}$, the USBD generates the USBEVENT event with SUSPEND bit set in register EVENTCAUSE. The software shall ensure that the current drawn from the USB supply line VBUS is within the specified limits before T_{2SUSP}, as defined in chapter 7 of the USB specification. In order to reduce idle current of USBD, the software must explicitly place the USBD in low power mode through writing `LowPower` to register LOWPOWER.

In order to save power, and provided that no other peripheral needs it, the crystal oscillator (HFXO) in CLOCK may be disabled by software during the USB suspend, while the USB pull-up is disconnected, or when VBUS is not present. Software must explicitly enable it at any other time. The USBD will not be able to respond to USB traffic unless HFXO is enabled and stable.

7.39.7.2 Host-initiated resume

Once the host resumes the bus activity, it has to be responsive to incoming requests on the USB bus within the time T_{RSMRCY} (as defined in chapter 7 of the USB specification) and revert to normal power consumption mode.

If the host resumes bus activity with or without a RESUME condition (in other words: bus activity is defined as any non-J state), the USBD peripheral will generate a USBEVENT event, with RESUME bit set in register EVENTCAUSE. If the host resumes bus activity simply by restarting sending frames, the USBD peripheral will generate SOF events.

7.39.7.3 Device-initiated remote wake-up

Assuming the remote wake-up is supported by the device and enabled by the host, the device can request the host to resume from suspend if wake-up condition is met.

To do so, the HFXO needs to be enabled first. After waking up the HFXO, the software must bring USBD out of the low power mode and into the normal power consumption mode through writing `ForceNormal` in register LOWPOWER. It can then instruct the USBD peripheral to drive a RESUME condition (K state) on the USB bus by triggering the DPDMDRIVE task, and hence attempt to wake up the host. By choosing `Resume` in DPDMVALUE, the duration of the RESUME state is under hardware control ($t_{USB,DRIVEK}$). By choosing `J` or `K`, the duration of that state is under software control (the J or K state is maintained until a DPDMNODRIVE task is triggered) and has to meet T_{DRSMUP} as specified in USB specification chapter 7.

Upon writing the `ForceNormal` in register LOWPOWER, a USBEVENT event is generated with the USBWUALLOWED bit set in register EVENTCAUSE.

The value in register DPDMVALUE on page 729 will only be captured and used when the DPDMDRIVE task is triggered. This value defines the state the bus will be forced into after the DPDMDRIVE task.

The device shall ensure that it does not initiate a remote wake-up request before T_{WTRSM} (according to USB specification chapter 7) after the bus has entered idle state. Using the recommended resume value in DPDMVALUE (rather than K) takes care of this, and postpones the RESUME state accordingly.

7.39.8 EasyDMA

The USBD peripheral implements EasyDMA for accessing memory without CPU involvement.

Each endpoint has an associated set of registers, tasks and events. EasyDMA and traffic on USB are tightly related. A number of events provide insight of what is happening on the USB bus with a number of tasks allowing an automated response to the traffic.

> **Note:** Endpoint 0 (IN and OUT) are implemented as control endpoint. For more information, see Control transfers on page 698.

Registers

Enabling endpoints is controlled through the EPINEN and EPOUTEN registers.

The following registers define the memory address of the buffer for a specific IN or OUT endpoint:

- EPIN[n].PTR, (n=0..7)
- EPOUT[n].PTR, (n=0..7)
- ISOIN.PTR
- ISOOUT.PTR

The following registers define the amount of bytes to be sent on USB for next transaction:

- EPIN[n].MAXCNT, (n=0..7)
- ISOIN.MAXCNT

The following registers define the length of the buffer (in bytes) for next transfer of incoming data:

- EPOUT[n].MAXCNT, (n=1..7)
- ISOOUT.MAXCNT

Since the host decides how many bytes are sent over USB, the MAXCNT value can be copied from register SIZE.EPOUT[n] (n=1..7) or register SIZE.ISOOUT.

Register EPOUT[0].MAXCNT defines the length of the OUT buffer (in bytes) for the control endpoint 0. Register SIZE.EPOUT[0] shall indicate the same value as `MaxPacketSize` from the device descriptor or `wLength` from the SETUP command, whichever is the least.

The .AMOUNT registers indicate how many bytes actually have been transferred over EasyDMA during the last transfer.

Stalling bulk/interrupt endpoints is controlled through the EPSTALL register.

> **Note:** Due to USB specification requirements, the effect of the stalling control endpoint 0 may be overridden by hardware, in particular when a new SETUP token is received.

EasyDMA will not copy the SETUP data to memory (it will only transfer data from the data stage). The following are separate registers in the USBD peripheral that have setup data.

- BMREQUESTTYPE
- BREQUEST
- WVALUEL
- WVALUEH
- WINDEXL
- WINDEXH

- WLENGTHL
- WLENGTHH

The EVENTCAUSE register provides details on what caused a given USBEVENT event, for instance if a CRC error is detected during a transaction, or if bus activity stops or resumes.

Tasks

Tasks STARTEPIN[n], STARTEPOUT[n] (n=0..7), STARTISOIN, and STARTISOOUT capture the values for .PTR and .MAXCNT registers. For IN endpoints, a transaction over USB gets automatically triggered when the EasyDMA transfer is complete. For OUT endpoints, it is up to software to allow the next transaction over USB. See the examples in Control transfers on page 698, Bulk and interrupt transactions on page 701, and Isochronous transactions on page 703.

For the control endpoint 0, OUT transactions are allowed through the EP0RCVOUT task. The EP0STATUS task allows a status stage to be initiated, and the EP0STALL task allows stalling further traffic (data or status stage) on the control endpoint.

Events

The STARTED event confirms that the values of the .PTR and .MAXCNT registers of the endpoints flagged in register EPSTATUS have been captured. Those can then be modified by software for the next transfer.

Events ENDEPIN[n], ENDEPOUT[n] (n=0..7), ENDISOIN, and ENDISOOUT events indicate that the entire buffer has been consumed. The buffer can be accessed safely by the software.

Only a single EasyDMA transfer can take place in USBD at any time. Software must ensure that tasks STARTEPIN[n] (n=0..7), STARTISOIN , STARTEPOUT[n] (n=0..7), or STARTISOOUT are not triggered before events ENDEPIN[n] (n=0..7), ENDISOIN, ENDEPOUT[n] (n=0..7), or ENDISOOUT are received from an on-going transfer.

The EPDATA event indicates that a successful (acknowledged) data transaction has occurred on the data endpoint(s) flagged in register EPDATASTATUS. A successful (acknowledged) data transaction on endpoint 0 is signalled by the EP0DATADONE event.

At any time a USBEVENT event may be sent, with details provided in EVENTCAUSE register.

The EP0SETUP event indicates that a SETUP token has been received on the control endpoint 0, and that the setup data is available in #unique_1728/unique_1728_Connect_42_setup_data_registers on page 697.

7.39.9 Control transfers

The USB specification mandates every USB device to implement endpoint 0 IN and OUT as control endpoints.

A control transfer consists of two or three stages:

- Setup stage
- Data stage (optional)
- Status stage

Each control transfer can be one of following types:

- Control read
- Control read no data
- Control write
- Control write no data

An EP0SETUP event indicates that the data in the setup stage (following the SETUP token) is available in registers.

The data in the data stage (following the IN or OUT token) is transferred from or to the desired location using EasyDMA.

The control endpoint buffer can be of any size.

After receiving the SETUP token, the USB controller will not accept (NAK) any incoming IN or OUT tokens until the software has finished decoding the command, determined the type of transfer, and prepared for the next stage (data or status) appropriately.

The software can stall a command when in the data and status stages, through the EP0STALL task, when the command is not supported or if its wValue, wIndex or wLength parameters are wrong. The following shows a stalled control read transfer, but the same mechanism (tasks) applies to stalling a control write transfer.

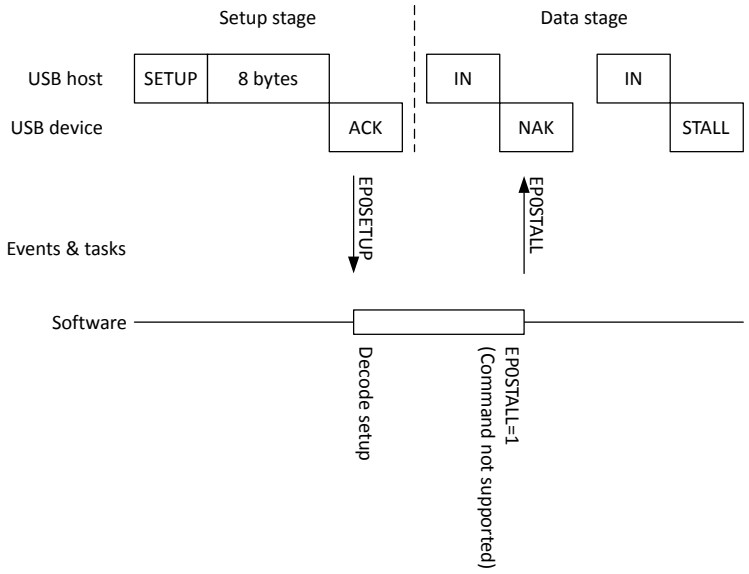


Figure 236: Control read gets stalled

See the *USB 2.0 Specification* and relevant class specifications for rules on stalling commands.

Note: The USBD peripheral handles the SetAddress transfer by itself. As a consequence, the software shall not process this command other than updating its state machine (see Device state diagram), nor initiate a status stage. If necessary, the address assigned by the host can be read out from the USBADDR register after the command has been processed.

7.39.9.1 Control read transfer

This section describes how the software behaves when responding to a control read transfer.

As mentioned earlier, the USB controller will not accept (NAK) any incoming IN tokens until software has finished decoding the command, determining the type of transfer, and preparing for the next stage (data or status) appropriately.

For a control read, transferring the data from memory into USBD will trigger a valid, acknowledged (ACK) IN transaction on USB.

The software has to prepare EasyDMA by pointing to the buffer containing the data to be transferred. If no other EasyDMA transfers are on-going with USBD, the software can send the STARTEPIN0 task, which will initiate the data transfer and transaction on USB.

A STARTED event (with EPIN0 bit set in the EPSTATUS register) will be generated as soon as the EPIN[0].PTR and .MAXCNT registers have been captured. Software may then prepare them for the next data transaction.

An ENDEPIN[0] event will be generated when the data has been transferred from memory to the USBD peripheral.

Finally, an EP0DATADONE event will be generated when the data has been transmitted over USB and acknowledged by the host.

The software can then either prepare and transmit the next data transaction by repeating the above sequence, or initiate the status stage through the EP0STATUS task.

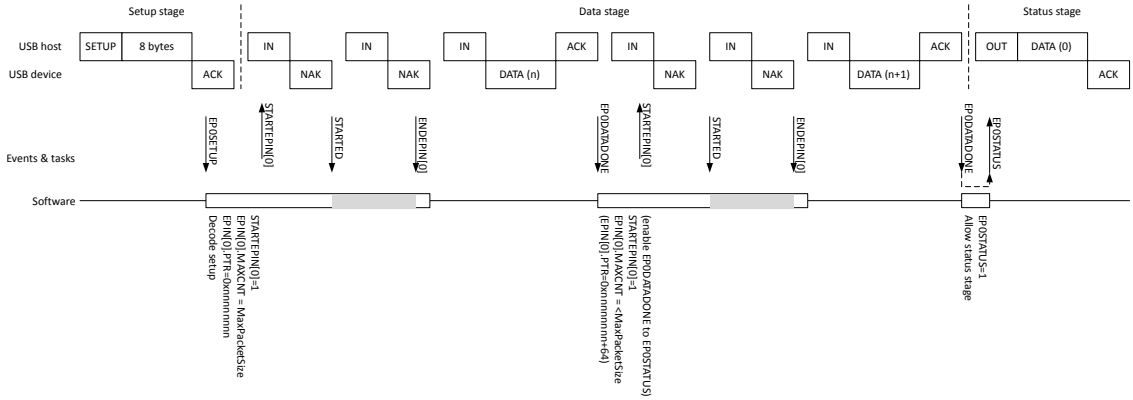


Figure 237: Control read transfer

It is possible to enable a shortcut from the EP0DATADONE event to the EP0STATUS task, typically if the data stage is expected to take a single transfer. If there is no data stage, the software can initiate the status stage through the EP0STATUS task right away, as as shown in the following figure.

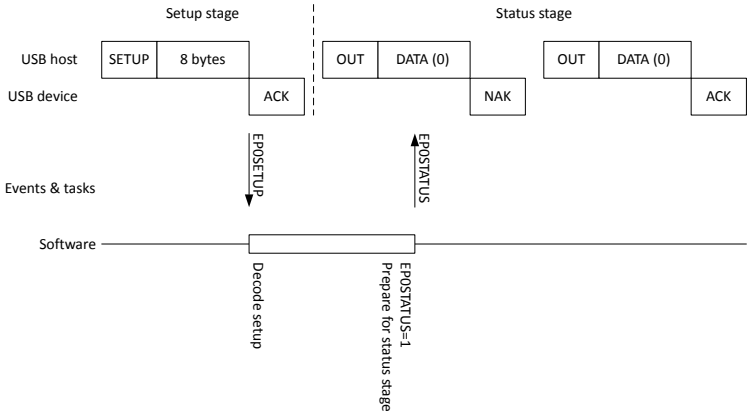


Figure 238: Control read no data transfer

7.39.9.2 Control write transfer

This section describes how the software responds to a control write transfer.

The software has to prepare EasyDMA by pointing to the buffer in memory that shall contain the incoming data. If no other EasyDMA transfers are ongoing with USBD, the software can then send the EP0RCVOUT task, which will make USBD acknowledge (ACK) the first OUT+DATA transaction from the host.

An EP0DATADONE event will be generated when a new OUT+DATA has been transmitted over USB, and is about to get acknowledged by the device.

After receiving the first transaction, a STARTED event (the EPOUT0 bit set in the EPSTATUS register) is generated when the EPOUT[0].PTR and .MAXCNT registers have been captured. Software may then prepare them for the next data transaction.

An ENDEPOUT[0] event will be generated when the data has been transferred from the USBD peripheral to memory. The software can then either prepare to receive the next data transaction by repeating the above sequence, or initiate the status stage through the EP0STATUS task. Until then, further incoming OUT +DATA transactions get a NAK response by the device.

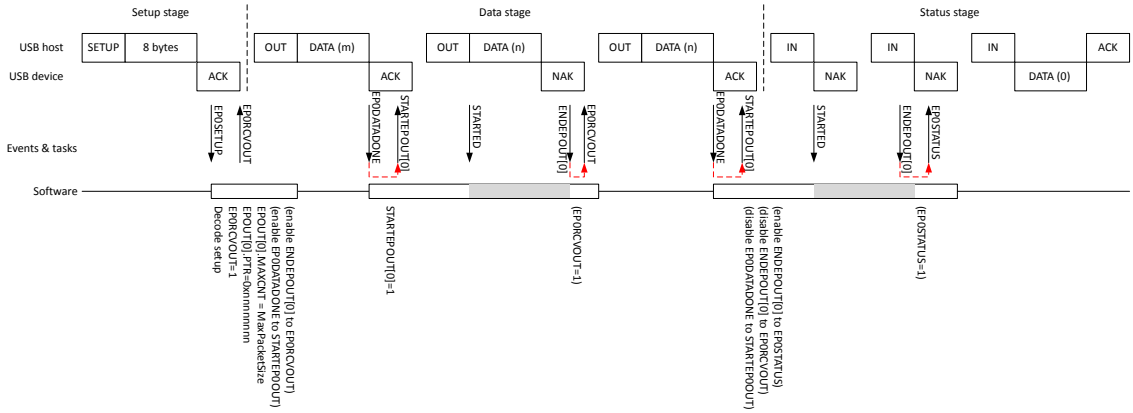


Figure 239: Control write transfer

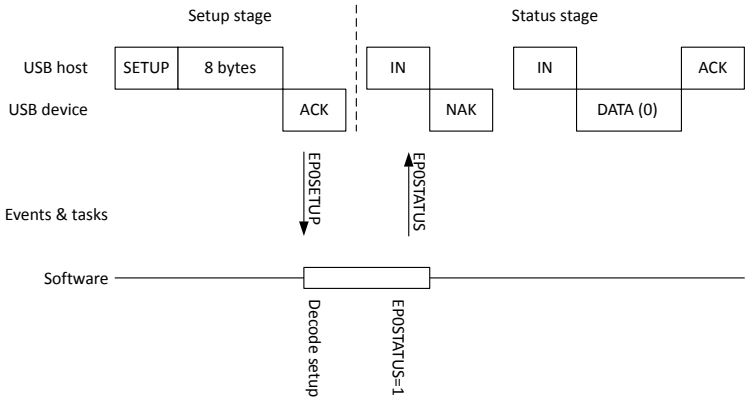


Figure 240: Control write no data transfer

7.39.10 Bulk and interrupt transactions

The USBD peripheral implements seven pairs of bulk/interrupt endpoints.

The bulk/interrupt endpoints have a fixed USB endpoint number, summarized in the following table.

Bulk endpoint #	USB IN endpoint	USB OUT endpoint
[1]	0x81	0x01
[2]	0x82	0x02
[3]	0x83	0x03
[4]	0x84	0x04
[5]	0x85	0x05
[6]	0x86	0x06
[7]	0x87	0x07

Table 177: Bulk/interrupt endpoint numbering

A bulk/interrupt transaction consists of a single data stage. Two consecutive, successful transactions are distinguished through alternating leading process ID (PID): DATA0 follows DATA1, DATA1 follows DATA0, etc. A repeated transaction is detected by re-using the same PID as previous transaction, i.e DATA0 follows DATA0, or DATA1 follows DATA1.

The USBD controller automatically toggles DATA0/DATA1 PIDs for every bulk/interrupt transaction.

If incoming data is corrupted (CRC does not match), the USBD controller automatically prevents DATA0/DATA1 from toggling, to request the host to resend the data.

In some specific cases, the software may want to force a data toggle (usually reset) on a specific IN endpoint, or force the expected toggle on an OUT endpoint, for instance as a consequence of the host issuing **`ClearFeature`**, **`SetInterface`**, or selecting an alternate setting. Controlling the data toggle of data IN or OUT endpoint n (n=1..7) is done through register DTOGGLE.

The bulk/interrupt transaction in USB full-speed can be of any size up to 64 bytes. It must be a multiple of four bytes and 32-bit aligned in memory.

When the USB transaction has completed, an EPDATA event is generated. Until new data has been transferred by EasyDMA from memory to the USBD peripheral (signalled by the ENDEPIN[n] event), the hardware will automatically respond with NAK to all incoming IN tokens. Software has to configure and start the EasyDMA transfer once it is ready to send more data.

Each IN or OUT data endpoint has to be explicitly enabled by software through register EPINEN or EPOUTEN, according to the configuration declared by the device and selected by the host through the **`SetConfig`** command.

A disabled data endpoint will not respond to any traffic from the host. An enabled data endpoint will normally respond NAK or ACK (depending on the readiness of the buffers), or STALL (if configured in register EPSTALL), in which case the endpoint is asked to halt. The halted (or not) state of a given endpoint can be read back from register HALTED.EPIN[n] or HALTED.EPOUT[n]. The format of the returned 16-bit value can be copied as is, as a response to a **`GetStatusEndpoint`** request from the host.

Enabling or disabling an endpoint will not change its halted state. However, a USB reset will disable and clear the halted state of all data endpoints.

The control endpoint 0 IN and OUT can also be enabled and/or halted using the same mechanisms, but due to USB specification, receiving a SETUP will override its state.

7.39.10.1 Bulk and interrupt IN transaction

The host issues IN tokens to receive bulk/interrupt data. In order to send data, the software has to enable the endpoint and prepare an EasyDMA transfer on the desired endpoint.

Bulk/interrupt IN endpoints are enabled or disabled through their respective INn bit (n=1..7) in EPINEN register.

It is also possible to stall or resume communication on an endpoint through the EPSTALL register.

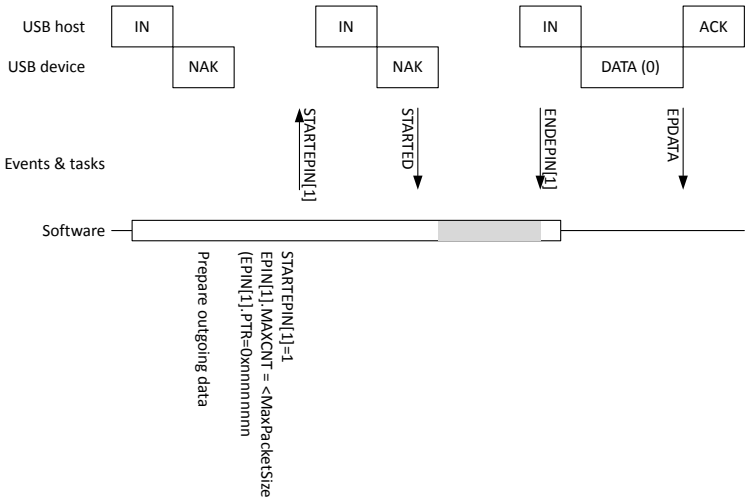


Figure 241: Bulk/interrupt IN transaction

It is possible (and in some situations it is required) to respond to an IN token with a zero-length data packet.

> **Note:** On many USB hosts, not responding (DATA+ACK or NAK) to three IN tokens on an interrupt endpoint would have the host disable that endpoint as a consequence. Re-enumerating the device (unplug-replug) may be required to restore functionality. Make sure that the relevant data endpoints are enabled for normal operation as soon as the device gets configured through a **`SetConfig`** request.

7.39.10.2 Bulk and interrupt OUT transaction

When the host wants to transmit bulk/interrupt data, it issues an OUT token (packet) followed by a DATA packet on a given endpoint n (n=1..7).

A NAK is returned until the software writes any value to register SIZE.EPOUT[n], indicating that the content of the local buffer can be overwritten. Upon receiving the next OUT+DATA transaction, an ACK is returned to the host while an EPDATA event is generated (and the EPDATASTATUS register flags are set to indicate on which endpoint this happened). Once the EasyDMA is prepared and enabled, by writing the EPOUT[n] registers and triggering the STARTEPOUT[n] task, the incoming data will be transferred to memory. Until that transfer is finished, the hardware will automatically NAK any other incoming OUT+DATA packets. Only when the EasyDMA transfer is done (signalled by the ENDEPOUT[n] event), or as soon as any values are written by the software in register SIZE.EPOUT[n], the endpoint n will accept incoming OUT+DATA again.

It is allowed for the host to send zero-length data packets.

Bulk/interrupt OUT endpoints are enabled or disabled through their respective OUTn bit (n=1..7) in the EPOUTEN register. It is also possible to stall or resume communication on an endpoint through the EPSTALL register.

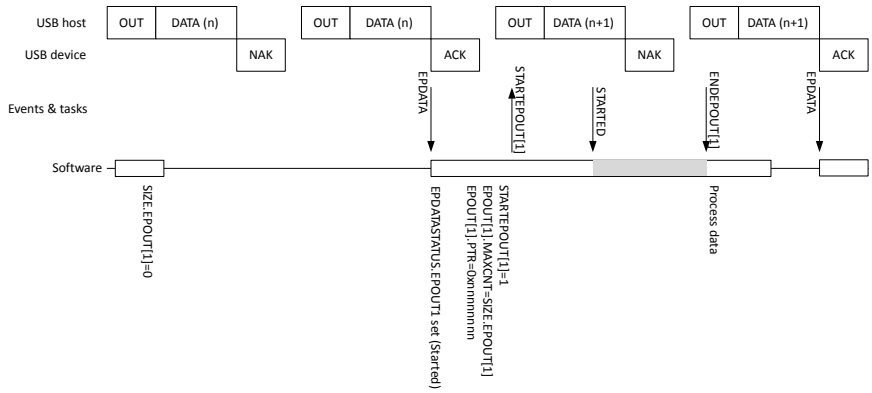


Figure 242: Bulk/interrupt OUT transaction

7.39.11 Isochronous transactions

The USBD peripheral implements isochronous (ISO) endpoints.

The ISO endpoints have a fixed USB endpoint number, summarized in the following table.

ISO endpoint #	USB IN endpoint	USB OUT endpoint
[0]	0x88	0x08

Table 178: Isochronous endpoint numbering

An isochronous transaction consists of a single, non-acknowledged data stage. The host sends out a start of frame at a regular interval (1 ms), and data follows IN or OUT tokens within each frame.

EasyDMA allows transferring ISO data directly from and to memory. EasyDMA transfers must be initiated by the software, which can synchronize with the SOF (start of frame) events.

Because the timing of the start of frame is very accurate, the SOF event can be used for jobs such as synchronizing a local timer through the SOF event and PPI. The SOF event gets synchronized to the 16 MHz clock prior to being made available to the PPI.

Every start of frame increments a free-running counter, which can be read by software through the FRAMECNTR register.

Each IN or OUT ISO data endpoint has to be explicitly enabled by software through register EPINEN or EPOUTEN, according to the configuration declared by the device and selected by the host through the **`SetConfig`** command. A disabled ISO IN data endpoint will not respond to any traffic from the host. A disabled ISO OUT data endpoint will ignore any incoming traffic from the host.

The USBD peripheral has an internal 1 kB buffer associated with ISO endpoints. The user can either allocate the full amount to the IN or the OUT endpoint, or split the buffer allocation between the two using register ISOSPLIT.

The internal buffer also sets the maximum size of the ISO OUT and ISO IN transfers: 1023 bytes when the full buffer is dedicated to either ISO OUT or ISO IN, and half when the buffer is split between the two.

7.39.11.1 Isochronous IN transaction

When the host wants to receive isochronous (ISO) data, it issues an IN token on the isochronous endpoint.

After the data has been transferred using the EasyDMA, the USB controller on the isochronous IN endpoint responds to the IN token with the transferred data using the ISOIN.MAXCNT for the size of the packet.

The ISO IN data endpoint has to be explicitly enabled by software through the ISOIN0 bit in register EPINEN.

When an ISO IN endpoint is enabled and no data transferred with EasyDMA, the response of the USBD depends on the setting of the RESPONSE field in register ISOINCONFIG. It can either provide no response to an IN token or respond with a zero-length data.

If the EasyDMA transfer on the isochronous endpoint is not completed before the next SOF event, the result of the transfer is undefined.

The maximum size of an ISO IN transfer in USB full-speed is 1023 bytes. The data buffer has to be a multiple of 4 bytes 32-bit aligned in memory. However, the amount of bytes transferred on the USB data endpoint can be of any size (up to 1023 bytes, if not shared with an OUT ISO endpoint).

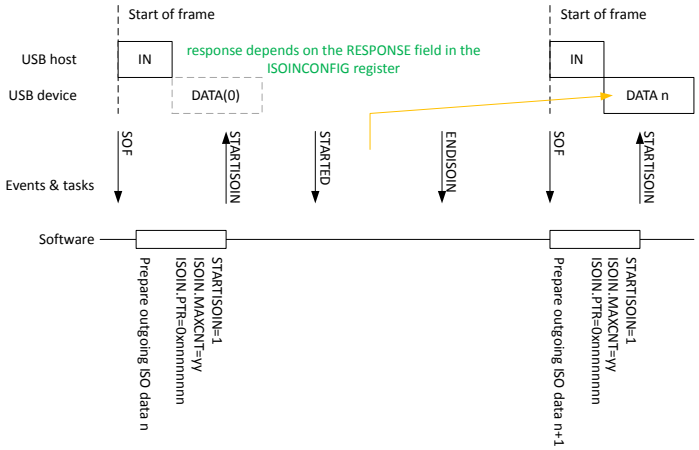


Figure 243: Isochronous IN transfer

7.39.11.2 Isochronous OUT transaction

When the host wants to send isochronous (ISO) data, it issues an OUT token on the isochronous endpoint, followed by data.

The ISO OUT data endpoint has to be explicitly enabled by software through the ISOOUT0 bit in register EPOUTEN.

The amount of last received ISO OUT data is provided in the SIZE.ISOOUT register. Software shall interpret the ZERO and SIZE fields as presented in the following table.

ZERO	SIZE	Last received data size
Normal	0	No data received at all
Normal	1..1023	1..1023 bytes of data received
ZeroData	(not of interest)	Zero-length data packet received

Table 179: ISO OUT incoming data size

When EasyDMA is prepared and started, triggering a STARTISOOUT task initiates an EasyDMA transfer to memory. Software shall synchronize ISO OUT transfers with the SOF events. EasyDMA uses the address in ISOOUT.PTR and size in ISOOUT.MAXCNT for every new transfer.

If the EasyDMA transfer on the isochronous endpoint is not completed before the next SOF event, the result of the transfer is undefined.

The maximum size of an isochronous OUT transfer in USB full-speed is 1023 bytes. The data buffer has to be a multiple of 4 bytes and 32-bit aligned in Data RAM. However, the amount of bytes transferred on the USB data endpoint can be of any size (up to 1023 bytes if not shared with an IN ISO endpoint).

If the last received ISO data packet is corrupted (wrong CRC), the USB controller generates an USBEVENT event (at the same time as SOF) and indicates a CRC error on ISOOUTCRC in register EVENTCAUSE. EasyDMA will transfer the data anyway if it has been set up properly.

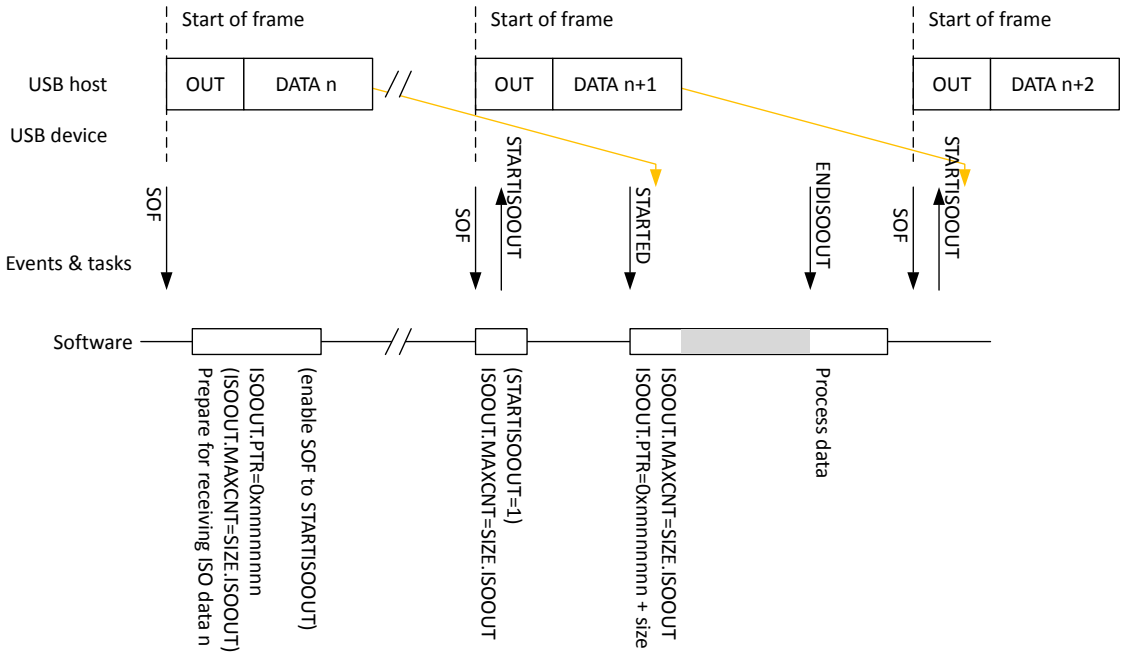


Figure 244: Isochronous OUT transfer

7.39.12 USB register access limitations

Some of the registers in USBD cannot be accessed in specific conditions.

This may be the case when USBD is not enabled (using the ENABLE register) and ready (signalled by the READY bit in EVENTCAUSE after a USBEVENT event), or when USBD is in low power mode while the USB bus is suspended.

Triggering any tasks, including the tasks triggered through the PPI, is affected by this behavior. In addition, the following registers are affected:

- HALTED.EPIN[0..7]
- HALTED.EPOUT[0..7]
- USBADDR
- BMREQUESTTYPE
- BREQUEST
- WVALUEL
- WVALUEH
- WINDEXL
- WINDEXH
- WLENGTHL
- WLENGTHH
- SIZE.EPOUT[0..7]
- SIZE.ISOOUT
- USBPULLUP
- DTOGGLE
- EPINEN
- EPOUTEN
- EPSTALL
- ISOSPLIT
- FRAMECNTR

7.39.13 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50036000 0x40036000	APPLICATION	USBD	USBD : S USBD : NS	US	SA	Universal serial bus device	

Table 180: Instances

Register	Offset	Security	Description
TASKS_STARTEPIN[n]	0x004		Captures the EPIN[n].PTR and EPIN[n].MAXCNT registers values, and enables endpoint IN n to respond to traffic from host
TASKS_STARTISOIN	0x024		Captures the ISOIN.PTR and ISOIN.MAXCNT registers values, and enables sending data on ISO endpoint
TASKS_STARTEPOUT[n]	0x028		Captures the EPOUT[n].PTR and EPOUT[n].MAXCNT registers values, and enables endpoint n to respond to traffic from host
TASKS_STARTISOOUT	0x048		Captures the ISOOUT.PTR and ISOOUT.MAXCNT registers values, and enables receiving of data on ISO endpoint
TASKS_EP0RCVOUT	0x04C		Allows OUT data stage on control endpoint 0
TASKS_EP0STATUS	0x050		Allows status stage on control endpoint 0
TASKS_EP0STALL	0x054		Stalls data and status stage on control endpoint 0
TASKS_DPDMDRIVE	0x058		Forces D+ and D- lines into the state defined in the DPDMVALUE register

Register	Offset	Security	Description
TASKS_DPDMNODRIVE	0x05C		Stops forcing D+ and D- lines into any state (USB engine takes control)
SUBSCRIBE_STARTEPIN[n]	0x084		Subscribe configuration for task STARTEPIN[n]
SUBSCRIBE_STARTISOIN	0x0A4		Subscribe configuration for task STARTISOIN
SUBSCRIBE_STARTEPOUT[n]	0x0A8		Subscribe configuration for task STARTEPOUT[n]
SUBSCRIBE_STARTISOOUT	0x0C8		Subscribe configuration for task STARTISOOUT
SUBSCRIBE_EP0RCVOUT	0x0CC		Subscribe configuration for task EP0RCVOUT
SUBSCRIBE_EP0STATUS	0x0D0		Subscribe configuration for task EP0STATUS
SUBSCRIBE_EP0STALL	0x0D4		Subscribe configuration for task EP0STALL
SUBSCRIBE_DPDMDRIVE	0x0D8		Subscribe configuration for task DPDMDRIVE
SUBSCRIBE_DPDMNODRIVE	0x0DC		Subscribe configuration for task DPDMNODRIVE
EVENTS_USBRESET	0x100		Signals that a USB reset condition has been detected on USB lines
EVENTS_STARTED	0x104		Confirms that the EPIN[n].PTR and EPIN[n].MAXCNT, or EPOUT[n].PTR and EPOUT[n].MAXCNT registers have been captured on all endpoints reported in the EPSTATUS register
EVENTS_ENDEPIN[n]	0x108		The whole EPIN[n] buffer has been consumed. The buffer can be accessed safely by software.
EVENTS_EP0DATADONE	0x128		An acknowledged data transfer has taken place on the control endpoint
EVENTS_ENDISOIN	0x12C		The whole ISOIN buffer has been consumed. The buffer can be accessed safely by software.
EVENTS_ENDEPOUT[n]	0x130		The whole EPOUT[n] buffer has been consumed. The buffer can be accessed safely by software.
EVENTS_ENDISOOUT	0x150		The whole ISOOUT buffer has been consumed. The buffer can be accessed safely by software.
EVENTS_SOF	0x154		Signals that a SOF (start of frame) condition has been detected on USB lines
EVENTS_USBEVENT	0x158		An event or an error not covered by specific events has occurred. Check EVENTCAUSE register to find the cause.
EVENTS_EP0SETUP	0x15C		A valid SETUP token has been received (and acknowledged) on the control endpoint
EVENTS_EPDATA	0x160		A data transfer has occurred on a data endpoint, indicated by the EPDATASTATUS register
PUBLISH_USBRESET	0x180		Publish configuration for event USBRESET
PUBLISH_STARTED	0x184		Publish configuration for event STARTED
PUBLISH_ENDEPIN[n]	0x188		Publish configuration for event ENDEPIN[n]
PUBLISH_EP0DATADONE	0x1A8		Publish configuration for event EP0DATADONE
PUBLISH_ENDISOIN	0x1AC		Publish configuration for event ENDISOIN
PUBLISH_ENDEPOUT[n]	0x1B0		Publish configuration for event ENDEPOUT[n]
PUBLISH_ENDISOOUT	0x1D0		Publish configuration for event ENDISOOUT
PUBLISH_SOF	0x1D4		Publish configuration for event SOF
PUBLISH_USBEVENT	0x1D8		Publish configuration for event USBEVENT
PUBLISH_EP0SETUP	0x1DC		Publish configuration for event EP0SETUP
PUBLISH_EPDATA	0x1E0		Publish configuration for event EPDATA
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
EVENTCAUSE	0x400		Details on what caused the USBEVENT event
HALTED.EPIN[n]	0x420		IN endpoint halted status. Can be used as is as response to a GetStatus() request to endpoint.
HALTED.EPOUT[n]	0x444		OUT endpoint halted status. Can be used as is as response to a GetStatus() request to endpoint.
EPSTATUS	0x468		Provides information on which endpoint's EasyDMA registers have been captured
EPDATASTATUS	0x46C		Provides information on which endpoint(s) an acknowledged data transfer has occurred (EPDATA event)
USBADDR	0x470		Device USB address

Register	Offset	Security	Description
BMREQUESTTYPE	0x480		SETUP data, byte 0, bmRequestType
BREQUEST	0x484		SETUP data, byte 1, bRequest
WVALUEL	0x488		SETUP data, byte 2, LSB of wValue
WVALUEH	0x48C		SETUP data, byte 3, MSB of wValue
WINDEXL	0x490		SETUP data, byte 4, LSB of wIndex
WINDEXH	0x494		SETUP data, byte 5, MSB of wIndex
WLENGTHL	0x498		SETUP data, byte 6, LSB of wLength
WLENGTHH	0x49C		SETUP data, byte 7, MSB of wLength
SIZE.EPOUT[n]	0x4A0		Number of bytes received last in the data stage of this OUT endpoint
SIZE.ISOOUT	0x4C0		Number of bytes received last on this ISO OUT data endpoint
ENABLE	0x500		Enable USB
USBPULLUP	0x504		Control of the USB pull-up
DPDMVALUE	0x508		State D+ and D- lines will be forced into by the DPDMDRIVE task. The DPDMNODRIVE task reverts the control of the lines to MAC IP (no forcing).
DTOGGLE	0x50C		Data toggle control and status
EPINEN	0x510		Endpoint IN enable
EPOUTEN	0x514		Endpoint OUT enable
EPSTALL	0x518		STALL endpoints
ISOSPLIT	0x51C		Controls the split of ISO buffers
FRAMECNTR	0x520		Returns the current value of the start of frame counter
LOWPOWER	0x52C		Controls USBD peripheral low power mode during USB suspend
ISOINCONFIG	0x530		Controls the response of the ISO IN endpoint to an IN token when no data is ready to be sent
EPIN[n].PTR	0x600		Data pointer
EPIN[n].MAXCNT	0x604		Maximum number of bytes to transfer
EPIN[n].AMOUNT	0x608		Number of bytes transferred in the last transaction
ISOIN.PTR	0x6A0		Data pointer
ISOIN.MAXCNT	0x6A4		Maximum number of bytes to transfer
ISOIN.AMOUNT	0x6A8		Number of bytes transferred in the last transaction
EPOUT[n].PTR	0x700		Data pointer
EPOUT[n].MAXCNT	0x704		Maximum number of bytes to transfer
EPOUT[n].AMOUNT	0x708		Number of bytes transferred in the last transaction
ISOOUT.PTR	0x7A0		Data pointer
ISOOUT.MAXCNT	0x7A4		Maximum number of bytes to transfer
ISOOUT.AMOUNT	0x7A8		Number of bytes transferred in the last transaction

Table 181: Register overview

7.39.13.1 TASKS_STARTEPIN[n] (n=0..7)

Address offset: 0x004 + (n × 0x4)

Captures the EPIN[n].PTR and EPIN[n].MAXCNT registers values, and enables endpoint IN n to respond to traffic from host

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STARTEPIN			Captures the EPIN[n].PTR and EPIN[n].MAXCNT registers values, and enables endpoint IN n to respond to traffic from host
			Trigger	1	Trigger task

7.39.13.2 TASKS_STARTISOIN

Address offset: 0x024

Captures the ISOIN.PTR and ISOIN.MAXCNT registers values, and enables sending data on ISO endpoint

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STARTISOIN			Captures the ISOIN.PTR and ISOIN.MAXCNT registers values, and enables sending data on ISO endpoint
			Trigger	1	Trigger task

7.39.13.3 TASKS_STARTEPOUT[n] (n=0..7)

Address offset: 0x028 + (n × 0x4)

Captures the EPOUT[n].PTR and EPOUT[n].MAXCNT registers values, and enables endpoint n to respond to traffic from host

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STARTEPOUT			Captures the EPOUT[n].PTR and EPOUT[n].MAXCNT registers values, and enables endpoint n to respond to traffic from host
			Trigger	1	Trigger task

7.39.13.4 TASKS_STARTISOOUT

Address offset: 0x048

Captures the ISOOUT.PTR and ISOOUT.MAXCNT registers values, and enables receiving of data on ISO endpoint

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STARTISOOUT			Captures the ISOOUT.PTR and ISOOUT.MAXCNT registers values, and enables receiving of data on ISO endpoint
			Trigger	1	Trigger task

7.39.13.5 TASKS_EP0RCVOUT

Address offset: 0x04C

Allows OUT data stage on control endpoint 0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_EP0RCVOUT			Allows OUT data stage on control endpoint 0
			Trigger	1	Trigger task

7.39.13.6 TASKS_EP0STATUS

Address offset: 0x050

Allows status stage on control endpoint 0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_EP0STATUS			Allows status stage on control endpoint 0
			Trigger	1	Trigger task

7.39.13.7 TASKS_EP0STALL

Address offset: 0x054

Stalls data and status stage on control endpoint 0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_EP0STALL			Stalls data and status stage on control endpoint 0
			Trigger	1	Trigger task

7.39.13.8 TASKS_DPDMDRIVE

Address offset: 0x058

Forces D+ and D- lines into the state defined in the DPDMVALUE register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_DPDMDRIVE			Forces D+ and D- lines into the state defined in the DPDMVALUE register
			Trigger	1	Trigger task

7.39.13.9 TASKS_DPDMNODRIVE

Address offset: 0x05C

Stops forcing D+ and D- lines into any state (USB engine takes control)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_DPDMNODRIVE			Stops forcing D+ and D- lines into any state (USB engine takes control)
			Trigger	1	Trigger task

7.39.13.10 SUBSCRIBE_STARTEPIN[n] (n=0..7)

Address offset: 0x084 + (n × 0x4)

Subscribe configuration for task STARTEPIN[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STARTEPIN[n] will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.11 SUBSCRIBE_STARTISOIN

Address offset: 0x0A4

Subscribe configuration for task STARTISOIN

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STARTISOIN will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.12 SUBSCRIBE_STARTEPOUT[n] (n=0..7)

Address offset: 0x0A8 + (n × 0x4)

Subscribe configuration for task STARTEPOUT[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STARTEPOUT[n] will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.13 SUBSCRIBE_STARTISOOUT

Address offset: 0x0C8

Subscribe configuration for task STARTISOOUT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STARTISOOUT will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.14 SUBSCRIBE_EP0RCVOUT

Address offset: 0x0CC

Subscribe configuration for task EP0RCVOUT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task EP0RCVOUT will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.15 SUBSCRIBE_EP0STATUS

Address offset: 0x0D0

Subscribe configuration for task EP0STATUS

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task EP0STATUS will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.16 SUBSCRIBE_EP0STALL

Address offset: 0x0D4

Subscribe configuration for task EP0STALL

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task EP0STALL will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.17 SUBSCRIBE_DPDMDRIVE

Address offset: 0x0D8

Subscribe configuration for task DPDMDRIVE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task DPDMDRIVE will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.18 SUBSCRIBE_DPDMNODRIVE

Address offset: 0x0DC

Subscribe configuration for task DPDMNODRIVE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task DPDMNODRIVE will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.39.13.19 EVENTS_USBRESET

Address offset: 0x100

Signals that a USB reset condition has been detected on USB lines

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_USBRESET			Signals that a USB reset condition has been detected on USB lines
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.20 EVENTS_STARTED

Address offset: 0x104

Confirms that the EPIN[n].PTR and EPIN[n].MAXCNT, or EPOUT[n].PTR and EPOUT[n].MAXCNT registers have been captured on all endpoints reported in the EPSTATUS register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_STARTED			Confirms that the EPIN[n].PTR and EPIN[n].MAXCNT, or EPOUT[n].PTR and EPOUT[n].MAXCNT registers have been captured on all endpoints reported in the EPSTATUS register
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.21 EVENTS_ENDEPIN[n] (n=0..7)

Address offset: 0x108 + (n × 0x4)

The whole EPIN[n] buffer has been consumed. The buffer can be accessed safely by software.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDEPIN			The whole EPIN[n] buffer has been consumed. The buffer can be accessed safely by software.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.22 EVENTS_EP0DATADONE

Address offset: 0x128

An acknowledged data transfer has taken place on the control endpoint

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_EP0DATADONE			An acknowledged data transfer has taken place on the control endpoint
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.23 EVENTS_ENDISOIN

Address offset: 0x12C

The whole ISOIN buffer has been consumed. The buffer can be accessed safely by software.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDISOIN			The whole ISOIN buffer has been consumed. The buffer can be accessed safely by software.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.24 EVENTS_ENDEPOUT[n] (n=0..7)

Address offset: 0x130 + (n × 0x4)

The whole EPOUT[n] buffer has been consumed. The buffer can be accessed safely by software.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDEPOUT			The whole EPOUT[n] buffer has been consumed. The buffer can be accessed safely by software.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.25 EVENTS_ENDISOOUT

Address offset: 0x150

The whole ISOOUT buffer has been consumed. The buffer can be accessed safely by software.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_ENDISOOUT			The whole ISOOUT buffer has been consumed. The buffer can be accessed safely by software.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.26 EVENTS_SOF

Address offset: 0x154

Signals that a SOF (start of frame) condition has been detected on USB lines

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_SOF			Signals that a SOF (start of frame) condition has been detected on USB lines
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.27 EVENTS_USBEVENT

Address offset: 0x158

An event or an error not covered by specific events has occurred. Check EVENTCAUSE register to find the cause.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_USBEVENT			An event or an error not covered by specific events has occurred. Check EVENTCAUSE register to find the cause.
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.28 EVENTS_EP0SETUP

Address offset: 0x15C

A valid SETUP token has been received (and acknowledged) on the control endpoint

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_EP0SETUP			A valid SETUP token has been received (and acknowledged) on the control endpoint
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.29 EVENTS_EPDATA

Address offset: 0x160

A data transfer has occurred on a data endpoint, indicated by the EPDATASTATUS register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_EPDATA			A data transfer has occurred on a data endpoint, indicated by the EPDATASTATUS register
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.39.13.30 PUBLISH_USBRESET

Address offset: 0x180

Publish configuration for event USBRESET

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event USBRESET will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.31 PUBLISH_STARTED

Address offset: 0x184

Publish configuration for event STARTED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event STARTED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.32 PUBLISH_ENDEPIN[n] (n=0..7)

Address offset: 0x188 + (n × 0x4)

Publish configuration for event ENDEPIN[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDEPIN[n] will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.33 PUBLISH_EP0DATADONE

Address offset: 0x1A8

Publish configuration for event EP0DATADONE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event EP0DATADONE will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.34 PUBLISH_ENDISOIN

Address offset: 0x1AC

Publish configuration for event ENDISOIN

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDISOIN will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.35 PUBLISH_ENDEPOUT[n] (n=0..7)

Address offset: 0x1B0 + (n × 0x4)

Publish configuration for event ENDEPOUT[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDEPOUT[n] will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.36 PUBLISH_ENDISOOUT

Address offset: 0x1D0

Publish configuration for event ENDISOOUT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event ENDISOOUT will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.37 PUBLISH_SOF

Address offset: 0x1D4

Publish configuration for event SOF

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event SOF will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.38 PUBLISH_USBEVENT

Address offset: 0x1D8

Publish configuration for event USBEVENT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event USBEVENT will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.39 PUBLISH_EP0SETUP

Address offset: 0x1DC

Publish configuration for event EP0SETUP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event EP0SETUP will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.40 PUBLISH_EPDATA

Address offset: 0x1E0

Publish configuration for event EPDATA

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event EPDATA will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.39.13.41 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EP0DATADONE_STARTEPIN0			Shortcut between event EP0DATADONE and task STARTEPIN[0]
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
B	RW	EP0DATADONE_STARTEP			Shortcut between event EP0DATADONE and task STARTEPOUT[0]
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
C	RW	EP0DATADONE_EP0STATUS			Shortcut between event EP0DATADONE and task EP0STATUS
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
D	RW	ENDEPOUT0_EP0STATUS			Shortcut between event ENDEPOUT[0] and task EP0STATUS
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut
E	RW	ENDEPOUT0_EP0RCVOUT			Shortcut between event ENDEPOUT[0] and task EP0RCVOUT
			Disabled	0	Disable shortcut
			Enabled	1	Enable shortcut

7.39.13.42 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	USBRESET			Enable or disable interrupt for event USBRESET
			Disabled	0	Disable
			Enabled	1	Enable
B	RW	STARTED			Enable or disable interrupt for event STARTED
			Disabled	0	Disable
			Enabled	1	Enable
C-J	RW	ENDEPIN[i] (i=0..7)			Enable or disable interrupt for event ENDEPIN[i]
			Disabled	0	Disable
			Enabled	1	Enable
K	RW	EP0DATADONE			Enable or disable interrupt for event EP0DATADONE
			Disabled	0	Disable
			Enabled	1	Enable
L	RW	ENDISOIN			Enable or disable interrupt for event ENDISOIN
			Disabled	0	Disable
			Enabled	1	Enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
M-T	RW	ENDEPOUT[i] (i=0..7)			Enable or disable interrupt for event ENDEPOUT[i]
			Disabled	0	Disable
			Enabled	1	Enable
U	RW	ENDISOOUT			Enable or disable interrupt for event ENDISOOUT
			Disabled	0	Disable
			Enabled	1	Enable
V	RW	SOF			Enable or disable interrupt for event SOF
			Disabled	0	Disable
			Enabled	1	Enable
W	RW	USBEVENT			Enable or disable interrupt for event USBEVENT
			Disabled	0	Disable
			Enabled	1	Enable
X	RW	EP0SETUP			Enable or disable interrupt for event EP0SETUP
			Disabled	0	Disable
			Enabled	1	Enable
Y	RW	EPDATA			Enable or disable interrupt for event EPDATA
			Disabled	0	Disable
			Enabled	1	Enable

7.39.13.43 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	USBRESET			Write '1' to enable interrupt for event USBRESET
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	STARTED			Write '1' to enable interrupt for event STARTED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C-J	RW	ENDEPIN[i] (i=0..7)			Write '1' to enable interrupt for event ENDEPIN[i]
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
K	RW	EP0DATADONE			Write '1' to enable interrupt for event EP0DATADONE
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
L	RW	ENDISOIN			Write '1' to enable interrupt for event ENDISOIN
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
M-T	RW	ENDEPOUT[i] (i=0..7)			Write '1' to enable interrupt for event ENDEPOUT[i]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
U	RW	ENDISOOUT			Write '1' to enable interrupt for event ENDISOOUT
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
V	RW	SOF			Write '1' to enable interrupt for event SOF
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
W	RW	USBEVENT			Write '1' to enable interrupt for event USBEVENT
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
X	RW	EP0SETUP			Write '1' to enable interrupt for event EP0SETUP
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Y	RW	EPDATA			Write '1' to enable interrupt for event EPDATA
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.39.13.44 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	USBRESET			Write '1' to disable interrupt for event USBRESET
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	STARTED			Write '1' to disable interrupt for event STARTED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
C-J	RW	ENDEPIN[i] (i=0..7)			Write '1' to disable interrupt for event ENDEPIN[i]
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
K	RW	EP0DATADONE			Write '1' to disable interrupt for event EP0DATADONE
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
L	RW	ENDISOIN			Write '1' to disable interrupt for event ENDISOIN
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
M-T	RW	ENDEPOUT[i] (i=0..7)			Write '1' to disable interrupt for event ENDEPOUT[i]
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
U	RW	ENDISOOUT			Write '1' to disable interrupt for event ENDISOOUT
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
V	RW	SOF			Write '1' to disable interrupt for event SOF
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
W	RW	USBEVENT			Write '1' to disable interrupt for event USBEVENT
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
X	RW	EP0SETUP			Write '1' to disable interrupt for event EP0SETUP
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
Y	RW	EPDATA			Write '1' to disable interrupt for event EPDATA
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.39.13.45 EVENTCAUSE

Address offset: 0x400

Details on what caused the USBEVENT event

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	ISOOUTCRC			CRC error was detected on isochronous OUT endpoint 8. Write '1' to clear.
			NotDetected	0	No error detected
			Detected	1	Error detected
B	RW	SUSPEND			Signals that USB lines have been idle long enough for the device to enter suspend. Write '1' to clear.
			NotDetected	0	Suspend not detected
			Detected	1	Suspend detected
C	RW	RESUME			Signals that a RESUME condition (K state or activity restart) has been detected on USB lines. Write '1' to clear.
			NotDetected	0	Resume not detected

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
			Detected	1	Resume detected
D	RW	USBWUALLOWED			USB MAC has been woken up and operational. Write '1' to clear.
			NotAllowed	0	Wake up not allowed
			Allowed	1	Wake up allowed
E	RW	READY			USB device is ready for normal operation. Write '1' to clear.
			NotDetected	0	USBEVENT was not issued due to USBD peripheral ready
			Ready	1	USBD peripheral is ready

7.39.13.46 HALTED.EPIN[n] (n=0..7)

Address offset: 0x420 + (n × 0x4)

IN endpoint halted status. Can be used as is as response to a GetStatus() request to endpoint.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	GETSTATUS			IN endpoint halted status. Can be used as is as response to a GetStatus() request to endpoint.
			NotHalted	0	Endpoint is not halted
			Halted	1	Endpoint is halted

7.39.13.47 HALTED.EPOUT[n] (n=0..7)

Address offset: 0x444 + (n × 0x4)

OUT endpoint halted status. Can be used as is as response to a GetStatus() request to endpoint.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	GETSTATUS			OUT endpoint halted status. Can be used as is as response to a GetStatus() request to endpoint.
			NotHalted	0	Endpoint is not halted
			Halted	1	Endpoint is halted

7.39.13.48 EPSTATUS

Address offset: 0x468

Provides information on which endpoint's EasyDMA registers have been captured

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				R Q P O N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-I	RW	EPIN[i] (i=0..8)			Captured state of endpoint's EasyDMA registers. Write '1' to clear.
			NoData	0	EasyDMA registers have not been captured for this endpoint
			DataDone	1	EasyDMA registers have been captured for this endpoint
J-R	RW	EPOUT[i] (i=0..8)			Captured state of endpoint's EasyDMA registers. Write '1' to clear.
			NoData	0	EasyDMA registers have not been captured for this endpoint
			DataDone	1	EasyDMA registers have been captured for this endpoint

7.39.13.49 EPDATASTATUS

Address offset: 0x46C

Provides information on which endpoint(s) an acknowledged data transfer has occurred (EPDATA event)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				N M L K J I H G F E D C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-G	RW	EPIN[i] (i=1..7)			Acknowledged data transfer on this IN endpoint. Write '1' to clear.
			NotDone	0	No acknowledged data transfer on this endpoint
			DataDone	1	Acknowledged data transfer on this endpoint has occurred
H-N	RW	EPOUT[i] (i=1..7)			Acknowledged data transfer on this OUT endpoint. Write '1' to clear.
			NotStarted	0	No acknowledged data transfer on this endpoint
			Started	1	Acknowledged data transfer on this endpoint has occurred

7.39.13.50 USBADDR

Address offset: 0x470

Device USB address

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	ADDR			Device USB address

7.39.13.51 BMREQUESTTYPE

Address offset: 0x480

SETUP data, byte 0, bmRequestType

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B B A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	RECIPIENT			Data transfer type
			Device	0	Device
			Interface	1	Interface
			Endpoint	2	Endpoint
			Other	3	Other
B	R	TYPE			Data transfer type
			Standard	0	Standard
			Class	1	Class
			Vendor	2	Vendor
C	R	DIRECTION			Data transfer direction
			HostToDevice	0	Host-to-device
			DeviceToHost	1	Device-to-host

7.39.13.52 BREQUEST

Address offset: 0x484

SETUP data, byte 1, bRequest

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	BREQUEST			SETUP data, byte 1, bRequest. Values provided for standard requests only, user must implement class and vendor values.
			STD_GET_STATUS	0	Standard request GET_STATUS
			STD_CLEAR_FEATURE	1	Standard request CLEAR_FEATURE
			STD_SET_FEATURE	3	Standard request SET_FEATURE
			STD_SET_ADDRESS	5	Standard request SET_ADDRESS
			STD_GET_DESCRIPTOR	6	Standard request GET_DESCRIPTOR
			STD_SET_DESCRIPTOR	7	Standard request SET_DESCRIPTOR
			STD_GET_CONFIGURATION	8	Standard request GET_CONFIGURATION
			STD_SET_CONFIGURATION	9	Standard request SET_CONFIGURATION
			STD_GET_INTERFACE	10	Standard request GET_INTERFACE
			STD_SET_INTERFACE	11	Standard request SET_INTERFACE
			STD_SYNCH_FRAME	12	Standard request SYNCH_FRAME

7.39.13.53 WVALUEL

Address offset: 0x488

SETUP data, byte 2, LSB of wValue

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	WVALUEL			SETUP data, byte 2, LSB of wValue

7.39.13.54 WVALUEH

Address offset: 0x48C

SETUP data, byte 3, MSB of wValue

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																									A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	R	WVALUEH			SETUP data, byte 3, MSB of wValue

7.39.13.55 WINDEXL

Address offset: 0x490

SETUP data, byte 4, LSB of wIndex

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																									A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	R	WINDEXL			SETUP data, byte 4, LSB of wIndex

7.39.13.56 WINDEXH

Address offset: 0x494

SETUP data, byte 5, MSB of wIndex

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																									A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	R	WINDEXH			SETUP data, byte 5, MSB of wIndex

7.39.13.57 WLENGTHL

Address offset: 0x498

SETUP data, byte 6, LSB of wLength

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																									A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	R	WLENGTHL			SETUP data, byte 6, LSB of wLength

7.39.13.58 WLENGTHH

Address offset: 0x49C

SETUP data, byte 7, MSB of wLength

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	WLENGTHH			SETUP data, byte 7, MSB of wLength

7.39.13.59 SIZE.EPOUT[n] (n=0..7)

Address offset: 0x4A0 + (n × 0x4)

Number of bytes received last in the data stage of this OUT endpoint

Write to any value to accept further OUT traffic on this endpoint, and overwrite the intermediate buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	SIZE			Number of bytes received last in the data stage of this OUT endpoint

7.39.13.60 SIZE.ISOOUT

Address offset: 0x4C0

Number of bytes received last on this ISO OUT data endpoint

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A A
Reset 0x00010000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	SIZE			Number of bytes received last on this ISO OUT data endpoint
B	R	ZERO			Zero-length data packet received
			Normal	0	No zero-length data received, use value in SIZE
			ZeroData	1	Zero-length data received, ignore value in SIZE

7.39.13.61 ENABLE

Address offset: 0x500

Enable USB

After writing Disabled to this register, reading the register will return Enabled until USBD is completely disabled.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	ENABLE			Enable USB
			Disabled	0	USB peripheral is disabled
			Enabled	1	USB peripheral is enabled

7.39.13.62 USBPULLUP

Address offset: 0x504

Control of the USB pull-up

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	CONNECT			Control of the USB pull-up on the D+ line
			Disabled	0	Pull-up is disconnected
			Enabled	1	Pull-up is connected to D+

7.39.13.63 DPDMVALUE

Address offset: 0x508

State D+ and D- lines will be forced into by the DPDMDRIVE task. The DPDMNODRIVE task reverts the control of the lines to MAC IP (no forcing).

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A A A A A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	STATE			State D+ and D- lines will be forced into by the DPDMDRIVE task
			Resume	1	D+ forced low, D- forced high (K state) for a timing preset in hardware (50 µs or 5 ms, depending on bus state)
			J	2	D+ forced high, D- forced low (J state)
			K	4	D+ forced low, D- forced high (K state)

7.39.13.64 DTOGGLE

Address offset: 0x50C

Data toggle control and status

First write this register with VALUE=Nop to select the endpoint, then either read it to get the status from VALUE, or write it again with VALUE=Data0 or Data1

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C C B A A A
Reset 0x00000100				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	EP			Select bulk endpoint number
B	RW	IO			Selects IN or OUT endpoint
			Out	0	Selects OUT endpoint
			In	1	Selects IN endpoint
C	RW	VALUE			Data toggle value
			Nop	0	No action on data toggle when writing the register with this value
			Data0	1	Data toggle is DATA0 on endpoint set by EP and IO
			Data1	2	Data toggle is DATA1 on endpoint set by EP and IO

7.39.13.65 EPINEN

Address offset: 0x510

Endpoint IN enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				I H G F E D C B A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A-H	RW	IN[i] (i=0..7)			Enable IN endpoint i
			Disable	0	Disable endpoint IN i (no response to IN tokens)
			Enable	1	Enable endpoint IN i (response to IN tokens)
I	RW	ISOIN			Enable ISO IN endpoint
			Disable	0	Disable ISO IN endpoint 8
			Enable	1	Enable ISO IN endpoint 8

7.39.13.66 EPOUTEN

Address offset: 0x514

Endpoint OUT enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				I H G F E D C B A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A-H	RW	OUT[i] (i=0..7)			Enable OUT endpoint i
			Disable	0	Disable endpoint OUT i (no response to OUT tokens)
			Enable	1	Enable endpoint OUT i (response to OUT tokens)
I	RW	ISOOUT			Enable ISO OUT endpoint 8
			Disable	0	Disable ISO OUT endpoint 8
			Enable	1	Enable ISO OUT endpoint 8

7.39.13.67 EPSTALL

Address offset: 0x518

STALL endpoints

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	EP			Select endpoint number
B	W	IO			Selects IN or OUT endpoint
			Out	0	Selects OUT endpoint
			In	1	Selects IN endpoint
C	W	STALL			Stall selected endpoint
			UnStall	0	Don't stall selected endpoint
			Stall	1	Stall selected endpoint

7.39.13.68 ISOSPLIT

Address offset: 0x51C

Controls the split of ISO buffers

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	SPLIT			Controls the split of ISO buffers
			OneDir	0x0000	Full buffer dedicated to either ISO IN or OUT
			HalfIN	0x0080	Lower half for IN, upper half for OUT

7.39.13.69 FRAMECNTR

Address offset: 0x520

Returns the current value of the start of frame counter

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	FRAMECNTR			Returns the current value of the start of frame counter

7.39.13.70 LOWPOWER

Address offset: 0x52C

Controls USBD peripheral low power mode during USB suspend

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	LOWPOWER			Controls USBD peripheral low-power mode during USB suspend
			ForceNormal	0	Software must write this value to exit low power mode and before performing a remote wake-up
			LowPower	1	Software must write this value to enter low power mode after DMA and software have finished interacting with the USB peripheral

7.39.13.71 ISOINCONFIG

Address offset: 0x530

Controls the response of the ISO IN endpoint to an IN token when no data is ready to be sent

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	RESPONSE			Controls the response of the ISO IN endpoint to an IN token when no data is ready to be sent
			NoResp	0	Endpoint does not respond in that case
			ZeroData	1	Endpoint responds with a zero-length data packet in that case

7.39.13.72 EPIN[n].PTR (n=0..7)

Address offset: 0x600 + (n × 0x14)

Data pointer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the memory chapter for details about which memories are available for EasyDMA.

7.39.13.73 EPIN[n].MAXCNT (n=0..7)

Address offset: 0x604 + (n × 0x14)

Maximum number of bytes to transfer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[64..0]	Maximum number of bytes to transfer

7.39.13.74 EPIN[n].AMOUNT (n=0..7)

Address offset: 0x608 + (n × 0x14)

Number of bytes transferred in the last transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT			Number of bytes transferred in the last transaction

7.39.13.75 ISOIN.PTR

Address offset: 0x6A0

Data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the memory chapter for details about which memories are available for EasyDMA.

7.39.13.76 ISOIN.MAXCNT

Address offset: 0x6A4

Maximum number of bytes to transfer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[1023..1]	Maximum number of bytes to transfer

7.39.13.77 ISOIN.AMOUNT

Address offset: 0x6A8

Number of bytes transferred in the last transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT			Number of bytes transferred in the last transaction

7.39.13.78 EPOUT[n].PTR (n=0..7)

Address offset: 0x700 + (n × 0x14)

Data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the memory chapter for details about which memories are available for EasyDMA.

7.39.13.79 EPOUT[n].MAXCNT (n=0..7)

Address offset: 0x704 + (n × 0x14)

Maximum number of bytes to transfer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT		[64..0]	Maximum number of bytes to transfer

7.39.13.80 EPOUT[n].AMOUNT (n=0..7)

Address offset: 0x708 + (n × 0x14)

Number of bytes transferred in the last transaction

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT			Number of bytes transferred in the last transaction

7.39.13.81 ISOOUT.PTR

Address offset: 0x7A0

Data pointer

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	PTR			Data pointer See the memory chapter for details about which memories are available for EasyDMA.

7.39.13.82 ISOOUT.MAXCNT

Address offset: 0x7A4

Maximum number of bytes to transfer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	MAXCNT			Maximum number of bytes to transfer

7.39.13.83 ISOOUT.AMOUNT

Address offset: 0x7A8

Number of bytes transferred in the last transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	AMOUNT			Number of bytes transferred in the last transaction

7.39.14 Electrical specification

7.39.14.1 USB Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
$R_{USB,PU,ACTIVE}$	Value of pull-up on D+, bus active (upstream device transmitting)	1425	2300	3090	Ω
$R_{USB,PU,IDLE}$	Value of pull-up on D+, bus idle	900	1200	1575	Ω
$t_{USB,DETRST}$	Minimum duration of an SE0 state to be detected as a USB reset condition			10	µs
$f_{USB,CLK}$	Frequency of local clock, USB active		48		MHz
$f_{USB,TOL}$	Accuracy of local clock, USB active[31]			±1000	ppm
$T_{USB,JITTER}$	Jitter on USB local clock, USB active			±1	ns

7.40 VMC — Volatile memory controller

VMC provides power control for RAM blocks.

Each RAM block, which may contain multiple RAM sections, can power up or power down independently in System ON and System OFF mode using RAM[n] registers. See the Memory chapter for more information about RAM blocks and sections.

7.40.1 RAM power states

The RAM power control registers are used for configuring the following:

- The RAM sections to be retained during System OFF
- The RAM sections to be retained and accessible during System ON

In System OFF, retention of a RAM section is configured in the RETENTION field of the corresponding register RAM[n].POWER (n=0..7) on page 736.

In System ON, retention and accessibility for a RAM section is configured in the RETENTION and POWER fields of the corresponding register RAM[n].POWER (n=0..7) on page 736.

[31] The local clock can be stopped during USB suspend

The following table summarizes the behavior of these registers.

Configuration			RAM section status	
System on/off	RAM[n].POWER.POWER	RAM[n].POWER.RETENTION	Accessible	Retained
Off	x	Off	No	No
Off	x	On	No	Yes
On	Off	Off	No	No
On	Off	On	No	Yes
On	On	x	Yes	Yes

Table 182: RAM section configuration

The advantage of not retaining RAM contents is that the overall current consumption is reduced.

See chapter Memory on page 18 for more information on RAM sections.

7.40.2 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50081000 0x40081000	APPLICATION	VMC	VMC : S VMC : NS	US	NA	Volatile memory controller	
0x41081000	NETWORK	VMC	VMC	NS	NA	Volatile memory controller	4 RAM slaves implemented 4 RAM slaves implemented

Table 183: Instances

Register	Offset	Security	Description
RAM[n].POWER	0x600		RAM[n] power control register
RAM[n].POWERSET	0x604		RAM[n] power control set register
RAM[n].POWERCLR	0x608		RAM[n] power control clear register

Table 184: Register overview

7.40.2.1 RAM[n].POWER (n=0..7)

Address offset: 0x600 + (n × 0x10)

RAM[n] power control register

[32] RAM section power off gives negligible reduction in current consumption when retention is on.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x0000FFFF				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A-P	RW	S[i]POWER (i=0..15)			Keep RAM section Si of RAM[n] on or off in System ON mode
					All RAM sections will be switched off in System OFF mode
			Off	0	Off
			On	1	On
Q-f	RW	S[i]RETENTION (i=0..15)			Keep retention on RAM section Si of RAM[n] when RAM section is switched off
			Off	0	Off
			On	1	On

7.40.2.2 RAM[n].POWERSET (n=0..7)

Address offset: 0x604 + (n × 0x10)

RAM[n] power control set register

When read, this register will return the value of the RAM[n].POWER register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x0000FFFF				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A-P	RW	S[i]POWER (i=0..15)			Keep RAM section Si of RAM[n] on or off in System ON mode
			On	1	On
Q-f	RW	S[i]RETENTION (i=0..15)			Keep retention on RAM section Si of RAM[n] when RAM section is switched off
			On	1	On

7.40.2.3 RAM[n].POWERCLR (n=0..7)

Address offset: 0x608 + (n × 0x10)

RAM[n] power control clear register

When read, this register will return the value of the RAM[n].POWER register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A
Reset 0x0000FFFF				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A-P	RW	S[i]POWER (i=0..15)			Keep RAM section Si of RAM[n] on or off in System ON mode
			Off	1	Off
Q-f	RW	S[i]RETENTION (i=0..15)			Keep retention on RAM section Si of RAM[n] when RAM section is switched off
			Off	1	Off

7.41 WDT — Watchdog timer

The countdown watchdog timer (WDT) uses the low-frequency clock source (LFCLK) and offers configurable and robust protection against application lock-up.

WDT must be configured before it is started. After configuration, WDT is started by triggering the START task.

When WDT is running, its configuration registers (CRV, RREN, and CONFIG) are blocked for further configuration.

WDT can be paused while the CPU is sleeping, or when the debugger has halted the CPU. WDT is implemented as a down-counter that generates a TIMEOUT event when it wraps over after counting down to 0. When WDT is started by the START task, the watchdog counter is loaded with the value specified in the CRV register. This counter is also reloaded with the value specified in the CRV register when a reload request is granted.

The timeout period for the watchdog is given by the following equation:

```
timeout [s] = ( CRV + 1 ) / 32768
```

When started, WDT will make the 32.768 kHz RC oscillator start if no other 32.768 kHz clock source is running and generating the 32.768 kHz system clock, see chapter CLOCK — Clock control on page 69.

7.41.1 Reload criteria

WDT has eight separate reload request registers. These registers are used to request WDT to reload its counter with the value specified in the CRV register. To reload the watchdog counter, write `0x6E524635` to all enabled reload registers.

One or more RR registers can be individually enabled through the RREN register.

7.41.2 Temporarily pausing the watchdog

By default, the watchdog will be active counting down the down-counter while the CPU is sleeping. It is possible to configure the watchdog to automatically pause when the CPU is sleeping or when it is stopped by the debugger.

Entering System OFF mode will stop and disable the watchdog.

7.41.3 Watchdog reset

A TIMEOUT event automatically leads to a watchdog reset.

If the watchdog is configured to generate an interrupt on the TIMEOUT event, the watchdog reset is postponed by two 32.768 kHz clock cycles after the TIMEOUT event is generated. Once the TIMEOUT event is generated, and unless the watchdog is stopped, the impending watchdog reset will occur.

The watchdog can be reset from several reset sources, see Application core reset behavior on page 64. After a reset, the watchdog configuration registers are available for configuration.

See RESET — Reset control on page 61 for more information about reset sources.

7.41.4 Stopping the watchdog

By default, the watchdog cannot be stopped. It is possible to configure the watchdog to allow the STOP task.

To stop the watchdog, perform the following steps.

1. Set the CONFIG register's STOPEN field to `Enable` during watchdog configuration.
2. Write the special value 0x6E524635 to the TSEN register.
3. Invoke the STOP task.

When these conditions are met, the watchdog is stopped and a STOPPED event is issued.

When the watchdog is stopped, its configuration registers CRV, RREN, and CONFIG are no longer blocked.

> **Note:** It is recommended to write zeros to TSEN on page 744 after the watchdog has stopped, to avoid runaway code triggering the STOP task.

7.41.5 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50018000 0x40018000	APPLICATION	WDT	WDT0 : S WDT0 : NS	US	NA	Watchdog timer 0	
0x50019000 0x40019000	APPLICATION	WDT	WDT1 : S WDT1 : NS	US	NA	Watchdog timer 1	
0x4100B000	NETWORK	WDT	WDT	NS	NA	Watchdog timer	

Table 185: Instances

Register	Offset	Security	Description
TASKS_START	0x000		Start WDT
TASKS_STOP	0x004		Stop WDT
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
EVENTS_TIMEOUT	0x100		Watchdog timeout
EVENTS_STOPPED	0x104		Watchdog stopped
PUBLISH_TIMEOUT	0x180		Publish configuration for event TIMEOUT
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
NMIENSET	0x324		Enable interrupt
NMIENCLR	0x328		Disable interrupt
RUNSTATUS	0x400		Run status
REQSTATUS	0x404		Request status
CRV	0x504		Counter reload value
RREN	0x508		Enable register for reload request registers
CONFIG	0x50C		Configuration register
TSEN	0x520		Task stop enable
RR[n]	0x600		Reload request n

Table 186: Register overview

7.41.5.1 TASKS_START

Address offset: 0x000

Start WDT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_START			Start WDT
			Trigger	1	Trigger task

7.41.5.2 TASKS_STOP

Address offset: 0x004

Stop WDT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	TASKS_STOP			Stop WDT
			Trigger	1	Trigger task

7.41.5.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task START

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task START will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.41.5.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that task STOP will subscribe to
B	RW	EN			
			Disabled	0	Disable subscription
			Enabled	1	Enable subscription

7.41.5.5 EVENTS_TIMEOUT

Address offset: 0x100

Watchdog timeout

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_TIMEOUT			Watchdog timeout
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.41.5.6 EVENTS_STOPPED

Address offset: 0x104

Watchdog stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	EVENTS_STOPPED			Watchdog stopped
			NotGenerated	0	Event not generated
			Generated	1	Event generated

7.41.5.7 PUBLISH_TIMEOUT

Address offset: 0x180

Publish configuration for event TIMEOUT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event TIMEOUT will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.41.5.8 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event STOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	CHIDX		[255..0]	DPPI channel that event STOPPED will publish to.
B	RW	EN			
			Disabled	0	Disable publishing
			Enabled	1	Enable publishing

7.41.5.9 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TIMEOUT			Write '1' to enable interrupt for event TIMEOUT
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	STOPPED			Write '1' to enable interrupt for event STOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.41.5.10 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TIMEOUT			Write '1' to disable interrupt for event TIMEOUT
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	STOPPED			Write '1' to disable interrupt for event STOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.41.5.11 NMIENSET

Address offset: 0x324

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TIMEOUT			Write '1' to enable interrupt for event TIMEOUT
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	STOPPED			Write '1' to enable interrupt for event STOPPED
			Set	1	Enable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.41.5.12 NMIENCLR

Address offset: 0x328

Disable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TIMEOUT			Write '1' to disable interrupt for event TIMEOUT
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled
B	RW	STOPPED			Write '1' to disable interrupt for event STOPPED
			Clear	1	Disable
			Disabled	0	Read: Disabled
			Enabled	1	Read: Enabled

7.41.5.13 RUNSTATUS

Address offset: 0x400

Run status

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	RUNSTATUSWDT			Indicates whether or not WDT is running
			NotRunning	0	Watchdog is not running
			Running	1	Watchdog is running

7.41.5.14 REQSTATUS

Address offset: 0x404

Request status

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					H G F E D C B A
Reset 0x00000001					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A-H	R	RR[i] (i=0..7)			Request status for RR[i] register
			DisabledOrRequested	0	RR[i] register is not enabled, or are already requesting reload
			EnabledAndUnrequested	1	RR[i] register is enabled, and are not yet requesting reload

7.41.5.15 CRV

Address offset: 0x504

Counter reload value

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	CRV		[0xF..0xFFFFFFFF]	Counter reload value in number of cycles of the 32.768 kHz clock

7.41.5.16 RREN

Address offset: 0x508

Enable register for reload request registers

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				H G F E D C B A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A-H	RW	RR[i] (i=0..7)			Enable or disable RR[i] register
			Disabled	0	Disable RR[i] register
			Enabled	1	Enable RR[i] register

7.41.5.17 CONFIG

Address offset: 0x50C

Configuration register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				F C A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	SLEEP			Configure WDT to either be paused, or kept running, while the CPU is sleeping
			Pause	0	Pause WDT while the CPU is sleeping
			Run	1	Keep WDT running while the CPU is sleeping
C	RW	HALT			Configure WDT to either be paused, or kept running, while the CPU is halted by the debugger
			Pause	0	Pause WDT while the CPU is halted by the debugger
			Run	1	Keep WDT running while the CPU is halted by the debugger
F	RW	STOPEN			Allow stopping WDT
			Disable	0	Do not allow stopping WDT
			Enable	1	Allow stopping WDT

7.41.5.18 TSEN

Address offset: 0x520

Task stop enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	TSEN			Allow stopping WDT
			Enable	0x6E524635	Value to allow stopping WDT

7.41.5.19 RR[n] (n=0..7)

Address offset: 0x600 + (n × 0x4)

Reload request n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	RR			Reload request register
			Reload	0x6E524635	Value to request a reload of the watchdog timer

7.41.6 Electrical specification

7.41.6.1 Watchdog Timer Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
t_{WDT}	Time out interval	..	..	..	

8 Debug and trace

The debug and trace system offers a flexible and powerful mechanism for non-intrusive debugging.

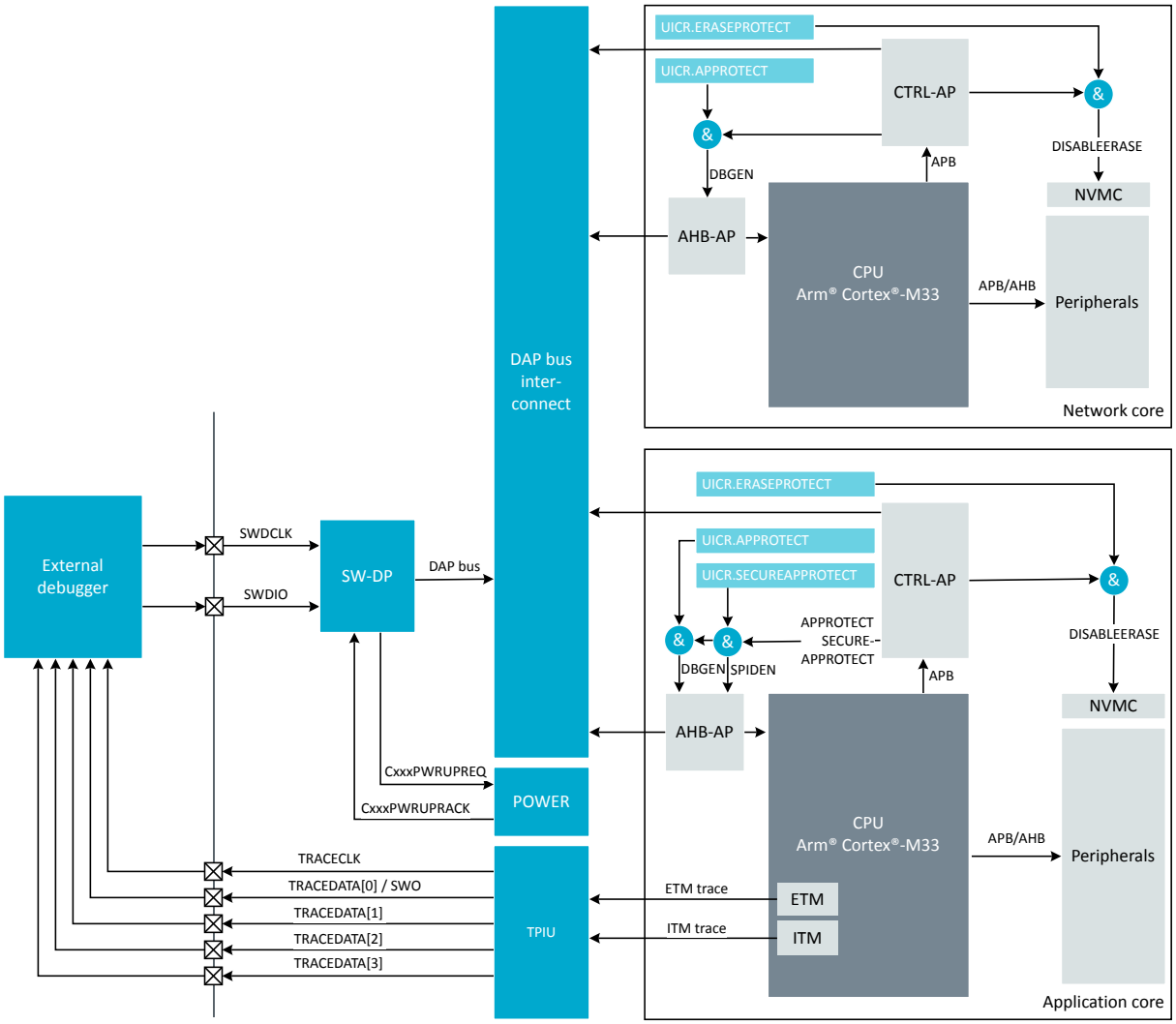


Figure 245: Debug and trace overview

The main features of the debug and trace system are:

- Access port connection to application core Arm Cortex-M33
  - Eight breakpoints
  - Four watchpoint comparators
  - Instrumentation trace macrocell (ITM)
  - Embedded trace macrocell (ETM)
  - Access protection through APPROTECT, ERASEPROTECT and SECUREAPPROTECT
- Access port connection to network core Arm Cortex-M33
  - Eight breakpoints
  - Four watchpoints
  - Access protection through APPROTECT and ERASEPROTECT
- Serial wire debug (SWD) interface protocol version 2 with multidrop support
- Trace port interface unit (TPIU)

- 4-bit parallel trace of ITM and ETM trace data
- Serial wire output (SWO) trace of ITM data

8.1 DAP — Debug access port

An external debugger can access the device using the DAP.

The DAP is a standard Arm CoreSight™ serial wire debug port (SW-DP) that implements the serial wire debug (SWD) protocol – a two-pin serial interface using SWDCLK and SWDIO pins (see Debug and trace overview on page 746).

Note:

- The SWDIO line has an internal pull-up resistor.
- The SWDCLK line has an internal pull-down resistor.

There are several access ports that connect to different parts of the system. See the following table for more information.

AP ID	Type	Description
0	AHB-AP	Application subsystem access port
1	AHB-AP	Network subsystem access port
2	CTRL-AP	Application subsystem control access port
3	CTRL-AP	Network subsystem control access port

Table 187: Access port overview

The AHB-AP and APB-AP access ports are standard Arm components documented in the *Arm CoreSight SoC-400 Technical Reference Manual, Revision r3p2*. The CTRL-AP access port is proprietary (see CTRL-AP - Control access port on page 757).

8.1.1 Registers

Register	Offset	Security	Description
TARGETID	0x042		The TARGETID register provides information about the target when the host is connected to a single device. The TARGETID register is accessed by a read of DP register 0x4 when the DPBANKSEL bit in the SELECT register is set to 0x2.
DLPIDR	0x043		The DLPIDR register provides information about the serial wire debug protocol version. Accessed by a read of DP register 0x4 when the DPBANKSEL bit in the SELECT register is set to 0x3.

Table 188: Register overview

8.1.1.1 TARGETID

Address offset: 0x042

The TARGETID register provides information about the target when the host is connected to a single device.

The TARGETID register is accessed by a read of DP register 0x4 when the DPBANKSEL bit in the SELECT register is set to 0x2.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C C C C B B B B B B B B B B B B A A A A A A A A A A
Reset 0x30070289				**0 0 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A	R	TDESIGNER			An 11-bit code JEDEC JEP106 continuation code and identity code. The ID identifies the designer of the part.
			NordicSemi	0x144	Nordic Semiconductor ASA.
B	R	TPARTNO			Part number.
			nRF53	7	nRF53 Series.
C	R	TREVISION			Target revision.

8.1.1.2 DLPIDR

Address offset: 0x043

The DLPIDR register provides information about the serial wire debug protocol version.

Accessed by a read of DP register 0x4 when the DPBANKSEL bit in the SELECT register is set to 0x3.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B B B A A A A
Reset 0x00000001				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A	R	PROTVSN			Protocol version.
			SWDPv2	1	SW protocol version 2.
B	R	TINSTANCE			Target instance. This value is set by the UICR.TINSTANCE register.

8.1.2 Electrical specification

8.1.2.1 SW-DP

Symbol	Description	Min.	Typ.	Max.	Units
R_{pull}	Internal SWDIO and SWDCLK pull up/down resistance		13		kΩ
f_{SWDCLK}	SWDCLK frequency	0.125		8	MHz

8.1.2.2 Trace port

Symbol	Description	Min.	Typ.	Max.	Units
T_{cyc}	Clock period, as defined by Arm in Embedded Trace Macrocell Architecture Specification (see Timing specifications in Trace Port Physical Interface section)	15.625		250	ns

8.2 Access port protection

The control access ports are always accessible from the debugger, while access to the system resources through each core's individual access ports (AHB-AP) can be protected in different ways.

The following tables give an overview of the access port protection methods.

Registers	Description
UICR.APPROTECT and CTRL-AP.APPROTECT.DISABLE	These registers control the generation of the application core AHB-AP DBGEN signal, which controls all non-secure access through the application core AHB-AP. This can be used to provide readback protection of the flash contents. See also Application core access port protection for non-secure debug access on page 750. For more information about the DBGEN signal, see the *Arm CoreSight SoC-400 Technical Reference Manual, Revision r3p2*.
UICR.SECUREAPPROTECT and CTRL-AP.SECUREAPPROTECT.DISABLE	These registers control the generation of the application core AHB-AP SPIDEN signal, which blocks all secure access through the application core AHB-AP. This means that only the non-secure code can be debugged and accessed. To enable access to the secure access port, APPROTECT must be unprotected. See also Application core access port protection for secure debug access on page 750. For more information about the SPIDEN signal, see the *Arm CoreSight SoC-400 Technical Reference Manual, Revision r3p2*.
UICR.ERASEPROTECT and CTRL-AP.ERASEPROTECT.DISABLE	Disables the application core CTRL-AP.ERASEALL and NVMC ERASEALL functionality. This can be used together with APPROTECT to provide read-back and re-purposing protection.

Table 189: Application core access port protection overview

Registers	Description
UICR.APPROTECT and CTRL-AP.APPROTECT.DISABLE	These registers control the generation of the network core AHB-AP DBGEN signal, which blocks all access through the network core AHB-AP. See also Network core access port protection for debug access on page 750. For the network core that does not feature TrustZone, only DBGEN can be controlled and SPIDEN is not used.
UICR.ERASEPROTECT	Disables the network core CTRL-AP.ERASEALL and NVMC ERASEALL functionality. This can be used together with APPROTECT to provide read-back and re-purposing protection.

Table 190: Network core access port protection overview

For both cores, UICR and CTRL-AP are combined to enable or disable the access port protection. The access port is normally protected, and is opened when the following conditions are met:

1. UICR.APPROTECT must be `Unprotected`.
2. CTRL-AP.APPROTECT.DISABLE on both CPU and debugger side must match. However, after reset the debugger side register value is known and CPU can open the port by writing `Unprotected` to the register.

The following tables lists the available APPROTECT combinations.

Application core UICR.APPROTECT	CPU and debugger side CTRL-AP.APPROTECT.DISABLE registers are equal	DBGEN	Debug access to application core AHB-AP
Protected	No	0	Not permitted
Protected	Yes	0	Not permitted
Unprotected	No	0	Not permitted
Unprotected	Yes	1	Permitted

Table 191: Application core access port protection for non-secure debug access

Application core UICR.SECUREAPPROTECT	CPU and debugger side CTRL-AP.SECUREAP-PROTECT.DISABLE registers are equal	SPIDEN	Secure debug access to application core AHB-AP
Protected	No	0	Not permitted
Protected	Yes	0	Not permitted
Unprotected	No	0	Not permitted
Unprotected	Yes	1	Permitted

Table 192: Application core access port protection for secure debug access

Network core UICR.AP-PROTECT	CPU and debugger side CTRL-AP.APPROTECT registers are equal	DBGEN	Debug access to AHB-AP
Protected	No	0	Not permitted
Protected	Yes	0	Not permitted
Unprotected	No	0	Not permitted
Unprotected	Yes	1	Permitted

Table 193: Network core access port protection for debug access

The access port is also open after the completion of the CTRL-AP.ERASEALL operation. After completing the erase operation, CTRL-AP will temporarily unprotect AHB-AP. AHB-AP will be protected when one of the following conditions are met:

- Power-on reset
- Brown-out reset
- Watchdog timer reset
- Pin reset

The following figure is an example on how nRF5340 with access port protection enabled can be erased, programmed, and configured to allow debugging. Operations sent from debugger as well as registers written by firmware will affect the access port state. The operation named Reset* is one of the conditions listed above.

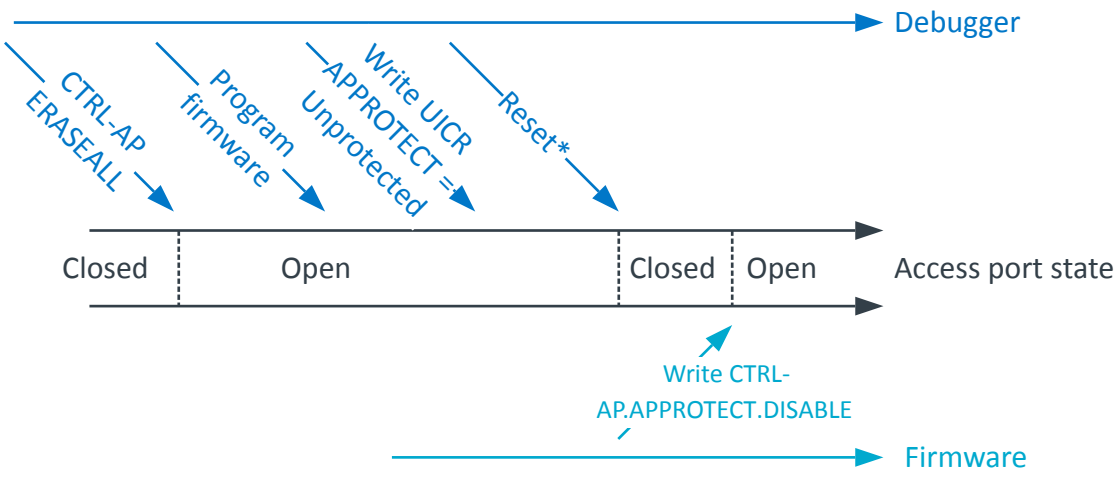


Figure 246: Access port unlocking

The debugger can read the access port protection status in the core's AHB-AP, using the Arm AHB-AP Control/Status Word register (CSW), defined in the *Arm CoreSight SoC-400 Technical Reference Manual, Revision r3p2*. The DbgStatus field indicates that the AHB-AP can perform AHB transfers, while the SPIStatus field indicates if secure AHB transfers are permitted. For a list of all debug access ports, see DAP — Debug access port on page 747.

For more details on CTRLAP.ERASEALL, CTRLAP.SECUREAPPROTECT, and CTRLAP.APPROTECT, see CTRL-AP - Control access port on page 757.

Note: Using SPU — System protection unit on page 585, the application core can be configured to grant the network core access to its resources. This grant also applies to the network core AHB-AP.

8.3 Debug Interface mode

Before the external debugger can use any of the access ports, the debugger must first request the device to power up via CxxxPWRUPREQ in the SWJ-DP.

As long as the debugger is requesting power via CxxxPWRUPREQ, the device will be in Debug interface mode. Otherwise, the device is in Normal mode. When a debug session is over, the external debugger must return the device back to Normal mode and perform a pin reset. This is due to the overall power consumption being higher in Debug interface mode compared to Normal mode.

Some peripherals behave differently in Debug interface mode compared to Normal mode. These differences are described in more detail in the chapters of the affected peripherals.

For details on how to use the debug capabilities, read the debug documentation of your IDE.

If the device is in System OFF when power is requested via CxxxPWRUPREQ, the device will wake up and the DIF flag in RESETREAS on page 66 will be set.

8.4 Real-time debug

The device supports real-time debugging, which allows interrupts to execute to completion in real time when breakpoints are set in Thread mode or lower priority interrupts.

Real-time debugging enables breakpoint setting and single-stepping through code without causing the failure of real-time event-driven threads running at higher priority. For example, this enables the device to

continue to service the high-priority interrupts of an external controller or sensor without failure or loss of state synchronization while stepping through code in a low-priority thread.

8.5 ROM tables

Each ROM Table on the SoC contains a listing of the components that are connected to the debug port or AHB-AP. These listings allow an external debugger or on-chip software to discover the CoreSight devices on the SoC.

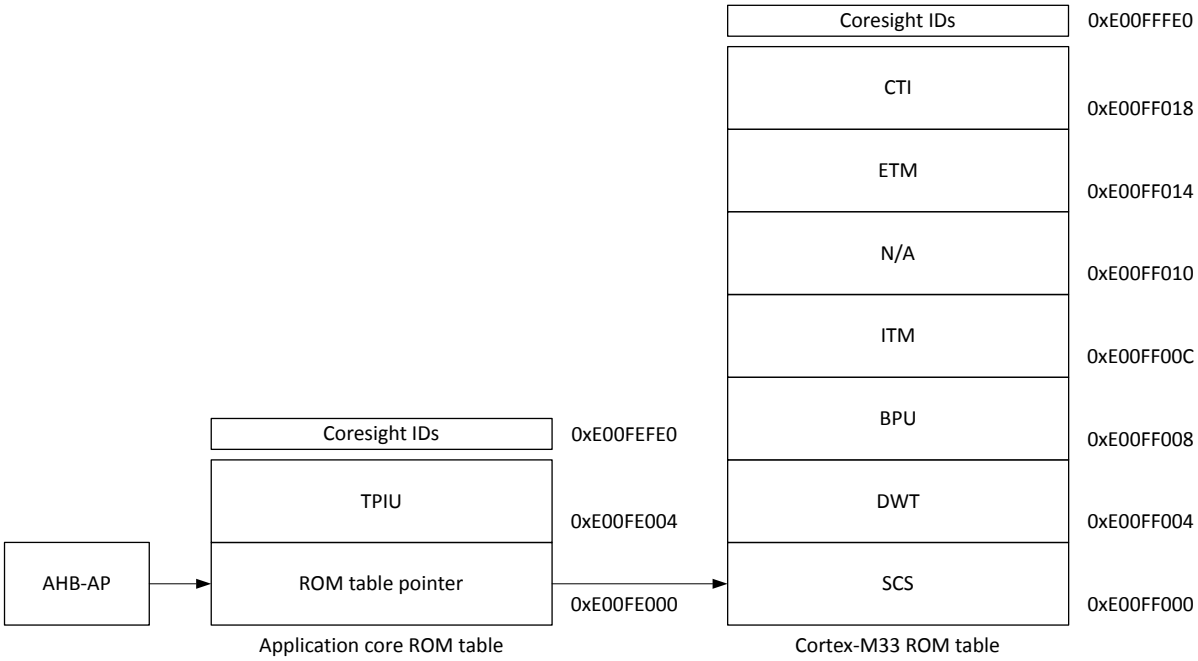


Figure 247: Application core ROM table overview

Address	Component	Value
0xE00FEFFC	CIDR3	0x000000B1
0xE00FEFF8	CIDR2	0x00000005
0xE00FEFF4	CIDR1	0x00000010
0xE00FEFF0	CIDR0	0x0000000D
0xE00FEFDC	PIDR7	0x00000000
0xE00FEFD8	PIDR6	0x00000000
0xE00FEFD4	PIDR5	0x00000000
0xE00FEFD0	PIDR4	0x00000002
0xE00FEFEC	PIDR3	0x00000000
0xE00FEFE8	PIDR2	0x0000001C
0xE00FEFE4	PIDR1	0x00000040
0xE00FEFE0	PIDR0	0x00000007
0xE00FEFCC	MEMTYPE	0x00000001
0xE00FE004	TPIU	0xFFF42003
0xE00FE000	ROM table	0x00001003

Table 194: Application core ROM table entries

Address	Component	Value
0xE00FF01C	MTB (not implemented)	0xFFF44002
0xE00FF018	CTI	0xFFF43003
0xE00FF014	ETM	0xFFF42003
0xE00FF00C	ITM	0xFFF01003
0xE00FF008	BPU	0xFFF03003
0xE00FF004	DWT	0xFFF02003
0xE00FF000	SCS	0xFFF0F003

Table 195: Application Arm Cortex-M33 ROM table entries

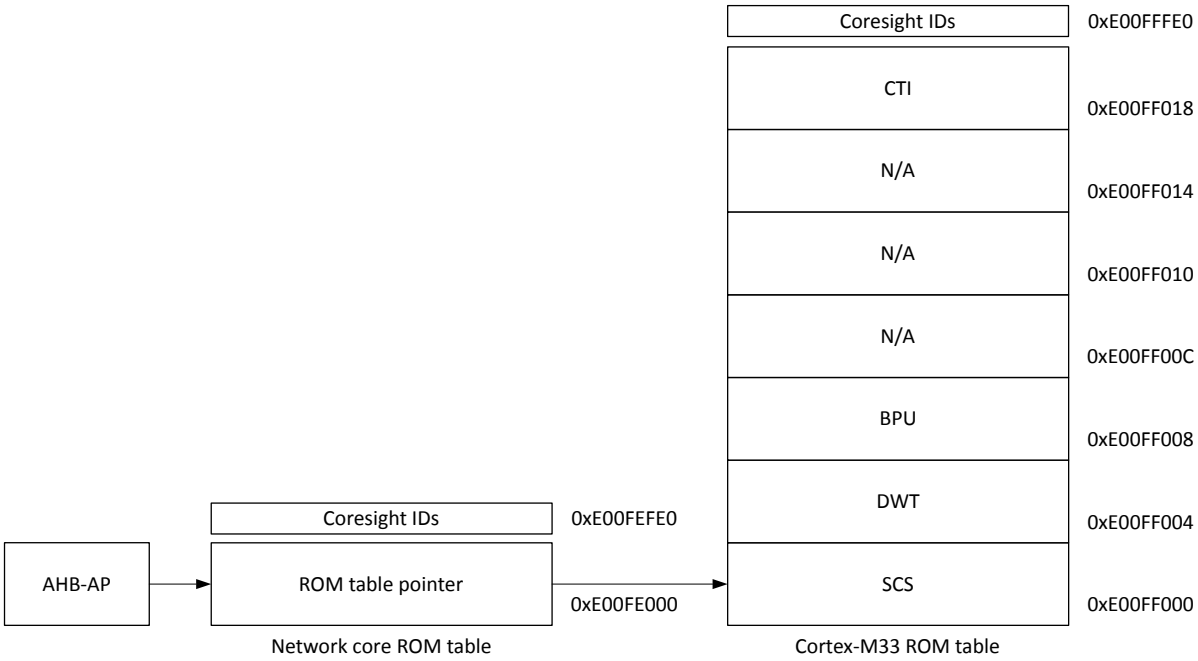


Figure 248: Network core ROM table overview

Address	Component	Value
0xE00FEFFC	CIDR3	0x000000B1
0xE00FEFF8	CIDR2	0x00000005
0xE00FEFF4	CIDR1	0x00000010
0xE00FEFF0	CIDR0	0x0000000D
0xE00FEFDC	PIDR7	0x00000000
0xE00FEFD8	PIDR6	0x00000000
0xE00FEFD4	PIDR5	0x00000000
0xE00FEFD0	PIDR4	0x00000002
0xE00FEFEC	PIDR3	0x00000000
0xE00FEFE8	PIDR2	0x0000001C
0xE00FEFE4	PIDR1	0x00000040
0xE00FEFE0	PIDR0	0x00000007
0xE00FEFCC	MEMTYPE	0x00000001
0xE00FE000	ROM table	0x00001003

Table 196: Network core ROM table entries

Address	Component	Value
0xE00FF01C	MTB (not implemented)	0xFFF44002
0xE00FF018	CTI	0xFFF43003
0xE00FF014	ETM (not implemented)	0xFFF42002
0xE00FF00C	ITM (not implemented)	0xFFF01002
0xE00FF008	BPU	0xFFF03003
0xE00FF004	DWT	0xFFF02003
0xE00FF000	SCS	0xFFF0F003

Table 197: Network Arm Cortex-M33 ROM table entries

8.6 Cross-trigger network

The debug system has a cross-trigger network used to simultaneously start and halt the cores in the system.

The following diagram shows an overview of the cross-trigger connections.

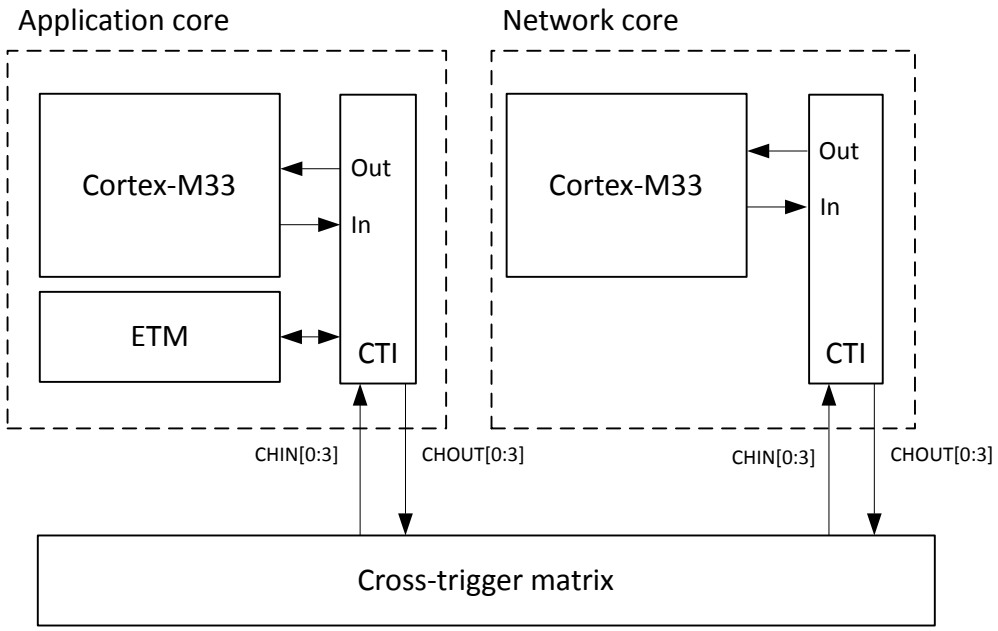


Figure 249: Cross-trigger network block diagram

Both the application and network cores have a cross-trigger interface (CTI) peripheral that can trigger events or be triggered by signals in the processor or debug blocks. The CTI can be configured to route trigger in-signals to trigger out-signals within the CTI or the cross-trigger matrix. The cross-trigger matrix has four channels in total that can be used to communicate trigger signals between cores.

You can stop the network core when the application core is stopped (due to a breakpoint or a stopped debug session), by doing the following:

1. Configure the application core CTI to generate an event on channel 0 for CTITRIGIN[0] (processor halted) using CTIINEN[0].
2. Configure the network core CTI to trigger CTITRIGOUT[0] (processor debug request) on channel 0 using CTIOUTEN[0].

Configuring the cross-trigger interface

In this example, the following CTI channels are used:

- Channel 0 is used to relay debug requests from the application to the network domain.

- Channel 1 is used to relay debug requests from the network to the application domain.
- Channel 2 is used by the debugger to send a common trigger for restarting both domains after a breakpoint.

For the application core, add the following code:

```
#define CTI_TRIGIN_CPUHALTED   0
#define CTI_TRIGOUT_DEBUGREQ   0
#define CTI_TRIGOUT_CPURESTART 1
...
// Enable global CTI routing
NRF_CTI_S->CTICONTROL = CTI_CTICONTROL_GLBEN_Enabled;
// Connect the CPU halted trigger of this domain to debug request of the other domain
NRF_CTI_S->CTIINEN[CTI_TRIGIN_CPUHALTED]    = CTI_CTIINEN_TRIGINEN_0_Msk;
NRF_CTI_S->CTIOUTEN[CTI_TRIGOUT_DEBUGREQ]   = CTI_CTIOUTEN_TRIGOUTEN_1_Msk;
NRF_CTI_S->CTIOUTEN[CTI_TRIGOUT_CPURESTART] = CTI_CTIOUTEN_TRIGOUTEN_2_Msk;
```

For the network core, add the following code:

```
#define CTI_TRIGIN_CPUHALTED   0
#define CTI_TRIGOUT_DEBUGREQ   0
#define CTI_TRIGOUT_CPURESTART 1
...
// Enable global CTI routing
NRF_CTI_NS->CTICONTROL = CTI_CTICONTROL_GLBEN_Enabled;
// Connect the CPU halted trigger of this domain to debug request of the other domain
NRF_CTI_NS->CTIINEN[CTI_TRIGIN_CPUHALTED]    = CTI_CTIINEN_TRIGINEN_1_Msk;
NRF_CTI_NS->CTIOUTEN[CTI_TRIGOUT_DEBUGREQ]   = CTI_CTIOUTEN_TRIGOUTEN_0_Msk;
NRF_CTI_NS->CTIOUTEN[CTI_TRIGOUT_CPURESTART] = CTI_CTIOUTEN_TRIGOUTEN_2_Msk;
```

See the following tables for more information about trigger connections to and from the CTI.

Signal	Description
CTITRIGIN[0]	Processor halted
CTITRIGIN[1]	DWT comparator output 0
CTITRIGIN[2]	DWT comparator output 1
CTITRIGIN[3]	DWT comparator output 2
CTITRIGIN[4]	ETM event output 0
CTITRIGIN[5]	ETM event output 1

Table 198: Application core triggers to CTI

Signal	Description
CTITRIGOUT[0]	Processor debug request
CTITRIGOUT[1]	Processor restart
CTITRIGOUT[2]	N/A
CTITRIGOUT[3]	N/A
CTITRIGOUT[4]	ETM event input 0
CTITRIGOUT[5]	ETM event input 1
CTITRIGOUT[6]	ETM event input 2
CTITRIGOUT[7]	ETM event input 3

Table 199: Application core triggers from CTI

Signal	Description
CTITRIGIN[0]	Processor halted
CTITRIGIN[1]	DWT comparator output 0
CTITRIGIN[2]	DWT comparator output 1
CTITRIGIN[3]	DWT comparator output 2

Table 200: Network core triggers to CTI

Signal	Description
CTITRIGOUT[0]	Processor debug request
CTITRIGOUT[1]	Processor restart

Table 201: Network core triggers from CTI

8.7 Multidrop serial wire debug

Multidrop serial wire debug allows simultaneous access to an unlimited number of devices through a single connection. This is useful for connectivity-constrained products that contain multiple chips with multidrop support.

In order to select a target in a multidrop capable product, the debugger must write the correct TINSTANCE, TPARTNO, and TDESIGNER fields into the SW-DP TARGETSEL register. The values for these fields are located in and fetched from two registers, TARGETID on page 747 and DLPIDR on page 748.

For more information about multidrop SWD, see *Arm Debug Interface Architecture Specification*, ADIv5.0 to ADIv5.2.

8.8 Trace

The device supports ETM and ITM trace.

Trace data from the ETM and the ITM is sent to an external debugger via a 4-bit wide parallel trace port (TPIU), see Debug and trace overview on page 746 (TRACEDATA[0] through TRACEDATA[3], and TRACECLK).

In addition to parallel trace, the TPIU supports serial trace via the serial wire output (SWO) trace protocol. Parallel and serial trace cannot be used at the same time. ETM trace is supported in Parallel trace mode only, while both parallel and Serial trace modes support the ITM trace. For details on how to use the trace capabilities, see the debug documentation of your IDE.

TPIU's dedicated trace pins are multiplexed with GPIOs. SWO and TRACEDATA[0] use the same GPIO. Trace is limited to dedicated pins. See Pin assignments on page 783 for more information.

Trace speed is configured in the register TRACEPORTSPEED (Retained) on page 782. The speed of the trace pins depends on the drive setting of the GPIOs that the trace pins are multiplexed with. The drive setting is configured using the DRIVE field of the GPIO register PIN_CNF[n] (n=0..31) (Retained) on page 229.

Only drive settings S0S1, H0H1, and E0E1 should be used for debugging. S0S1 is the default drive at reset. If parallel or serial trace port signals are not fast enough in the debugging conditions, all GPIOs in use for tracing should be set to high drive (H0H1), or extra high drive (E0E1) for the fastest trace port speeds. Ensure that the drive setting of the GPIOs are not overwritten by software during the debugging session.

In addition to DRIVE, the GPIO pin must be assigned to trace and debug (TND), using the MCUSEL field of the the PIN_CNF register. When pins are assigned to TND, these GPIOs are output-only, and other functionality of the pins is disabled.

8.9 Enabling the trace port

A specific sequence of operations must be performed to enable the trace port.

1. Enable the debug master.

```
NRF_TAD_S->ENABLE = TAD_ENABLE_ENABLE_Msk;
```

2. Request clock startup.

```
NRF_TAD_S->CLOCKSTART = TAD_CLOCKSTART_START_Msk;
```

3. Configure the trace port to use pins P0.08 through P0.12.

```
NRF_TAD_S->PSEL.TRACECLK   = TAD_PSEL_TRACECLK_PIN_Traceclk;
NRF_TAD_S->PSEL.TRACEDATA0 = TAD_PSEL_TRACEDATA0_PIN_Tracedata0;
NRF_TAD_S->PSEL.TRACEDATA1 = TAD_PSEL_TRACEDATA1_PIN_Tracedata1;
NRF_TAD_S->PSEL.TRACEDATA2 = TAD_PSEL_TRACEDATA2_PIN_Tracedata2;
NRF_TAD_S->PSEL.TRACEDATA3 = TAD_PSEL_TRACEDATA3_PIN_Tracedata3;
```

4. Configure the GPIO pins so that the trace and debug system can control them. Set high drive strength to ensure sufficiently fast operation. Do this for all trace pins that should be used.

```
// Clear the bitfields before configuring to make sure the correct value is written
NRF_P0_S->PIN_CNF[TAD_PSEL_TRACECLK_PIN_Traceclk]
                    &= ~(GPIO_PIN_CNF_MCUSEL_Msk | GPIO_PIN_CNF_DRIVE_Msk);
NRF_P0_S->PIN_CNF[TAD_PSEL_TRACECLK_PIN_Traceclk]
                    |= (GPIO_PIN_CNF_MCUSEL_TND << GPIO_PIN_CNF_MCUSEL_Pos)
                     | (GPIO_PIN_CNF_DRIVE_E0E1 << GPIO_PIN_CNF_DRIVE_Pos);
```

5. Set trace port speed to 64 MHz.

```
NRF_TAD_S->TRACEPORTSPEED = TAD_TRACEPORTSPEED_TRACEPORTSPEED_64MHz;
```

> **Note:** Although possible, it is not recommended to run the trace port at less than half the CPU frequency, as it risks dropping some trace packets.

6. Configure Arm CoreSight components (see Arm CoreSight documentation for more information).

8.10 CTRL-AP - Control access port

The control access port (CTRL-AP) is a custom access port that enables control of the device when other debug access ports (DAP) have been disabled by the access port protection.

For an overview of the other debug access ports, see DAP — Debug access port on page 747.

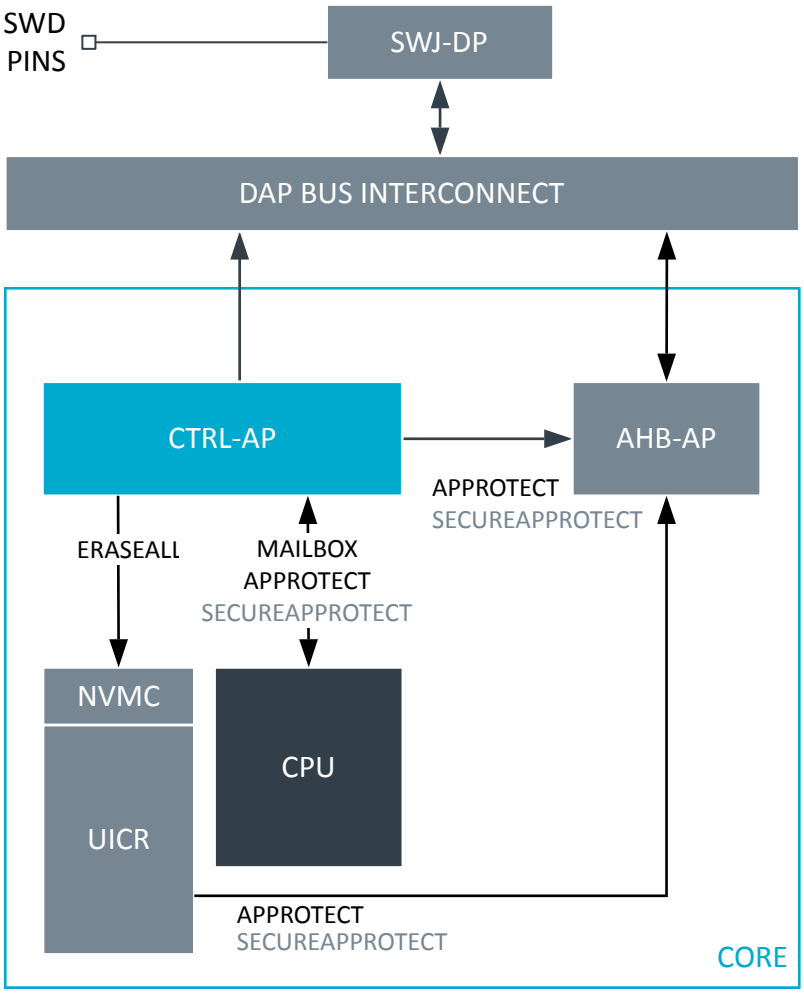


Figure 250: Control access port details

Access port protection (APPROTECT) blocks the debugger access to the AHB-AP, and prevents read and write access to all CPU registers and memory-mapped addresses. To enable port protection access for both secure and non-secure modes, use the registers UICR.SECUREAPPROTECT and UICR.APPROTECT , as well as CTRLAP.APPROTECT.DISABLE and CTRLAP.SECUREAPPROTECT.DISABLE. The debugger can use the register to read the status of secure and non-secure access port protection.

Erase protection (ERASEPROTECT) protects the flash and UICR parts of the non-volatile memory from being erased. Erase protection can be temporarily disabled from the control access port.

CTRL-AP has the following features:

- Soft reset
- Erase all
- Mailbox interface
- Debug of protected devices

8.10.1 Reset request

The debugger can request the device to perform a soft reset.

Use the register RESET on page 761 to request a soft reset. Once the soft reset is performed, the reset reason is accessible on the on-chip firmware through the RESETREAS register. For more information about the soft reset, see RESET — Reset control on page 61.

8.10.2 Erase all

The erase all function lets the debugger trigger an erase of flash, user information configuration registers (UICR), RAM, all peripheral settings, and also temporarily removes the access port protection.

To trigger an erase all function, the debugger writes to the register ERASEALL on page 762. The register ERASEALLSTATUS on page 762 will read as busy for the duration of the operation. The ERASEALL mechanism completes its tasks by writing UICR.APPROTECT to the `Unprotected` value, in addition to writing the CPU side CTRLAP.SECUREAPPROTECT.DISABLE and CTRLAP.APPROTECT.DISABLE registers to the value `0x50FA50FA`. After the next soft reset, the access port protection is temporarily removed. This temporary unprotection is removed by the next pin reset, power-on reset, brown-out reset, or watchdog timer reset. For more information about access port protection, see Access port protection on page 748.

If the debugger performs an erase all function on a slave MCU, the erase sequence will always erase the application MCU first, independently of how the application is protected, before erasing the slave MCU.

Erase all protection

It is possible to prevent the debugger from performing an erase all operation by writing to the UICR.ERASEPROTECT register. Once the register is configured and the device is reset, the CTRL-AP ERASEALL operation is disabled, and all flash write and erase operations are restricted to the firmware. In addition, it is still possible to write/erase from the debugger as long as the UICR.APPROTECT register is not set.

Note: Setting the UICR.ERASEPROTECT register only affects the erase all operation and not the debugger access.

The register ERASEPROTECT.STATUS on page 763 holds the status for erase protection.

8.10.3 Mailbox interface

CTRL-AP implements a mailbox interface which enables the CPU to communicate with a debugger over the SWD interface.

The mailbox interface consists of a transmit register MAILBOX.TXDATA on page 764 with its corresponding status register MAILBOX.TXSTATUS on page 764, and a receive register MAILBOX.RXDATA on page 764 with its corresponding status register MAILBOX.RXSTATUS on page 764. Status bits in the TXSTATUS/RXSTATUS registers are set and cleared automatically when the TXDATA/RXDATA registers are written to and read from, independently of the direction.

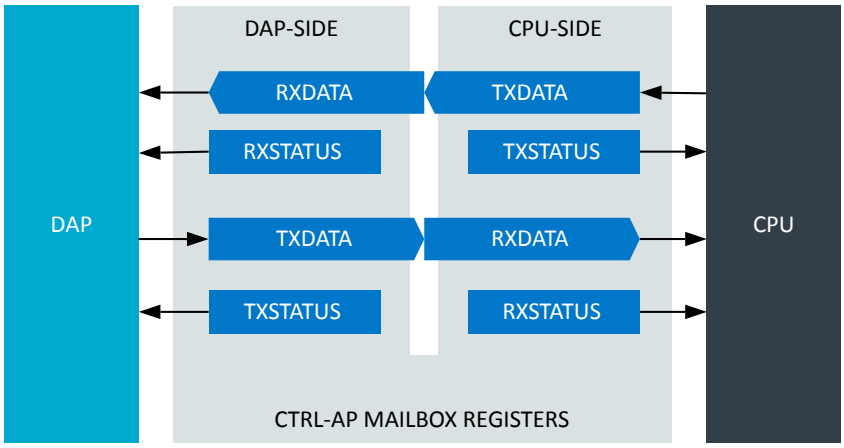


Figure 251: Mailbox register interface

Mailbox transfer sequence

1. Sender writes TXDATA.
2. Hardware sets sender's TXSTATUS to DataPending.
3. Hardware sets receiver's RXSTATUS to DataPending.
4. Receiver reads RXDATA.
5. Hardware sets receiver's RXSTATUS to NoDataPending.
6. Hardware sets sender's TXSTATUS to NoDataPending.

8.10.4 Disabling erase protection

The erase protection mechanism can be disabled to return a device to factory default settings on next reset.

The debugger can read the erase protection status in the register ERASEPROTECT.STATUS on page 763.

If ERASEPROTECT has been enabled, both the debugger and on-chip firmware must write the same non-zero 32-bit KEY value into their respective ERASEPROTECT.DISABLE registers to disable the erase protection. When both registers have been written with the same non-zero 32-bit KEY value, the device is automatically erased as described in Erase all on page 759. The access ports will be re-enabled on the next reset once the secure erase sequence has completed.

The write-once register ERASEPROTECT.LOCK on page 767 should be set to `Locked` as early as possible in the start-up sequence, preferably as soon as the on-chip firmware has determined it does not need to communicate with a debugger over the CTRL-AP mailbox interface. Once written, it will not be possible to remove the erase protection until the next reset.

8.10.5 Disabling access port protection

The access port protection mechanisms can be temporarily disabled to debug the device.

The disabling of the access port protection is done through a combination of UICR and CTRL-AP registers.

Disabling non-secure access port protection

If UICR.APPROTECT has been enabled from UICR, the device access port is protected.

If UICR.APPROTECT has not been enabled from UICR, both the debugger and on-chip firmware must write the same non-zero 32-bit KEY value into their respective registers CTRLAP.APPROTECT.DISABLE (CPU-side) and CTRLAP.APPROTECT.DISABLE (debugger-side) to disable the access port protection to non-secure mode.

The write-once register APPROTECT.LOCK on page 767 should be set to `Locked` as early as possible in the start-up sequence. Once written, it will not be possible to remove the non-secure mode access port protection until next reset.

Disabling secure access port protection

If UICR.APPROTECT has been enabled from UICR, the device access port is protected.

If UICR.SECUREAPPROTECT has not been enabled from UICR, both the debugger and on-chip firmware must write the same non-zero 32-bit KEY value into their respective registers CTRLAP.SECUREAPPROTECT.DISABLE (CPU-side) and CTRLAP.SECUREAPPROTECT.DISABLE (debugger-side) to disable the access port protection to secure mode.

The write-once register SECUREAPPROTECT.LOCK on page 768 should be set to `Locked` as early as possible in the start-up sequence, preferably as soon as on-chip firmware has determined it does not need to communicate with a debugger over the CTRL-AP mailbox interface. Once written, it will not be possible to remove the secure mode access port protection until next reset.

Note: If secure mode debug is enabled, an ERASEALL sequence can also be initiated by writing the same 32-bit KEY value into the respective ERASEPROTECT.DISABLE registers

The CTRLAP.APPROTECT.DISABLE and CTRLAP.SECUREAPPROTECT.DISABLE registers are only reset by pin reset, brown-out reset, or watchdog timer reset. This allows keeping the debug session active through soft resets.

After an ERASEALL sequence has completed, the access port protection of the core's AHB-AP is disabled until the next pin reset, power-on reset, brown-out reset, or watchdog timer reset. This will allow initial firmware to be written. For more details on ERASEALL, see Erase all on page 759.

Access port protection status

The debugger can read the access port protection status in the core's AHB-AP, using the Arm AHB-AP Control/Status Word register (CSW), defined in the *Arm CoreSight SoC-400 Technical Reference Manual, Revision r3p2*. The `DbgStatus` field indicates that the AHB-AP can perform AHB transfers, while the `SPIStatus` indicates if secure AHB transfers are permitted. For a list of all debug access ports, see DAP — Debug access port on page 747.

8.10.6 Debugger registers

CTRL-AP has a set of registers that can only be accessed from the debugger over the SWD interface. These are not accessible from the CPU.

The SECUREAPPROTECT fields and registers only apply for cores that have the Arm Cortex-M33 with TrustZone technology.

8.10.6.1 Registers

Register	Offset	Security	Description
RESET	0x000		System reset request.
ERASEALL	0x004		Perform a secure erase of the device, where flash, SRAM, and UICR will be erased in sequence. The device will be returned to factory default settings upon next reset.
ERASEALLSTATUS	0x008		This is the status register for the ERASEALL operation.
APPROTECT.DISABLE	0x010		This register disables APPROTECT and enables debug access to non-secure mode.
SECUREAPPROTECT.DISABLE	0x014		This register disables SECUREAPPROTECT and enables debug access to secure mode.
ERASEPROTECT.STATUS	0x018		This is the status register for the UICR ERASEPROTECT configuration.
ERASEPROTECT.DISABLE	0x01C		This register disables ERASEPROTECT and performs ERASEALL.
MAILBOX.TXDATA	0x020		Data sent from the debugger to the CPU.
MAILBOX.TXSTATUS	0x024		This register shows a status that indicates if data sent from the debugger to the CPU has been read.
MAILBOX.RXDATA	0x028		Data sent from the CPU to the debugger.
MAILBOX.RXSTATUS	0x02C		This register shows a status that indicates if data sent from the CPU to the debugger has been read.
IDR	0x0FC		CTRL-AP Identification Register, IDR.

Table 202: Register overview

8.10.6.1.1 RESET

Address offset: 0x000

System reset request.

This register is automatically deactivated during an ERASEALL operation.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	RESET			System reset request and status
			NoReset	0	Write to release reset
					Reading '0' means reset is not active
			Reset	1	Write to hold reset
					Reading '1' means reset is active

8.10.6.1.2 ERASEALL

Address offset: 0x004

Perform a secure erase of the device, where flash, SRAM, and UICR will be erased in sequence. The device will be returned to factory default settings upon next reset.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	ERASEALL			Return device to factory default settings
			NoOperation	0	No operation
			Erase	1	Erase flash, SRAM, and UICR in sequence

8.10.6.1.3 ERASEALLSTATUS

Address offset: 0x008

This is the status register for the ERASEALL operation.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	ERASEALLSTATUS			Status bit for the ERASEALL operation
			Ready	0	ERASEALL is ready
			Busy	1	ERASEALL is busy (on-going)

8.10.6.1.4 APPROTECT.DISABLE

Address offset: 0x010

This register disables APPROTECT and enables debug access to non-secure mode.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x50FA50FA				**0 1 0 1 0 0 0 0 1 1 1 1 1 0 1 0 0 1 0 1 0 0 0 0 1 1 1 1 1 0 1 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	KEY			Disable APPROTECT and enable debug access to non-secure mode until the next pin reset if KEY fields match. The current APPROTECT value as configured from CTRL-AP is disabled if the value of KEY fields are non-zero and the KEY fields match on both the CPU and debugger sides. To enable debug access, both CTRL-AP and UICR.APPROTECT protection needs to be disabled.

8.10.6.1.5 SECUREAPPROTECT.DISABLE

Address offset: 0x014

This register disables SECUREAPPROTECT and enables debug access to secure mode.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x50FA50FA				**0 1 0 1 0 0 0 0 1 1 1 1 1 0 1 0 0 1 0 1 0 0 0 0 1 1 1 1 1 0 1 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	KEY			Disable SECUREAPPROTECT and enable debug of secure mode until the next pin reset if KEY fields match. The current SECUREAPPROTECT value as configured from CTRL-AP is disabled if the value of KEY fields are non-zero and the KEY fields match on both the CPU and debugger sides. To enable debug access, both CTRL-AP and UICR.SECUREAPPROTECT protection needs to be disabled.

8.10.6.1.6 ERASEPROTECT.STATUS

Address offset: 0x018

This is the status register for the UICR ERASEPROTECT configuration.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	PALL			Status bit for erase protection **Note:** The reset value is auto read from the ERASEPROTECT register in UICR.
			Enabled	0	ERASEPROTECT is enabled
			Disabled	1	ERASEPROTECT is not enabled and ERASEALL can be performed

8.10.6.1.7 ERASEPROTECT.DISABLE

Address offset: 0x01C

This register disables ERASEPROTECT and performs ERASEALL.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	KEY			The ERASEALL sequence will be initiated if value of the KEY fields are non-zero and the KEY fields match on both the CPU and debugger sides.

8.10.6.1.8 MAILBOX.TXDATA

Address offset: 0x020

Data sent from the debugger to the CPU.

Writing to this register will automatically set a DataPending value in the TXSTATUS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	Data			Data sent from debugger

8.10.6.1.9 MAILBOX.TXSTATUS

Address offset: 0x024

This register shows a status that indicates if data sent from the debugger to the CPU has been read.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	Status			Status of register DATA
			NoDataPending	0	No data pending in register TXDATA
			DataPending	1	Data pending in register TXDATA

8.10.6.1.10 MAILBOX.RXDATA

Address offset: 0x028

Data sent from the CPU to the debugger.

Reading from this register will automatically set a NoDataPending value in the RXSTATUS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	Data			Data sent from CPU

8.10.6.1.11 MAILBOX.RXSTATUS

Address offset: 0x02C

This register shows a status that indicates if data sent from the CPU to the debugger has been read.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	Status			Status of register DATA
			NoDataPending	0	No data pending in register RXDATA
			DataPending	1	Data pending in register RXDATA

8.10.6.1.12 IDR

Address offset: 0x0FC

CTRL-AP Identification Register, IDR.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				E E E E D D D D C C C C C C B B B B A A A A A A A A
Reset 0x12880000				**0 0 0 1 0 0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	APID			AP Identification
B	R	CLASS			Access Port (AP) class
			NotDefined	0x0	No defined class
			MEMAP	0x8	Memory Access Port
C	R	JEP106ID			JEDEC JEP106 identity code
D	R	JEP106CONT			JEDEC JEP106 continuation code
E	R	REVISION			Revision

8.10.7 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0x50006000 0x40006000	APPLICATION	CTRLAPPERI	CTRLAP : S CTRLAP : NS	US	NSA	Control access port CPU side	
0x41006000	NETWORK	CTRLAPPERI	CTRLAP	NS	NA	Control access port CPU side	SECUREAPPROTECT.LOCK, SECUREAPPROTECT.DISABLI and STATUS.SECUREAPPROTECT registers not supported.

Table 203: Instances

Register	Offset	Security	Description
MAILBOX.RXDATA	0x400		Data sent from the debugger to the CPU.
MAILBOX.RXSTATUS	0x404		This register shows a status that indicates if data sent from the debugger to the CPU has been read.
MAILBOX.TXDATA	0x480		Data sent from the CPU to the debugger.
MAILBOX.TXSTATUS	0x484		This register shows a status that indicates if the data sent from the CPU to the debugger has been read.
ERASEPROTECT.LOCK	0x500		This register locks the ERASEPROTECT.DISABLE register from being written until next reset.
ERASEPROTECT.DISABLE	0x504		This register disables the ERASEPROTECT register and performs an ERASEALL operation.
APPROTECT.LOCK	0x540		This register locks the APPROTECT.DISABLE register from being written to until next reset.

Register	Offset	Security	Description
APPROTECT.DISABLE	0x544		This register disables the APPROTECT register and enables debug access to non-secure mode.
SECUREAPPROTECT.LOCK	0x548		This register locks the SECUREAPPROTECT.DISABLE register from being written until next reset.
SECUREAPPROTECT.DISABLE	0x54C		This register disables the SECUREAPPROTECT register and enables debug access to secure mode.
STATUS	0x600		Status bits for CTRL-AP peripheral.

Table 204: Register overview

8.10.7.1 MAILBOX.RXDATA

Address offset: 0x400

Data sent from the debugger to the CPU.

Reading from this register will automatically set a NoDataPending value in the RXSTATUS register.

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	R	RXDATA			Data received from debugger

8.10.7.2 MAILBOX.RXSTATUS

Address offset: 0x404

This register shows a status that indicates if data sent from the debugger to the CPU has been read.

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	R	RXSTATUS			Status of data in register RXDATA
			NoDataPending	0	No data pending in register RXDATA
			DataPending	1	Data pending in register RXDATA

8.10.7.3 MAILBOX.TXDATA

Address offset: 0x480

Data sent from the CPU to the debugger.

Writing to this register will automatically set a DataPending value in the TXSTATUS register.

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	RW	TXDATA			Data sent to debugger

8.10.7.4 MAILBOX.TXSTATUS

Address offset: 0x484

This register shows a status that indicates if the data sent from the CPU to the debugger has been read.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	TXSTATUS			Status of data in register TXDATA
			NoDataPending	0	No data pending in register TXDATA
			DataPending	1	Data pending in register TXDATA

8.10.7.5 ERASEPROTECT.LOCK

Address offset: 0x500

This register locks the ERASEPROTECT.DISABLE register from being written until next reset.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW1	LOCK			Lock ERASEPROTECT.DISABLE register from being written until next reset
			Unlocked	0	Register ERASEPROTECT.DISABLE is writeable
			Locked	1	Register ERASEPROTECT.DISABLE is read-only

8.10.7.6 ERASEPROTECT.DISABLE

Address offset: 0x504

This register disables the ERASEPROTECT register and performs an ERASEALL operation.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW1	KEY			The ERASEALL sequence is initiated if the value of the KEY fields are non-zero and the KEY fields match on both the CPU and debugger sides.

8.10.7.7 APPROTECT.LOCK

Address offset: 0x540

This register locks the APPROTECT.DISABLE register from being written to until next reset.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW1	LOCK			Lock the APPROTECT.DISABLE register from being written to until next reset
			Unlocked	0	Register APPROTECT.DISABLE is writeable
			Locked	1	Register APPROTECT.DISABLE is read-only

8.10.7.8 APPROTECT.DISABLE

Address offset: 0x544

This register disables the APPROTECT register and enables debug access to non-secure mode.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW	KEY			If the value of the KEY field is non-zero, and the KEY fields match on both the CPU and debugger sides, disable APPROTECT and enable debug access to non-secure mode until the next pin reset, brown-out reset, power-on reset, or watchog timer reset. After reset the debugger side register has a fixed KEY value. To enable debug access, both CTRL-AP and UICR.APPROTECT protection needs to be disabled.

8.10.7.9 SECUREAPPROTECT.LOCK

Address offset: 0x548

This register locks the SECUREAPPROTECT.DISABLE register from being written until next reset.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	RW1	LOCK			Lock register SECUREAPPROTECT.DISABLE from being written until next reset
			Unlocked	0	Register SECUREAPPROTECT.DISABLE is writeable
			Locked	1	Register SECUREAPPROTECT.DISABLE is read-only

8.10.7.10 SECUREAPPROTECT.DISABLE

Address offset: 0x54C

This register disables the SECUREAPPROTECT register and enables debug access to secure mode.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	KEY			If the value of the KEY field is non-zero, and the KEY fields match on both the CPU and debugger sides, disable SECUREAPPROTECT and enable debug access to secure mode until the next pin reset, brown-out reset, power-on reset, or watchog timer reset. After reset the debugger side register has a fixed KEY value. To enable debug access, both CTRL-AP and UICR.SECUREAPPROTECT protection needs to be disabled.

8.10.7.11 STATUS

Address offset: 0x600

Status bits for CTRL-AP peripheral.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C B A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	UICRAPPROTECT			Status bit for UICR part of access port protection at last reset. The reset value is automatically read from the APPROTECT register in UICR.
			Enabled	0	APPROTECT was enabled in UICR
			Disabled	1	APPROTECT wasdisabled in UICR
B	R	UICRSECUREAPPROTECT			Status bit for UICR part of secure access port protection at last reset. The reset value is automatically read from the SECUREAPPROTECT register in UICR.
			Enabled	0	SECUREAPPROTECT was enabled in UICR
			Disabled	1	SECUREAPPROTECT was disabled in UICR
C	R	DBGIFACEMODE			Status bit for device debug interface mode
			Disabled	0	No debugger attached
			Enabled	1	Debugger is attached and device is in debug interface mode

8.11 CTI - Cross Trigger Interface

Configuration interface for the Cross Trigger Interface

Please refer to the CTI section for more information about how to configure the Cross Trigger Interface.

8.11.1 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0xE0042000	APPLICATION	CTI	CTI	S	NA	Cross-trigger interface	Application core CTI
0xE0042000	NETWORK	CTI	CTI	NS	NA	Cross-trigger interface	Network core CTI

Table 205: Instances

Register	Offset	Security	Description
CTICONTROL	0x000		CTI Control register
CTIINTACK	0x010		CTI Interrupt Acknowledge register
CTIAPPSET	0x014		CTI Application Trigger Set register
CTIAPPCLEAR	0x018		CTI Application Trigger Clear register
CTIAPPPULSE	0x01C		CTI Application Pulse register
CTIINEN[n]	0x020		CTI Trigger to Channel Enable register
CTIOUTEN[n]	0x0A0		CTI Channel to Trigger Enable register
CTITRIGINSTATUS	0x130		CTI Trigger In Status register
CTITRIGOUTSTATUS	0x134		CTI Trigger Out Status register
CTICHINSTATUS	0x138		CTI Channel In Status register
CTIGATE	0x140		Enable CTI Channel Gate register
DEVARCH	0xFBC		Device Architecture register
DEVID	0xFC8		Device Configuration register
DEVTYPE	0xFCC		Device Type Identifier register
PIDR4	0xFD0		Peripheral ID4 Register
PIDR5	0xFD4		Peripheral ID5 register
PIDR6	0xFD8		Peripheral ID6 register
PIDR7	0xFDC		Peripheral ID7 register
PIDR0	0xFE0		Peripheral ID0 Register
PIDR1	0xFE4		Peripheral ID1 Register
PIDR2	0xFE8		Peripheral ID2 Register
PIDR3	0xFEC		Peripheral ID3 Register
CIDR0	0xFF0		Component ID0 Register
CIDR1	0xFF4		Component ID1 Register
CIDR2	0xFF8		Component ID2 Register
CIDR3	0xFFC		Component ID3 Register

Table 206: Register overview

8.11.1.1 CTICONTROL

Address offset: 0x000

CTI Control register

The CTICONTROL register enables the CTI.

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A
Reset 0x00000000	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

ID	R/W	Field	Value ID	Value	Description
A	RW	GLBEN			Enables or disables the CTI.
			Disabled	0	All cross-triggering mapping logic functionality is disabled.
			Enabled	1	Cross-triggering mapping logic functionality is enabled.

8.11.1.2 CTIINTACK

Address offset: 0x010

CTI Interrupt Acknowledge register

The CTIINTACK register is a software acknowledge for a trigger output. This register is used when ctitrigout is used as a sticky output. That is, no hardware acknowledge is available and a software acknowledge is required.

Bit number						31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																														H	G	F	E	D	C	B	A
Reset 0x00000000						**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
ID	R/W	Field		Value ID		Value								Description																							
A-H	W	INTACK[i] (i=0..7)												Acknowledges the ctitrigout i output.																							
				Acknowledge		1								Clears the ctitrigout.																							

8.11.1.3 CTIAPPSET

Address offset: 0x014

CTI Application Trigger Set register

Writing to the CTIAPPSET register causes a channel event to be raised, corresponding to the bit written to.

Bit number						31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																		D	C	B	A
Reset 0x00000000						**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
ID	R/W	Field		Value ID		Value								Description																							
A-D	RW	APPSET[i] (i=0..3)												Application trigger event for channel i.																							
				Inactive		0								Application trigger i is inactive.																							
				Active		1								Application trigger i is active.																							
				Activate		1								Generate channel event for channel i.																							

8.11.1.4 CTIAPPCLEAR

Address offset: 0x018

CTI Application Trigger Clear register

Writing to a bit in the CTIAPPCLEAR register clears the corresponding channel event.

Bit number						31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																		D	C	B	A
Reset 0x00000000						**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
ID	R/W	Field		Value ID		Value								Description																							
A-D	W	APPCLEAR[i] (i=0..3)												Sets the corresponding bits in the CTIAPPSET to 0. There is one bit of the register for each channel.																							
				Clear		1								Clears the event for channel i.																							

8.11.1.5 CTIAPPPULSE

Address offset: 0x01C

CTI Application Pulse register

A write to this register causes a channel event pulse of one cticlk period to be generated. This corresponds to the bit that was written to. The pulse external to the CTI can be extended to multi-cycle by the

handshaking interface circuits. This register clears itself immediately, so it can be repeatedly written to without software having to clear it.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					D C B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-D	W	APPULSE[i] (i=0..3)			Setting a bit HIGH generates a channel event pulse for the selected channel. There is one bit of the register for each channel.
			Generate	1	Generates an event pulse on channel i.

8.11.1.6 CTIINEN[n] (n=0..7)

Address offset: 0x020 + (n × 0x4)

CTI Trigger to Channel Enable register

The CTIINENn register enables the signaling of an event on CTM channels when a trigger event is received by the CTI. There is a bit for each of the four channels implemented. This register does not affect the application trigger operations.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					D C B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-D	RW	TRIGINEN[i] (i=0..3)			Enables a cross trigger event to channel i when a ctitrigin input is activated.
			Disabled	0	Input trigger n events are ignored by channel i.
			Enabled	1	When an event is received on input trigger n (ctitrigin[n]), generate an event on channel i.

8.11.1.7 CTIOUTEN[n] (n=0..7)

Address offset: 0x0A0 + (n × 0x4)

CTI Channel to Trigger Enable register

The CTIOUTENn register defines which channels can generate a ctitrigout[n] output. There is a bit for each of the four channels implemented. This register affects the mapping from application trigger to trigger outputs.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					D C B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-D	RW	TRIGOUTEN[i] (i=0..3)			Enables a cross trigger event to ctitrigout when channel i is activated.
			Disabled	0	Channel i is ignored by output trigger n.
			Enabled	1	When an event occurs on channel i, generate an event on output event n (ctitrigout[n]).

8.11.1.8 CTITRIGINSTATUS

Address offset: 0x130

CTI Trigger In Status register

The CTITRIGINSTATUS register provides the status of the ctitrigin inputs.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					H G F E D C B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-H	R	TRIGINSTATUS[i] (i=0..7)			Shows the status of ctitrigini input.
			Active	1	Ctitrigin i is active.
			Inactive	0	Ctitrigin i is inactive.

8.11.1.9 CTITRIGOUTSTATUS

Address offset: 0x134

CTI Trigger Out Status register

The CTITRIGOUTSTATUS register provides the status of the ctitrigout outputs.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					H G F E D C B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-H	R	TRIGOUTSTATUS[i] (i=0..7)			Shows the status of ctitrigouti output.
			Active	1	Ctitrigout i is active.
			Inactive	0	Ctitrigout i is inactive.

8.11.1.10 CTICHINSTATUS

Address offset: 0x138

CTI Channel In Status register

The CTICHINSTATUS register provides the status of the ctichin inputs.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID					D C B A
Reset 0x00000000					**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A-D	R	CTICHINSTATUS[i] (i=0..3)			Shows the status of the ctitrigin i input.
			Active	1	Ctichin i is active.
			Inactive	0	Ctichin i is inactive.

8.11.1.11 CTIGATE

Address offset: 0x140

Enable CTI Channel Gate register

The CTIGATE register prevents the channels from propagating through the CTM to other CTIs. This enables local cross-triggering (e.g. causing an interrupt when the ETM trigger occurs). It can be used effectively with CTIAPPSET, CTIAPPCLEAR, and CTIAPPPULSE for asserting trigger outputs by asserting channels, without affecting the rest of the system. On reset, this register is 0xF and channel propagation is enabled.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				D C B A
Reset 0x0000000F				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A-D	RW	CTIGATEEN[i] (i=0..3)			Enable ctichouti.
			Enabled	1	Enable ctichout channel i propagation.
			Disabled	0	Disable ctichout channel i propagation.

8.11.1.12 DEVARCH

Address offset: 0xFBC

Device Architecture register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x47701A14				**0 1 0 0 0 1 1 1 0 1 1 1 0 0 0 0 0 0 0 1 1 0 1 0 0 0 0 1 0 1 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	Architecture			Contains the CTI device architecture.

8.11.1.13 DEVID

Address offset: 0xFC8

Device Configuration register

The DEVID register indicates the capabilities of the component.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C C C C B B B B B B B B A A A A A
Reset 0x00040800				**0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	EXTMUXNUM			Indicates the number of multiplexers available on Trigger Inputs and Trigger Outputs that are using asicctl. The default value of 0b00000 indicates that no multiplexing is present.
B	R	NUMTRIG			Number of ECT triggers available.
C	R	NUMCH			Number of ECT channels available.

8.11.1.14 DEVTYPE

Address offset: 0xFCC

Device Type Identifier register

The DEVTYPE register provides a debugger with information about the component when the Part Number field is not recognized. The debugger can then report this information.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B B B A A A A
Reset 0x00000014				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	MAJOR			Major classification of the type of the debug component as specified in the Arm Architecture Specification for this debug and trace component.
			Controller	4	Indicates that this component allows a debugger to control other components in an Arm CoreSight SoC-400 system.
B	R	SUB			Sub-classification of the type of the debug component as specified in the Arm Architecture Specification within the major classification as specified in the MAJOR field.
			Crosstrigger	1	Indicates that this component is a sub-triggering component.

8.11.1.15 PIDR4

Address offset: 0xFD0

Peripheral ID4 Register

The PIDR4 register is part of the set of peripheral identification registers. It contains part of the designer identity and the memory size.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B B B A A A A
Reset 0x00000004				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	DES_2			Together, PIDR1.DES_0, PIDR2.DES_1, and PIDR4.DES_2 identify the designer of the component.
			Code	4	JEDEC continuation code.
B	R	SIZE			Always 0b0000. Indicates that the device only occupies 4KB of memory.

8.11.1.16 PIDR5

Address offset: 0xFD4

Peripheral ID5 register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description

8.11.1.17 PIDR6

Address offset: 0xFD8

Peripheral ID6 register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description

8.11.1.18 PIDR7

Address offset: 0xFDC

Peripheral ID7 register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description

8.11.1.19 PIDR0

Address offset: 0xFE0

Peripheral ID0 Register

The PIDR0 register is part of the set of peripheral identification registers. It contains part of the designer-specific part number.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000021				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1**
ID	R/W	Field	Value ID	Value	Description
A	R	PART_0			Bits[7:0] of the 12-bit part number of the component. The designer of the component assigns this part number.
			PartnumberL	0x21	Indicates bits[7:0] of the part number of the component.

8.11.1.20 PIDR1

Address offset: 0xFE4

Peripheral ID1 Register

The PIDR1 register is part of the set of peripheral identification registers. It contains part of the designer-specific part number and part of the designer identity.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B B B A A A A
Reset 0x000000BD				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 1**
ID	R/W	Field	Value ID	Value	Description
A	R	PART_1			Bits[11:8] of the 12-bit part number of the component. The designer of the component assigns this part number.
			PartnumberH	13	Indicates bits[11:8] of the part number of the component.
B	R	DES_0			Together, PIDR1.DES_0, PIDR2.DES_1, and PIDR4.DES_2 identify the designer of the component.
			Arm	11	Arm. Bits[3:0] of the JEDEC JEP106 Identity Code

8.11.1.21 PIDR2

Address offset: 0xFE8

Peripheral ID2 Register

The PIDR2 register is part of the set of peripheral identification registers. It contains part of the designer identity and the product revision.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				C C C C B A A A
Reset 0x0000000B				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1**
ID	R/W	Field	Value ID	Value	Description
A	R	DES_1			Together, PIDR1.DES_0, PIDR2.DES_1, and PIDR4.DES_2 identify the designer of the component.
			Arm	3	Arm. Bits[6:4] of the JEDEC JEP106 Identity Code
B	R	JEDEC			Always 1. Indicates that the JEDEC-assigned designer ID is used.
C	R	REVISION			Peripheral revision
			Rev0p0	0	This device is at r0p0

8.11.1.22 PIDR3

Address offset: 0xFEC

Peripheral ID3 Register

The PIDR3 register is part of the set of peripheral identification registers. It contains the REVAND and CMOD fields.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B B B A A A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	R	CMOD			Customer Modified. Indicates whether the customer has modified the behavior of the component. In most cases, this field is 0b0000. Customers change this value when they make authorized modifications to this component.
			Unmodified	0	Indicates that the customer has not modified this component.
B	R	REVAND			Indicates minor errata fixes specific to the revision of the component being used, for example metal fixes after implementation. In most cases, this field is 0b0000. Arm recommends that the component designers ensure that a metal fix can change this field if required, for example, by driving it from registers that reset to 0b0000.
			NoErrata	0	Indicates that there are no errata fixes to this component.

8.11.1.23 CIDR0

Address offset: 0xFF0

Component ID0 Register

The CIDR0 register is a component identification register that indicates the presence of identification registers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x0000000D				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1**
ID	R/W	Field	Value ID	Value	Description
A	R	PRMBL_0			Preamble[0]. Contains bits[7:0] of the component identification code.
			Value	0x0D	Bits[7:0] of the identification code.

8.11.1.24 CIDR1

Address offset: 0xFF4

Component ID1 Register

The CIDR1 register is a component identification register that indicates the presence of identification registers. This register also indicates the component class.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B B B B A A A A
Reset 0x00000090				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	R	PRMBL_1			Preamble[1]. Contains bits[11:8] of the component identification code.
			Value	0	Bits[11:8] of the identification code.
B	R	CLASS			Class of the component, for example, whether the component is a ROM table or a generic CoreSight component. Contains bits[15:12] of the component identification code
			Coresight	9	Indicates that the component is a CoreSight component.

8.11.1.25 CIDR2

Address offset: 0xFF8

Component ID2 Register

The CIDR2 register is a component identification register that indicates the presence of identification registers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x00000005				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1**

ID	R/W	Field	Value ID	Value	Description
A	R	PRMBL_2			Preamble[2]. Contains bits[23:16] of the component identification code.
			Value	0x05	Bits[23:16] of the identification code.

8.11.1.26 CIDR3

Address offset: 0xFFC

Component ID3 Register

The CIDR3 register is a component identification register that indicates the presence of identification registers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A A A A A A A
Reset 0x000000B1				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 0 0 1**

ID	R/W	Field	Value ID	Value	Description
A	R	PRMBL_3			Preamble[3]. Contains bits[31:24] of the component identification code.
			Value	0xB1	Bits[31:24] of the identification code.

8.12 TAD - Trace and debug control

Configuration interface for trace and debug

Please refer to the Trace section for more information about how to configure the trace and debug interface.

Note: Although there are PSEL registers for the trace port, each function can only be mapped to a single pin due to pin speed requirements. Setting the PIN field to anything else will not have any effect. See Pin assignment chapter for more information

8.12.1 Registers

Base address	Domain	Peripheral	Instance	Secure mapping	DMA security	Description	Configuration
0xE0080000	APPLICATION	TAD	TAD	S	NA	Trace and debug control	

Table 207: Instances

Register	Offset	Security	Description	
CLOCKSTART	0x004		Start all trace and debug clocks.	
CLOCKSTOP	0x008		Stop all trace and debug clocks.	
ENABLE	0x500		Enable debug domain and aquire selected GPIOs	
PSEL.TRACECLK	0x504		Pin configuration for TRACECLK	
PSEL.TRACEDATA0	0x508		Pin configuration for TRACEDATA[0]	
PSEL.TRACEDATA1	0x50C		Pin configuration for TRACEDATA[1]	
PSEL.TRACEDATA2	0x510		Pin configuration for TRACEDATA[2]	
PSEL.TRACEDATA3	0x514		Pin configuration for TRACEDATA[3]	
TRACEPORTSPEED	0x518		Clocking options for the Trace Port debug interface Reset behavior is the same as debug components	Retained

Table 208: Register overview

8.12.1.1 CLOCKSTART

Address offset: 0x004

Start all trace and debug clocks.

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID	A
Reset 0x00000000	**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**

ID	R/W	Field	Value ID	Value	Description
A	W	START			
			Start	1	Start all trace and debug clocks.

8.12.1.2 CLOCKSTOP

Address offset: 0x008

Stop all trace and debug clocks.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	W	STOP			
			Stop	1	Stop all trace and debug clocks.

8.12.1.3 ENABLE

Address offset: 0x500

Enable debug domain and aquire selected GPIOs

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	ENABLE			
			DISABLED	0	Disable debug domain and release selected GPIOs
			ENABLED	1	Enable debug domain and aquire selected GPIOs

8.12.1.4 PSEL.TRACECLK

Address offset: 0x504

Pin configuration for TRACECLK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN			Pin number
			Traceclk	12	TRACECLK pin
					Note: Only this pin is valid
B	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

8.12.1.5 PSEL.TRACEDATA0

Address offset: 0x508

Pin configuration for TRACEDATA[0]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN			Pin number
			Tracedata0	11	TRACEDATA0 pin
					Note: Only this pin is valid
B	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

8.12.1.6 PSEL.TRACEDATA1

Address offset: 0x50C

Pin configuration for TRACEDATA[1]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN			Pin number
			Tracedata1	10	TRACEDATA1 pin
					Note: Only this pin is valid
B	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

8.12.1.7 PSEL.TRACEDATA2

Address offset: 0x510

Pin configuration for TRACEDATA[2]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN			Pin number
			Tracedata2	9	TRACEDATA2 pin
					Note: Only this pin is valid
B	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

8.12.1.8 PSEL.TRACEDATA3

Address offset: 0x514

Pin configuration for TRACEDATA[3]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				B A A A A A
Reset 0xFFFFFFFF				**1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1**
ID	R/W	Field	Value ID	Value	Description
A	RW	PIN			Pin number
			Tracedata3	8	TRACEDATA3 pin
					Note: Only this pin is valid
B	RW	CONNECT			Connection
			Disconnected	1	Disconnect
			Connected	0	Connect

8.12.1.9 TRACEPORTSPEED (Retained)

Address offset: 0x518

This register is a retained register

Clocking options for the Trace Port debug interface

Reset behavior is the same as debug components

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
ID				A A
Reset 0x00000000				**0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0**
ID	R/W	Field	Value ID	Value	Description
A	RW	TRACEPORTSPEED			Speed of Trace Port clock. Note that the TRACECLK pin output will be divided again by two from the Trace Port clock.
			64MHz	0	Trace Port clock is: 64MHz
			32MHz	1	Trace Port clock is: 32MHz
			16MHz	2	Trace Port clock is: 16MHz
			8MHz	3	Trace Port clock is: 8MHz

9 Hardware and layout

9.1 Pin assignments

This section describes the pin assignment and the pin functions.

This device provides flexibility when it comes to routing and configuration of the GPIO pins. However, some pins have recommendations for how the pin should be configured or what it should be used for.

In addition to the information in the pinout tables for the respective packages, the following peripherals have dedicated pins that should be used for proper operation:

- TWI - For the fastest TWI 1 Mbps mode, the two high-speed TWI pins must be configured in the TWI peripheral's PSEL registers, and the 20 mA open drain driver enabled using the E0E1 drive setting in the DRIVE field of the PIN_CNF GPIO register.
- QSPI - For QSPI only the dedicated GPIO pins from the following table shall be used. These must be enabled using the Peripheral option of the PIN_CNF[p].MCUSEL register. The GPIO must use the high drive H0H1 configuration in the DRIVE field of the PIN_CNF GPIO register.
- SPIM4 - For the 32 Mbps SPI mode, the special purpose GPIO pins are enabled using the Peripheral option of the PIN_CNF[p].MCUSEL register. When activated, the SPIM PSEL settings are ignored, and the dedicated pins are used. The GPIO must use the high drive H0H1 configuration in the DRIVE field of the PIN_CNF GPIO register.
- TRACE - When using trace, the TRACEDATA[n] and TRACECLK GPIO pins must all use the extra high drive E0E1 configuration in the DRIVE field of the PIN_CNF GPIO register. Also, the TND option of the PIN_CNF[p].MCUSEL register must be used.

GPIO pin	Description
P0.08 - P0.12	Drive configuration E0E1 is available and must be used for TRACE. For 32 Mbps high-speed SPI using SPIM4, drive configuration H0H1 must be used.
P0.13 - P0.18	The H0H1 drive configuration features the highest speeds of quad SPI using the direct connection of the QSPI peripheral.
P1.02 and P1.03	The E0E1 drive configuration activates a 20 mA open-drain driver specifically designed for high-speed TWI.
Remaining pins	The E0E1 drive configuration is not supported. Using the E0E1 drive configuration will cause incorrect operation.

Table 209: Special GPIO considerations

Note: The extra high drive E0E1 drive configuration has limited availability. It is only available for the dedicated TRACE pins on P0.08 through P0.12. For the dedicated, high-speed TWIM pins on P1.02 and P1.03, the E0E1 drive configuration activates a powerful 20 mA *open-drain* driver specifically designed for high-speed TWI.

For all high-speed signals, the printed circuit board (PCB) layout must ensure that connections are made using short PCB traces. Refer to the manufacturer's PCB design recommendations for additional information.

9.1.1 aQFN94 pin assignments

The aQFN94 package has 94 pins in addition to four corner pads and a die pad.

Figure 252: aQFN pin assignments, top view. Corner and die pad is not illustrated.

Pin	Name	Function	Description	Recommended usage
A5	VBUS	Power	5 V input for USB 3.3 V regulator	
A13	DECA	Power	Analog regulator supply decoupling	
A15	DECD	Power	Digital regulator supply decoupling	
A17	P1.13	Digital I/O	General purpose I/O	
A19	VDD	Power	Power supply	
A21	DCC	Power	DC/DC converter output	
A23	DECN	Power	Regulator supply decoupling	
A25	N.C.			
A27	DECR	Power	Regulator supply decoupling	
B2	D+	USB	USB D+	
B4	D-	USB	USB D-	
B6	DECUSB	Power	USB 3.3 V regulator supply decoupling	
B8	VDD	Power	Power supply	
B10	DCCD	Power	DC/DC converter output	
B12	N.C.			
B14	P1.15	Digital I/O	General purpose I/O	
B16	P1.14	Digital I/O	General purpose I/O	
B18	P1.12	Digital I/O	General purpose I/O	
B20	P1.11	Digital I/O	General purpose I/O	
B22	P0.31	Digital I/O	General purpose I/O	
B24	P0.30	Digital I/O	General purpose I/O	
B26	N.C.			
B28	VDD	Power	Power supply	
B30	XC2	Analog input	Connection for 32 MHz crystal	
C1	VDD	Power	Power supply	
C31	XC1	Analog input	Connection for 32 MHz crystal	
D2	N.C.			
E1	VDDH	Power	Power supply	
E31	VDD	Power	Power supply	
F2	N.C.			
G1	N.C.			
G31	DECRF	Power	RADIO power supply decoupling	
H2	N.C.			
J1	DCCH	Power	DC/DC converter output	
J31	N.C.			
K2	N.C.			
L1	VDD	Power	Power supply	
L31	ANT	RF	Single-ended antenna connection	
M2	P1.00	Digital I/O	General purpose I/O	
N1	P0.00	Digital I/O	General purpose I/O	
	XL1	Analog input	Connection for 32 kHz crystal	
N31	VDD	Power	Power supply	
P2	P1.01	Digital I/O	General purpose I/O	
R1	P0.01	Digital I/O	General purpose I/O	
	XL2	Analog input	Connection for 32 kHz crystal	
R31	P1.10	Digital I/O	General purpose I/O	
T2	N.C.			
U1	VDD	Power	Power supply	
U31	P0.29	Digital I/O	General purpose I/O	
V2	P0.04	Digital I/O	General purpose I/O	
	AIN0	Analog input	Analog input	
W1	P0.02	Digital I/O	General purpose I/O	
	NFC1	NFC input	NFC antenna connection	

Pin	Name	Function	Description	Recommended usage
W31	SWDCLK	Debug	Serial wire debug clock input for debug and programming	
Y2	P0.05	Digital I/O	General purpose I/O	
	AIN1	Analog input	Analog input	
AA1	P0.03	Digital I/O	General purpose I/O	
	NFC2	NFC input	NFC antenna connection	
AA31	SWDIO	Debug	Serial wire debug I/O for debug and programming	
AB2	P0.06	Digital I/O	General purpose I/O	
	AIN2	Analog input	Analog input	
AC1	VDD	Power	Power supply	
AC31	nRESET	Reset	Pin RESET with internal pull-up resistor	
AD2	P0.07	Digital I/O	General purpose I/O	
	AIN3	Analog input	Analog input	
AE1	P1.02	Digital I/O	General purpose I/O	TWI
	TWI	TWI 1 Mbps	High-speed pin for 1 Mbps TWI	
AE31	P0.28	Digital I/O	General purpose I/O	
	AIN7	Analog input	Analog input	
AF2	P1.03	Digital I/O	General purpose I/O	TWI
	TWI	TWI 1 Mbps	High-speed pin for 1 Mbps TWI	
AG1	VDD	Power	Power supply	
AG31	N.C.			
AH2	P0.08	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACEDATA3	Trace data	Trace buffer TRACEDATA[3]	
	SCK	SCK for SPIM4	Dedicated pin for high-speed SPI	
AJ1	P0.09	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACEDATA2	Trace data	Trace buffer TRACEDATA[2]	
	MOSI	MOSI for SPIM4	Dedicated pin for high-speed SPI	
AJ31	VDD	Power	Power supply	
AK2	P0.10	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACEDATA1	Trace data	Trace buffer TRACEDATA[1]	
	MISO	MISO for SPIM4	Dedicated pin for high-speed SPI	
AK4	P0.11	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACEDATA0	Trace data	Trace buffer TRACEDATA[0]	
	CSN	CSN for SPIM4	Dedicated pin for high-speed SPI	
AK6	P0.12	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACECLK	Trace clock	Trace buffer clock	
	DCX	DCX for SPIM4	Dedicated pin for high-speed SPI	
AK8	P0.14	Digital I/O	General purpose I/O	QSPI
	IO1	IO1 for QSPI	Dedicated pin for Quad SPI	
AK10	P0.15	Digital I/O	General purpose I/O	QSPI
	IO2	IO2 for QSPI	Dedicated pin for Quad SPI	
AK12	P0.17	Digital I/O	General purpose I/O	QSPI
	SCK	SCK for QSPI	Dedicated pin for Quad SPI	
AK14	P0.18	Digital I/O	General purpose I/O	QSPI
	CSN	CSN for QSPI	Dedicated pin for Quad SPI	
AK16	P0.20	Digital I/O	General purpose I/O	
AK18	P0.22	Digital I/O	General purpose I/O	
AK20	P0.23	Digital I/O	General purpose I/O	
AK22	P1.05	Digital I/O	General purpose I/O	
AK24	P1.07	Digital I/O	General purpose I/O	
AK26	P1.09	Digital I/O	General purpose I/O	
AK28	P0.25	Digital I/O	General purpose I/O	
	AIN4	Analog input	Analog input	
AK30	P0.27	Digital I/O	General purpose I/O	

Pin	Name	Function	Description	Recommended usage
	AIN6	Analog input	Analog input	
AL3	VDD	Power	Power supply	
AL5	P0.13	Digital I/O	General purpose I/O	QSPI
	IO0	IO0 for QSPI	Dedicated pin for Quad SPI	
AL7	VDD	Power	Power supply	
AL9	P0.16	Digital I/O	General purpose I/O	QSPI
	IO3	IO3 for QSPI	Dedicated pin for Quad SPI	
AL11	VDD	Power	Power supply	
AL13	P0.19	Digital I/O	General purpose I/O	
AL15	P0.21	Digital I/O	General purpose I/O	
AL17	VDD	Power	Power supply	
AL19	P1.04	Digital I/O	General purpose I/O	
AL21	P1.06	Digital I/O	General purpose I/O	
AL23	P1.08	Digital I/O	General purpose I/O	
AL25	VDD	Power	Power supply	
AL27	P0.24	Digital I/O	General purpose I/O	
AL29	P0.26	Digital I/O	General purpose I/O	
	AIN5	Analog input	Analog input	
Corner pads				
A1	N.C.			
A31	N.C.			
AL1	N.C.			
AL31	N.C.			
Bottom of chip				
Die pad	VSS	Power	Ground pad. Exposed die pad must be connected to ground (VSS) for proper device operation.	

Table 210: aQFN pin assignments

9.1.2 WLCSP pin assignments

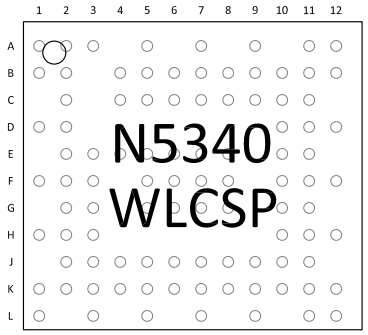


Figure 253: WLCSP pin assignments, top view

Pin	Name	Function	Description	Recommended usage
A1	XC1	Analog input	Connection for 32 MHz crystal	
A2	XC2	Analog input	Connection for 32 MHz crystal	
A3	VDD	Power	Power supply	
A5	VSS	Power	Ground	
A7	DECD	Power	Regulator supply decoupling	
A9	DCCD	Power	DC/DC converter output	
A11	D-	USB	USB D-	
A12	D+	USB	USB D+	
B1	DECRF	Power	RADIO power supply decoupling	
B2	VSS	Power	Ground	
B4	DECR	Power	Regulator supply decoupling	
B5	DECN	Power	Regulator supply decoupling	
B6	DCC	Power	DC/DC converter output	
B7	VDD	Power	Power supply	
B8	DECA	Power	Regulator supply decoupling	
B9	VSS	Power	Ground	
B10	VDD	Power	Power supply	
B11	VBUS	Power	Power	
B12	VDDH	Power	Power supply	
C2	VSS	Power	Ground	
C4	P0.30	Digital I/O	General purpose I/O	
C5	P0.31	Digital I/O	General purpose I/O	
C6	P1.11	Digital I/O	General purpose I/O	
C7	P1.12	Digital I/O	General purpose I/O	
C8	P1.13	Digital I/O	General purpose I/O	
C9	P1.14	Digital I/O	General purpose I/O	
C10	P1.15	Digital I/O	General purpose I/O	
C11	DECUSB	Power	USB 3.3 V regulator supply decoupling	
D1	ANT	RF	Single-ended antenna connection	
D2	VDD	Power	Power supply	
D10	P1.00	Digital I/O	General purpose I/O	
D11	DCCH	Power	DC/DC converter output	
D12	VSS	Power	Ground	
E2	P0.29	Digital I/O	General purpose I/O	
E3	P0.28	Digital I/O	General purpose I/O	
	AIN7	Analog input	Analog input	
E4	P1.10	Digital I/O	General purpose I/O	
E5	VSS	Power	Ground	
E6	VSS	Power	Ground	
E7	VSS	Power	Ground	
E8	VSS	Power	Ground	
E10	P1.01	Digital I/O	General purpose I/O	
E11	VDD	Power	Power supply	
F1	SWDIO	Debug	Serial wire debug I/O for debug and programming	
F2	SWDCLK	Debug	Serial wire debug clock input for debug and programming	
F3	P1.08	Digital I/O	General purpose I/O	
F5	VSS	Power	Ground	
F6	VSS	Power	Ground	
F7	VSS	Power	Ground	
F8	VSS	Power	Ground	
F10	P0.05	Digital I/O	General purpose I/O	
	AIN1	Analog input	Analog input	
F11	P0.00	Digital I/O	General purpose I/O	

Pin	Name	Function	Description	Recommended usage
	XL1	Analog input	Connection for 32 kHz crystal	
F12	P0.01	Digital I/O	General purpose I/O	
	XL2	Analog input	Connection for 32 kHz crystal	
G2	nRESET	Reset	Pin RESET with internal pull-up resistor	
G3	P1.07	Digital I/O	General purpose I/O	
G5	AVSS	Power	Ground	
G6	VSS	Power	Ground	
G7	VSS	Power	Ground	
G8	VSS	Power	Ground	
G10	P0.04	Digital I/O	General purpose I/O	
	AIN0	Analog input	Analog input	
G11	P0.02	Digital I/O	General purpose I/O	
	NFC1	NFC input	NFC antenna connection	
H1	P0.27	Digital I/O	General purpose I/O	
	AIN6	Analog input	Analog input	
H2	P1.09	Digital I/O	General purpose I/O	
H3	P0.23	Digital I/O	General purpose I/O	
H10	P0.06	Digital I/O	General purpose I/O	
	AIN2	Analog input	Analog input	
H11	VDD	Power	Power supply	
H12	P0.03	Digital I/O	General purpose I/O	
	NFC2	NFC input	NFC antenna connection	
J2	P0.26	Digital I/O	General purpose I/O	
	AIN5	Analog input	Analog input	
J3	P1.06	Digital I/O	General purpose I/O	
J4	P0.21	Digital I/O	General purpose I/O	
J5	P0.19	Digital I/O	General purpose I/O	
J6	P0.12	Digital I/O	General purpose I/O	
	TRACECLK	Trace clock	Trace buffer clock	
	DCX	DCX for SPIM4	Dedicated pin for high-speed SPI	
J7	P0.11	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACEDATA0	Trace data	Trace buffer TRACEDATA[0]	
	CSN	CSN for SPIM4	Dedicated pin for high-speed SPI	
J8	P0.10	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACEDATA1	Trace data	Trace buffer TRACEDATA[1]	
	MISO	MISO for SPIM4	Dedicated pin for high-speed SPI	
J9	P0.09	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACEDATA2	Trace data	Trace buffer TRACEDATA[2]	
	MOSI	MOSI for SPIM4	Dedicated pin for high-speed SPI	
J10	P0.07	Digital I/O	General purpose I/O	
	AIN3	Analog input	Analog input	
J11	P1.02	Digital I/O	General purpose I/O	TWI
	TWI	TWI 1 Mbps	High-speed pin for 1 Mbps TWI	
K1	VDD	Power	Power supply	
K2	P0.24	Digital I/O	General purpose I/O	
K3	P1.04	Digital I/O	General purpose I/O	
K4	P0.22	Digital I/O	General purpose I/O	
K5	P0.20	Digital I/O	General purpose I/O	
K6	AVSS	Power	Ground	
K7	P0.18	Digital I/O	General purpose I/O	QSPI
	CSN	CSN for QSPI	Dedicated pin for Quad SPI	
K8	P0.16	Digital I/O	General purpose I/O	QSPI
	IO3	IO3 for QSPI	Dedicated pin for Quad SPI	

Pin	Name	Function	Description	Recommended usage
K9	P0.14	Digital I/O	General purpose I/O	QSPI
	IO1	IO1 for QSPI	Dedicated pin for Quad SPI	
K10	P0.13	Digital I/O	General purpose I/O	QSPI
	IO0	IO0 for QSPI	Dedicated pin for Quad SPI	
K11	AVSS	Power	Ground	
K12	P1.03	Digital I/O	General purpose I/O	TWI
	TWI	TWI 1 Mbps	High-speed pin for 1 Mbps TWI	
L1	P0.25	Digital I/O	General purpose I/O	
	AIN4	Analog input	Analog input	
L3	P1.05	Digital I/O	General purpose I/O	
L5	VDD	Power	Power supply	
L7	P0.17	Digital I/O	General purpose I/O	QSPI
	SCK	SCK for QSPI	Dedicated pin for Quad SPI	
L9	P0.15	Digital I/O	General purpose I/O	QSPI
	IO2	IO2 for QSPI	Dedicated pin for Quad SPI	
L11	VDD	Power	Power supply	
L12	P0.08	Digital I/O	General purpose I/O	Trace, SPIM4
	TRACEDATA3	Trace data	Trace buffer TRACEDATA[3]	
	SCK	SCK for SPIM4	Dedicated pin for high-speed SPI	

Table 211: WLCSP pin assignments

9.2 Mechanical specifications

The mechanical specifications for the packages show the dimensions in millimeters.

9.2.1 aQFN94 7 x 7 mm package

Dimensions in millimeters for the aQFN94 7 x 7 mm package.

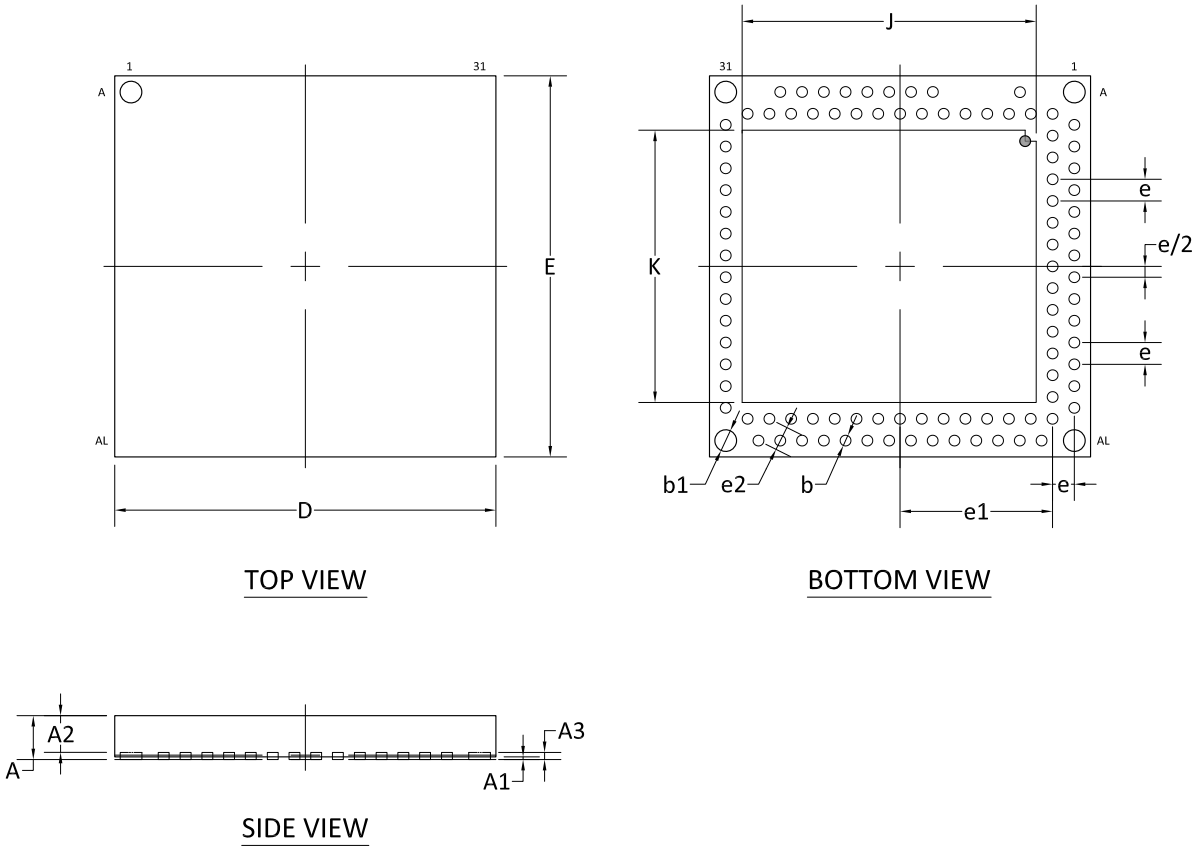


Figure 254: aQFN94 7 x 7 mm package

	A	A1	A2	A3	b	b1	D, E	e	e1	e2	J	K
Min.		0.02			0.15						5.3	4.9
Nom.		0.05	0.675	0.13	0.20	0.4	7.00	0.4	2.8	0.447	5.4	5.0
Max.	0.85	0.08			0.25						5.5	5.1

Table 212: aQFN94 dimensions in millimeters

9.2.2 WLCSP package

Dimensions in millimeters for the WLCSP package.

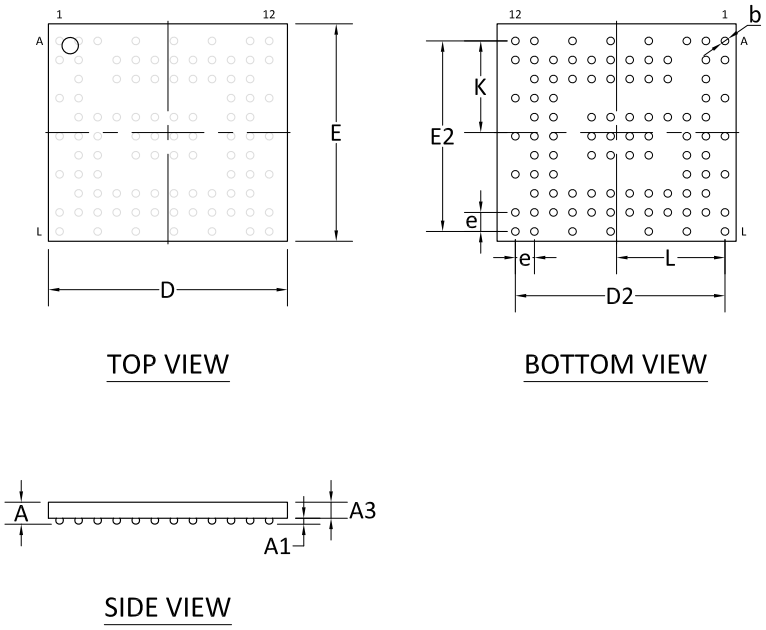


Figure 255: WLCSP 4.4 x 4.0 mm package

	A	A1	A3	b	D	E	D2	E2	e	K	L
Min.	0.361	0.095	0.244	0.12							
Nom.	0.404		0.269		4.390	3.994	3.85	3.5	0.35	1.75	1.925
Max.	0.447	0.125	0.294	0.18							

Table 213: WLCSP dimensions in millimeters

9.3 Reference circuitry

To ensure good RF performance when designing PCBs, it is highly recommended to use the PCB layouts and component values provided by Nordic Semiconductor.

Documentation for the different package reference circuits, including Altium Designer files, PCB layout files, and PCB production files can be downloaded from www.nordicsemi.com.

In this section, there is a reference circuit for QKAA aQFN94, showing the components and component values to support on-chip features in a design.

Some general guidance is summarized here:

- External supply from VDD is only available when power is supplied to VDDH.
- When supplying power from a USB source only, VBUS pin must be connected to VDDH pin if USB is to be used.
- Components required for DC/DC function are only needed if DC/DC mode is enabled for that regulator.
- NFC can be used in any configuration.
- USB can be used in any configuration as long as VBUS is supplied by the USB host.
- The schematics include an optional, but recommended, series resistor on the USB supply for improved immunity to transient over-voltage during VBUS connection.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 1	Battery/Ext. regulator	N/A	Yes	Yes	Yes	Yes	No
Config. 2	N/A	Battery/Ext. regulator	No	No	No	Yes	Yes
Config. 3	N/A	Battery/Ext. regulator	No	No	Yes	No	No
Config. 4	USB (VDDH = VBUS)	N/A	No	No	No	Yes	No

Table 214: Circuit configurations for QKAA aQFN94

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 1	Battery/Ext. regulator	N/A	Yes	Yes	Yes	Yes	No
Config. 2	N/A	Battery/Ext. regulator	No	No	No	Yes	Yes
Config. 3	N/A	Battery/Ext. regulator	No	No	Yes	No	No
Config. 4	USB (VDDH = VBUS)	N/A	No	No	No	Yes	No

Table 215: Circuit configurations for CLAA WLCSP

9.3.1 Circuit configuration no. 1 for QKAA aQFN94

Circuit configuration number 1 for QKAA aQFN94 is showing the schematic and the bill of materials table.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 1	Battery/Ext. regulator	N/A	Yes	Yes	Yes	Yes	No

Table 216: Configuration summary for circuit configuration no. 1

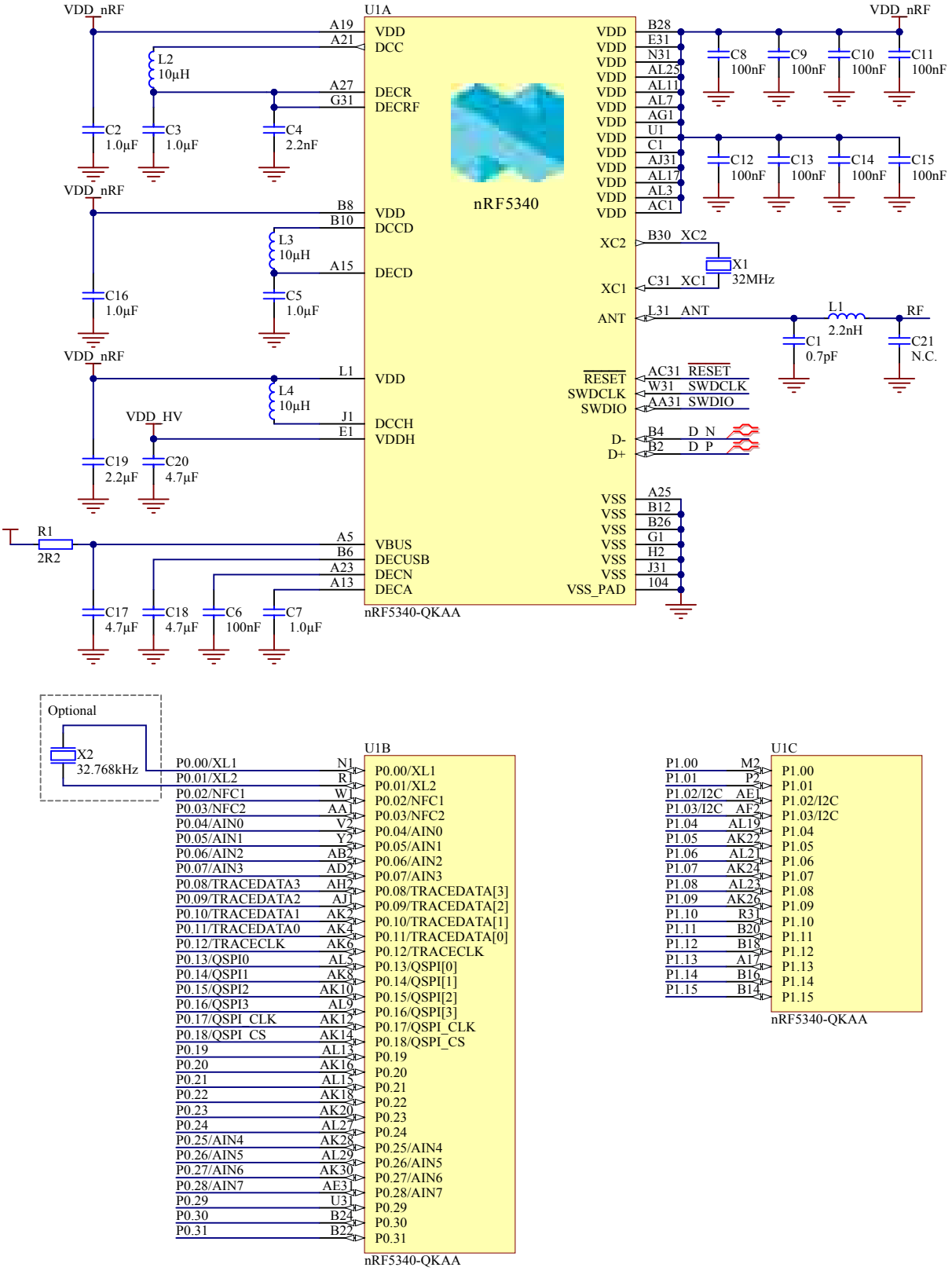


Figure 256: Circuit configuration no. 1 schematic

Note: For PCB reference layouts, see the product page for the nRF5340 on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1	0.7 pF	Capacitor, NP0, ±0.05 pF	0201
C2, C3, C5, C7, C16	1.0 µF	Capacitor, X7S, ±10%	0402
C4	2.2 nF	Capacitor, X7R, ±10%	0201
C6, C8, C9, C10, C11, C12, C13, C14, C15	100 nF	Capacitor, X7S, ±10%	0201
C17, C20	4.7 µF	Capacitor, X7S, ±10%	0603
C18	4.7 µF	Capacitor, X7R, ±10%	0603
C19	2.2 µF	Capacitor, X7R, ±10%	0603
C21	N.C.	Not mounted	0201
L1	2.2 nH	High frequency chip inductor, ±5%	0201
L2, L3	10 µH	Inductor, 50 mA, ±20%	0603
L4	10 µH	Inductor, 80 mA, ±20%	0603
R1	2.2 Ω	Resistor, ±1%, 0.05 W	0201
U1	nRF5340-QKAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, ANT, and 2.4GHz proprietary System on Chip	AQFN-94
X1	32 MHz	Crystal SMD 2016, 32 MHz, Cl=8 pF, Total Tol: ±30 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 98.	XTAL_2016
X2	32.768 kHz	Crystal SMD 2012, 32.768 kHz, Cl=9 pF, Total Tol: ±50 ppm	XTAL_2012

Table 217: Bill of material for circuit configuration no. 1

9.3.2 Circuit configuration no. 2 for QKAA aQFN94

Circuit configuration number 2 for QKAA aQFN94 is showing the schematic and the bill of materials table.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 2	N/A	Battery/Ext. regulator	No	No	No	Yes	Yes

Table 218: Configuration summary for circuit configuration no. 2

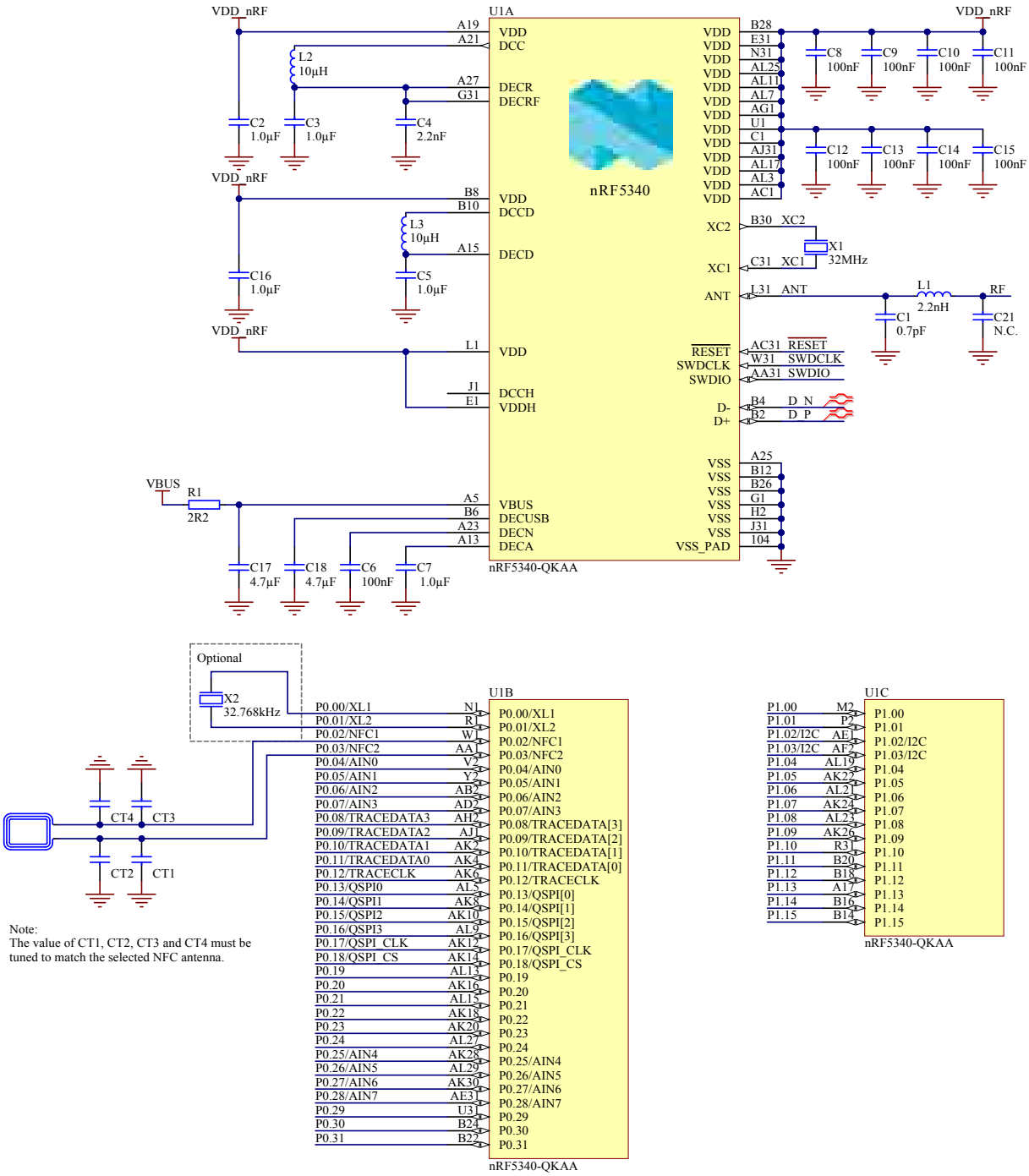


Figure 257: Circuit configuration no. 2 schematic

Note: For PCB reference layouts, see the product page for the nRF5340 on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1	0.7 pF	Capacitor, NP0, ±0.05 pF	0201
C2, C3, C5, C7, C16	1.0 µF	Capacitor, X7S, ±10%	0402
C4	2.2 nF	Capacitor, X7R, ±10%	0201
C6, C8, C9, C10, C11, C12, C13, C14, C15	100 nF	Capacitor, X7S, ±10%	0201
C17	4.7 µF	Capacitor, X7S, ±10%	0603
C18	4.7 µF	Capacitor, X7R, ±10%	0603
C21	N.C.	Not mounted	0201
CT1, CT2, CT3, CT4	Antenna dependent	Capacitor, NP0, ±5%	0201
L1	2.2 nH	High frequency chip inductor, ±5%	0201
L2, L3	10 µH	Inductor, 50 mA, ±20%	0603
R1	2.2 Ω	Resistor, ±1%, 0.05 W	0201
U1	nRF5340-QKAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, ANT, and 2.4GHz proprietary System on Chip	AQFN-94
X1	32 MHz	Crystal SMD 2016, 32 MHz, Cl=8 pF, Total Tol: ±30 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 98.	XTAL_2016
X2	32.768 kHz	Crystal SMD 2012, 32.768 kHz, Cl=9 pF, Total Tol: ±50 ppm	XTAL_2012

Table 219: Bill of material for circuit configuration no. 2

9.3.3 Circuit configuration no. 3 for QKAA aQFN94

Circuit configuration number 3 for QKAA aQFN94 is showing the schematic and the bill of materials table.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 3	N/A	Battery/Ext. regulator	No	No	Yes	No	No

Table 220: Configuration summary for circuit configuration no. 3

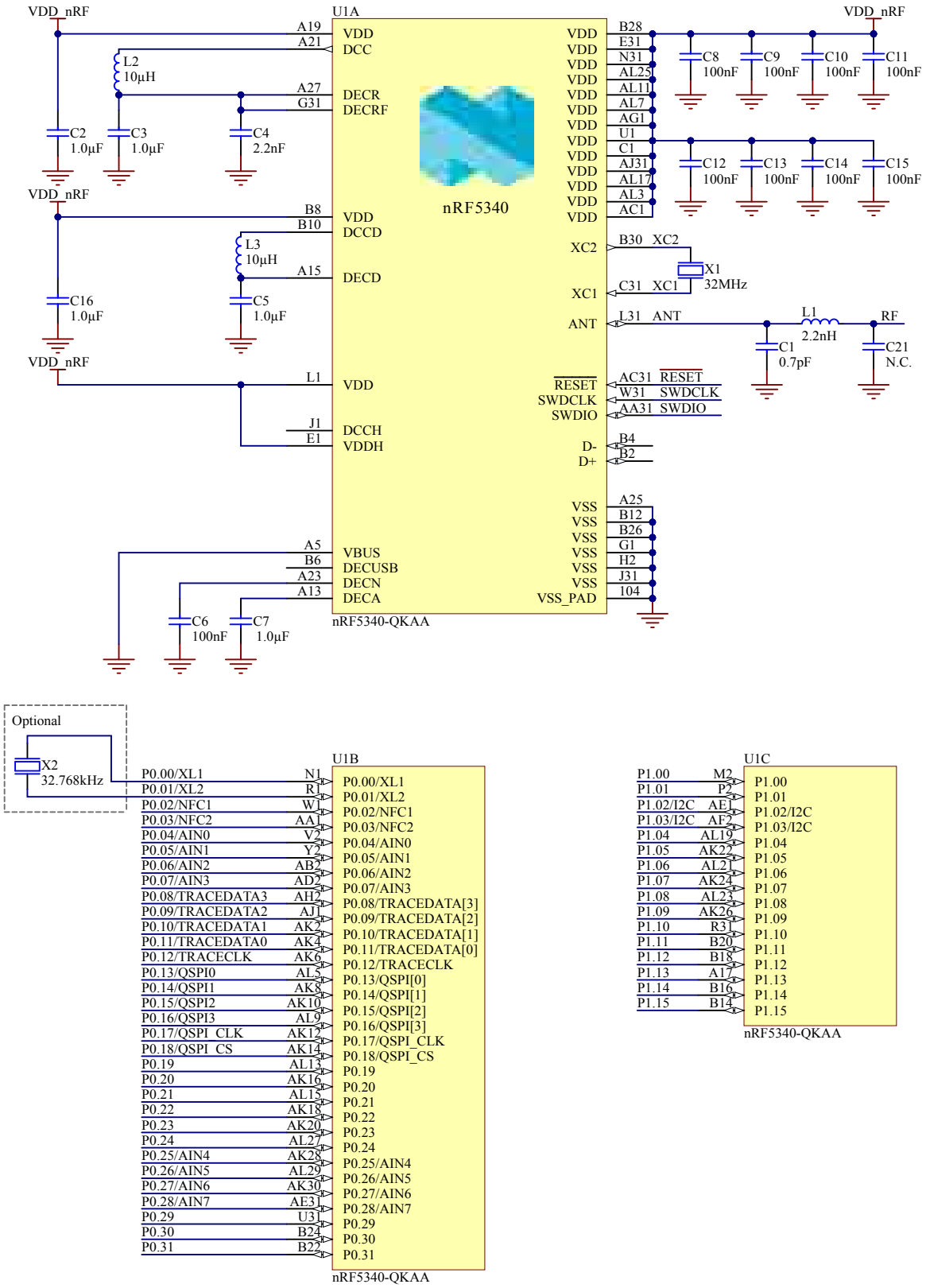


Figure 258: Circuit configuration no. 3 schematic

Note: For PCB reference layouts, see the product page for the nRF5340 on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1	0.7 pF	Capacitor, NP0, ±0.05 pF	0201
C2, C3, C5, C7, C16	1.0 µF	Capacitor, X7S, ±10%	0402
C4	2.2 nF	Capacitor, X7R, ±10%	0201
C6, C8, C9, C10, C11, C12, C13, C14, C15	100 nF	Capacitor, X7S, ±10%	0201
C21	N.C.	Not mounted	0201
L1	2.2 nH	High frequency chip inductor, ±5%	0201
L2, L3	10 µH	Inductor, 50 mA, ±20%	0603
U1	nRF5340-QKAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, ANT, and 2.4GHz proprietary System on Chip	AQFN-94
X1	32 MHz	Crystal SMD 2016, 32 MHz, Cl=8 pF, Total Tol: ±30 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 98.	XTAL_2016
X2	32.768 kHz	Crystal SMD 2012, 32.768 kHz, Cl=9 pF, Total Tol: ±50 ppm	XTAL_2012

Table 221: Bill of material for circuit configuration no. 3

9.3.4 Circuit configuration no. 4 for QKAA aQFN94

Circuit configuration number 4 for QKAA aQFN94 is showing the schematic and the bill of materials table.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 4	USB (VDDH = VBUS)	N/A	No	No	No	Yes	No

Table 222: Configuration summary for circuit configuration no. 4

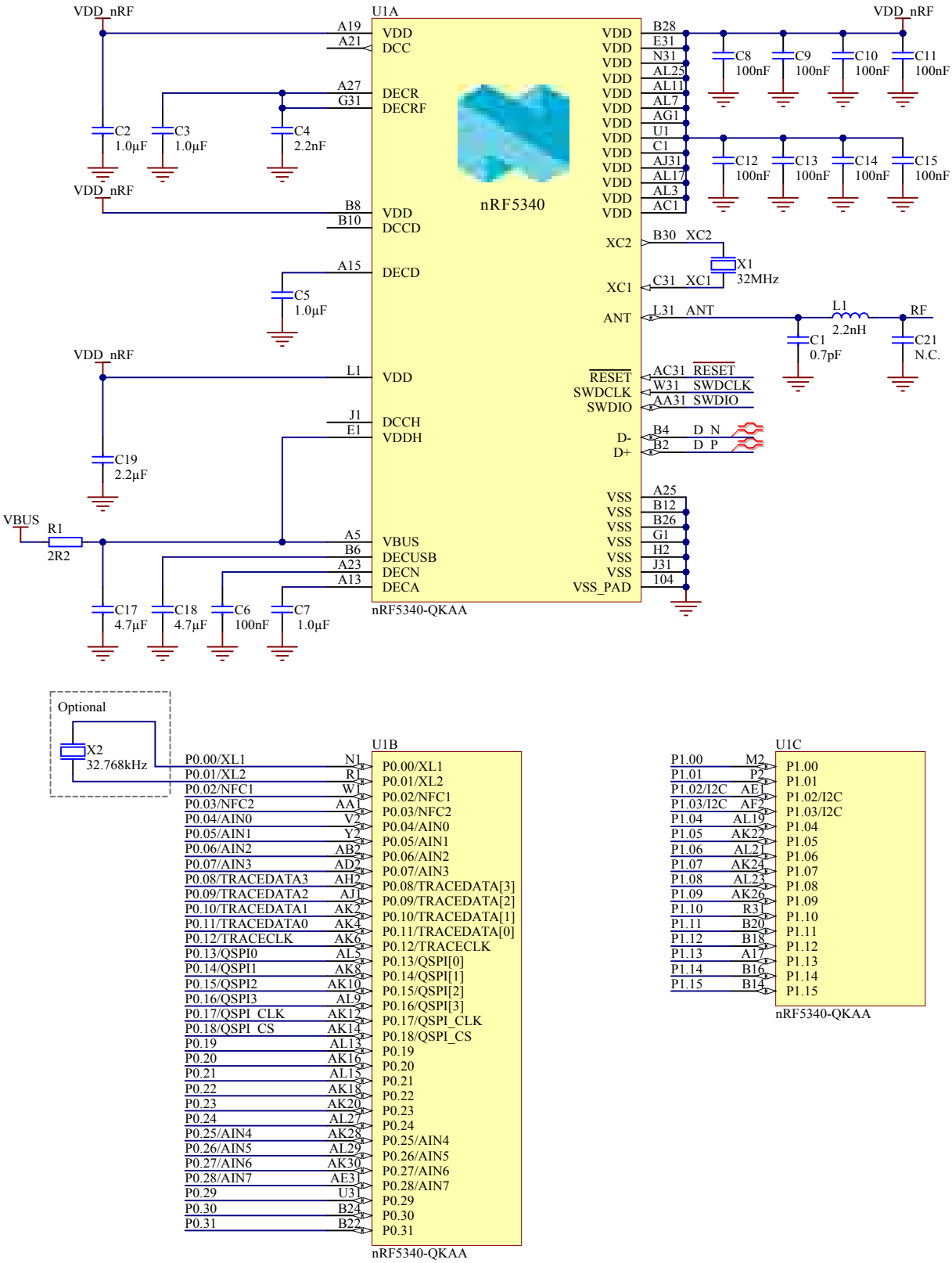


Figure 259: Circuit configuration no. 4 schematic

Note: For PCB reference layouts, see the product page for the nRF5340 on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1	0.7 pF	Capacitor, NP0, ±0.05 pF	0201
C2, C3, C5, C7	1.0 µF	Capacitor, X7S, ±10%	0402
C4	2.2 nF	Capacitor, X7R, ±10%	0201
C6, C8, C9, C10, C11, C12, C13, C14, C15	100 nF	Capacitor, X7S, ±10%	0201
C17	4.7 µF	Capacitor, X7S, ±10%	0603
C18	4.7 µF	Capacitor, X7R, ±10%	0603
C19	2.2 µF	Capacitor, X7R, ±10%	0603
C21	N.C.	Not mounted	0201
L1	2.2 nH	High frequency chip inductor, ±5%	0201
R1	2.2 Ω	Resistor, ±1%, 0.05 W	0201
U1	nRF5340-QKAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, ANT, and 2.4GHz proprietary System on Chip	AQFN-94
X1	32 MHz	Crystal SMD 2016, 32 MHz, Cl=8 pF, Total Tol: ±30 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 98.	XTAL_2016
X2	32.768 kHz	Crystal SMD 2012, 32.768 kHz, Cl=9 pF, Total Tol: ±50 ppm	XTAL_2012

Table 223: Bill of material for circuit configuration no. 4

9.3.5 Circuit configuration no. 1 for CLAA WLCSP

Circuit configuration number 1 for CLAA WLCSP is showing the schematic and the bill of materials table.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 1	Battery/Ext. regulator	N/A	Yes	Yes	Yes	Yes	No

Table 224: Configuration summary for circuit configuration no. 1

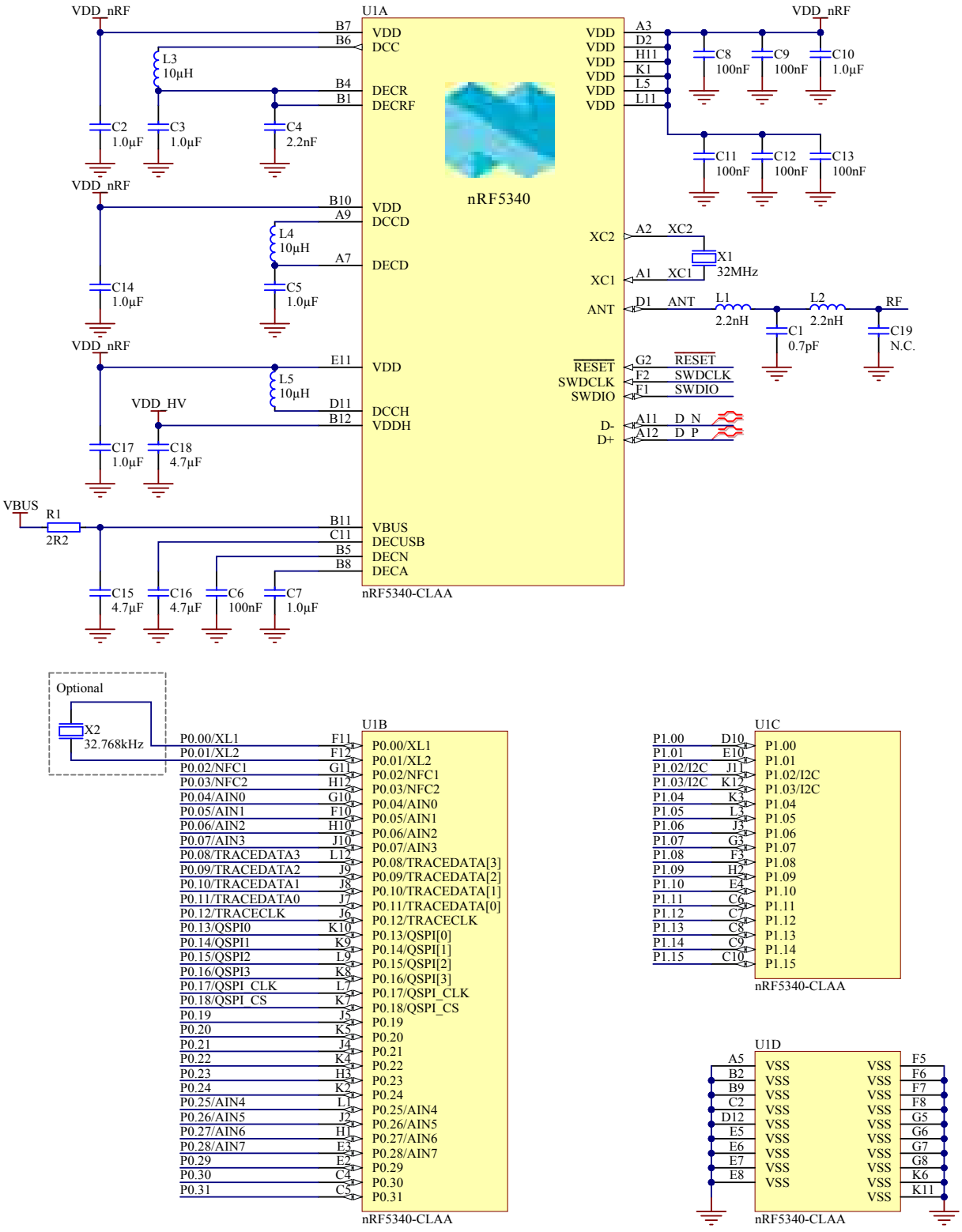


Figure 260: Circuit configuration no. 1 schematic

Note: For PCB reference layouts, see the product page for the nRF5340 on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1	0.7 pF	Capacitor, NP0, ±0.05 pF	0201
C2, C3, C5, C7, C10, C14, C17	1.0 µF	Capacitor, X7S, ±10%	0402
C4	2.2 nF	Capacitor, X7R, ±10%	0201
C6, C8, C9, C11, C12, C13	100 nF	Capacitor, X7S, ±10%	0201
C15, C18	4.7 µF	Capacitor, X7S, ±10%	0603
C16	4.7 µF	Capacitor, X7R, ±10%	0603
C19	N.C.	Not mounted	0201
L1, L2	2.2 nH	High frequency chip inductor, ±5%	0201
L3, L4	10 µH	Inductor, 50 mA, ±20%	0603
L5	10 µH	Inductor, 80 mA, ±20%	0603
R1	2.2 Ω	Resistor, ±1%, 0.05 W	0201
U1	nRF5340-CLAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, ANT, and 2.4GHz proprietary System on Chip	WLCSP-95
X1	32 MHz	Crystal SMD 2016, 32 MHz, Cl=8 pF, Total Tol: ±30 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 98.	XTAL_2016
X2	32.768 kHz	Crystal SMD 2012, 32.768 kHz, Cl=9 pF, Total Tol: ±50 ppm	XTAL_2012

Table 225: Bill of material for circuit configuration no. 1

9.3.6 Circuit configuration no. 2 for CLAA WLCSP

Circuit configuration number 2 for CLAA WLCSP is showing the schematic and the bill of materials table.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 2	N/A	Battery/Ext. regulator	No	No	No	Yes	Yes

Table 226: Configuration summary for circuit configuration no. 2

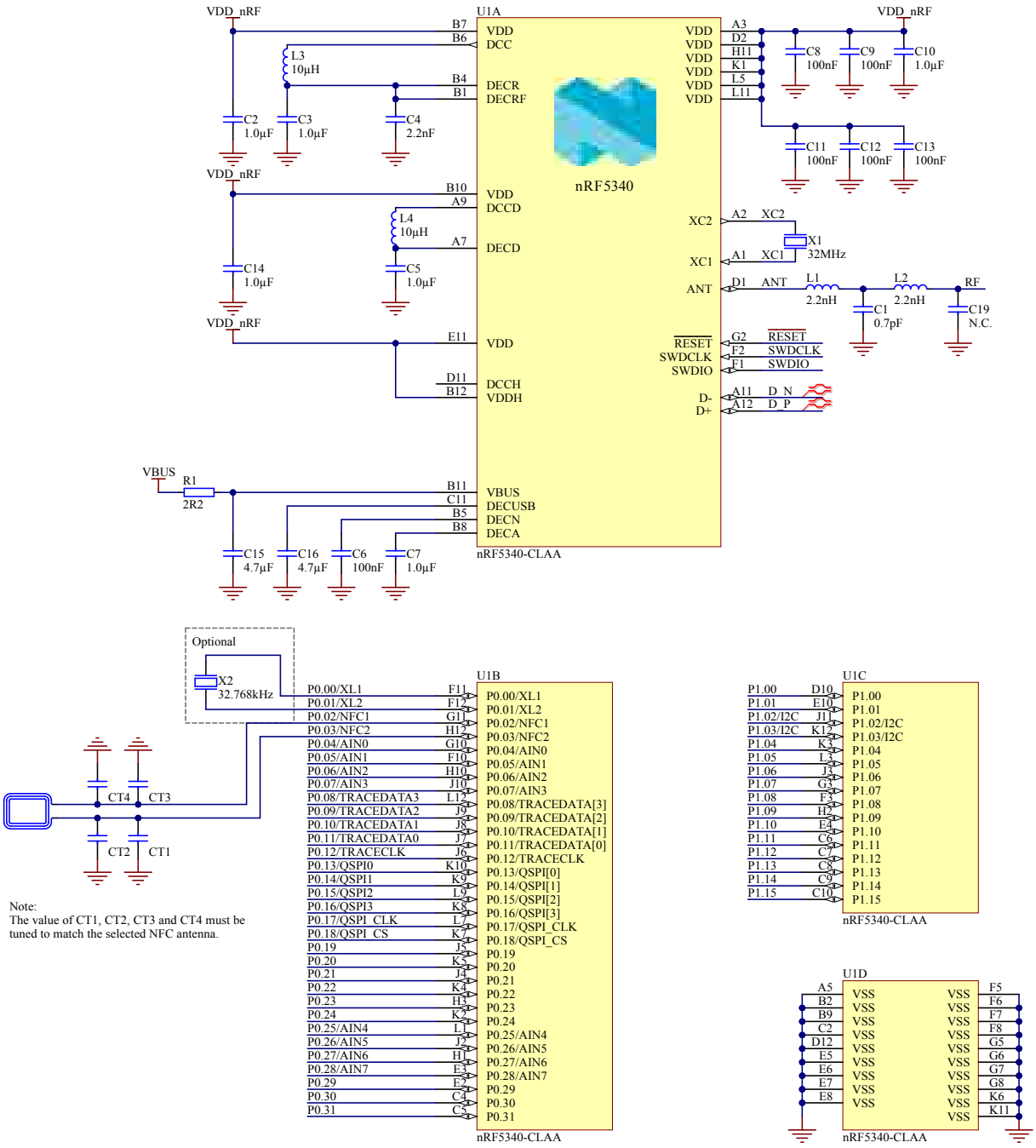


Figure 261: Circuit configuration no. 2 schematic

Note: For PCB reference layouts, see the product page for the nRF5340 on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1	0.7 pF	Capacitor, NP0, ±0.05 pF	0201
C2, C3, C5, C7, C10, C14	1.0 µF	Capacitor, X7S, ±10%	0402
C4	2.2 nF	Capacitor, X7R, ±10%	0201
C6, C8, C9, C11, C12, C13	100 nF	Capacitor, X7S, ±10%	0201
C15	4.7 µF	Capacitor, X7S, ±10%	0603
C16	4.7 µF	Capacitor, X7R, ±10%	0603
C19	N.C.	Not mounted	0201
CT1, CT2, CT3, CT4	Antenna dependent	Capacitor, NP0, ±5%	0201
L1, L2	2.2 nH	High frequency chip inductor, ±5%	0201
L3, L4	10 µH	Inductor, 50 mA, ±20%	0603
R1	2.2 Ω	Resistor, ±1%, 0.05 W	0201
U1	nRF5340-CLAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, ANT, and 2.4GHz proprietary System on Chip	WLCSP-95
X1	32 MHz	Crystal SMD 2016, 32 MHz, Cl=8 pF, Total Tol: ±30 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 98.	XTAL_2016
X2	32.768 kHz	Crystal SMD 2012, 32.768 kHz, Cl=9 pF, Total Tol: ±50 ppm	XTAL_2012

Table 227: Bill of material for circuit configuration no. 2

9.3.7 Circuit configuration no. 3 for CLAA WLCSP

Circuit configuration number 3 for CLAA WLCSP is showing the schematic and the bill of materials table.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 3	N/A	Battery/Ext. regulator	No	No	Yes	No	No

Table 228: Configuration summary for circuit configuration no. 3

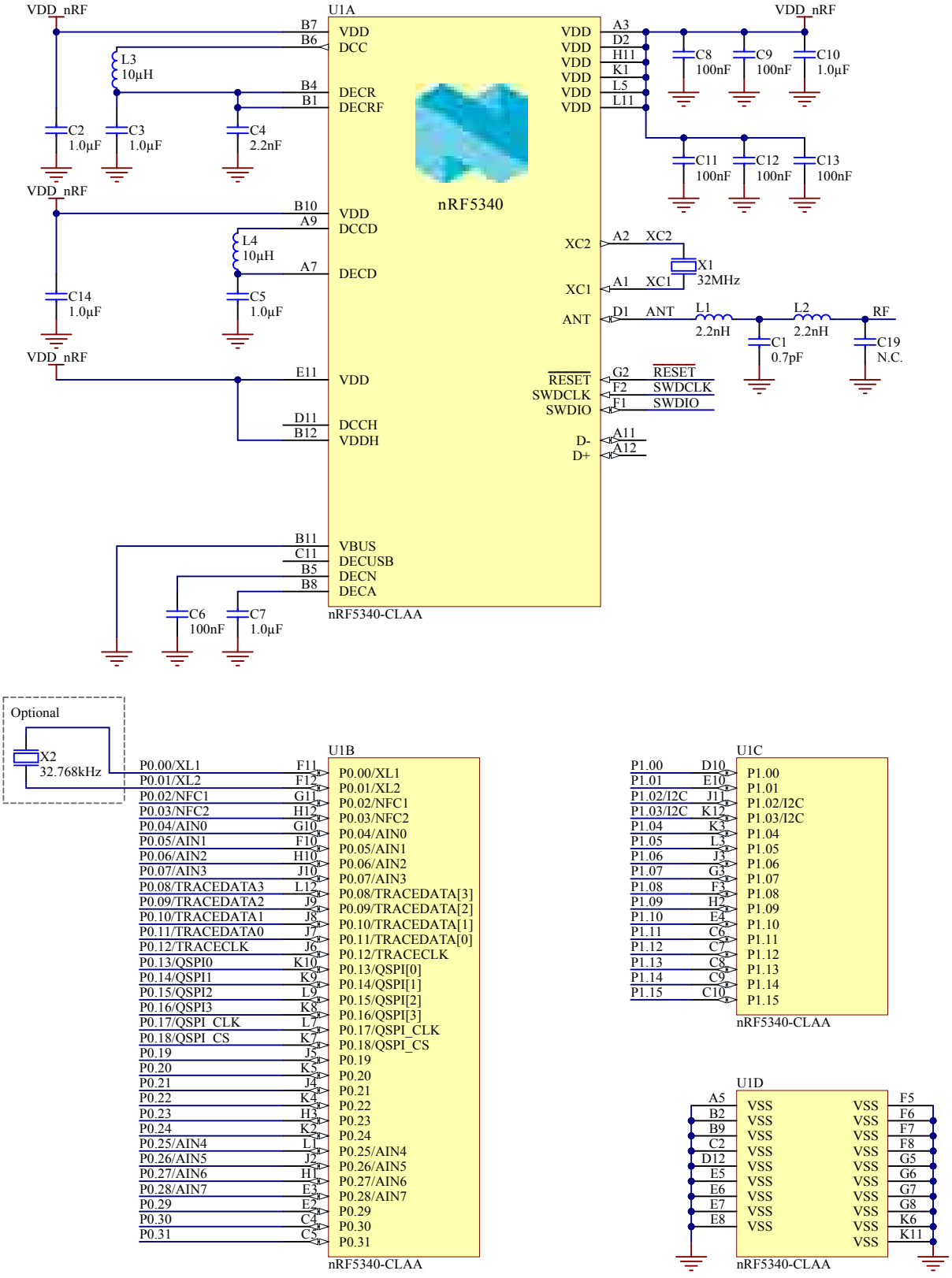


Figure 262: Circuit configuration no. 3 schematic

Note: For PCB reference layouts, see the product page for the nRF5340 on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1	0.7 pF	Capacitor, NP0, ±0.05 pF	0201
C2, C3, C5, C7, C10, C14	1.0 µF	Capacitor, X7S, ±10%	0402
C4	2.2 nF	Capacitor, X7R, ±10%	0201
C6, C8, C9, C11, C12, C13	100 nF	Capacitor, X7S, ±10%	0201
C19	N.C.	Not mounted	0201
L1, L2	2.2 nH	High frequency chip inductor, ±5%	0201
L3, L4	10 µH	Inductor, 50 mA, ±20%	0603
U1	nRF5340-CLAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, ANT, and 2.4GHz proprietary System on Chip	WLCSP-95
X1	32 MHz	Crystal SMD 2016, 32 MHz, Cl=8 pF, Total Tol: ±30 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 98.	XTAL_2016
X2	32.768 kHz	Crystal SMD 2012, 32.768 kHz, Cl=9 pF, Total Tol: ±50 ppm	XTAL_2012

Table 229: Bill of material for circuit configuration no. 3

9.3.8 Circuit configuration no. 4 for CLAA WLCSP

Circuit configuration number 4 for CLAA WLCSP is showing the schematic and the bill of materials table.

Config no.	Supply configuration		Enabled features				
	VDDH	VDD	EXTSUPPLY	DCDC on VREGH	DCDC on VREGMAIN and VREGRADIO	USB	NFC
Config. 4	USB (VDDH = VBUS)	N/A	No	No	No	Yes	No

Table 230: Configuration summary for circuit configuration no. 4

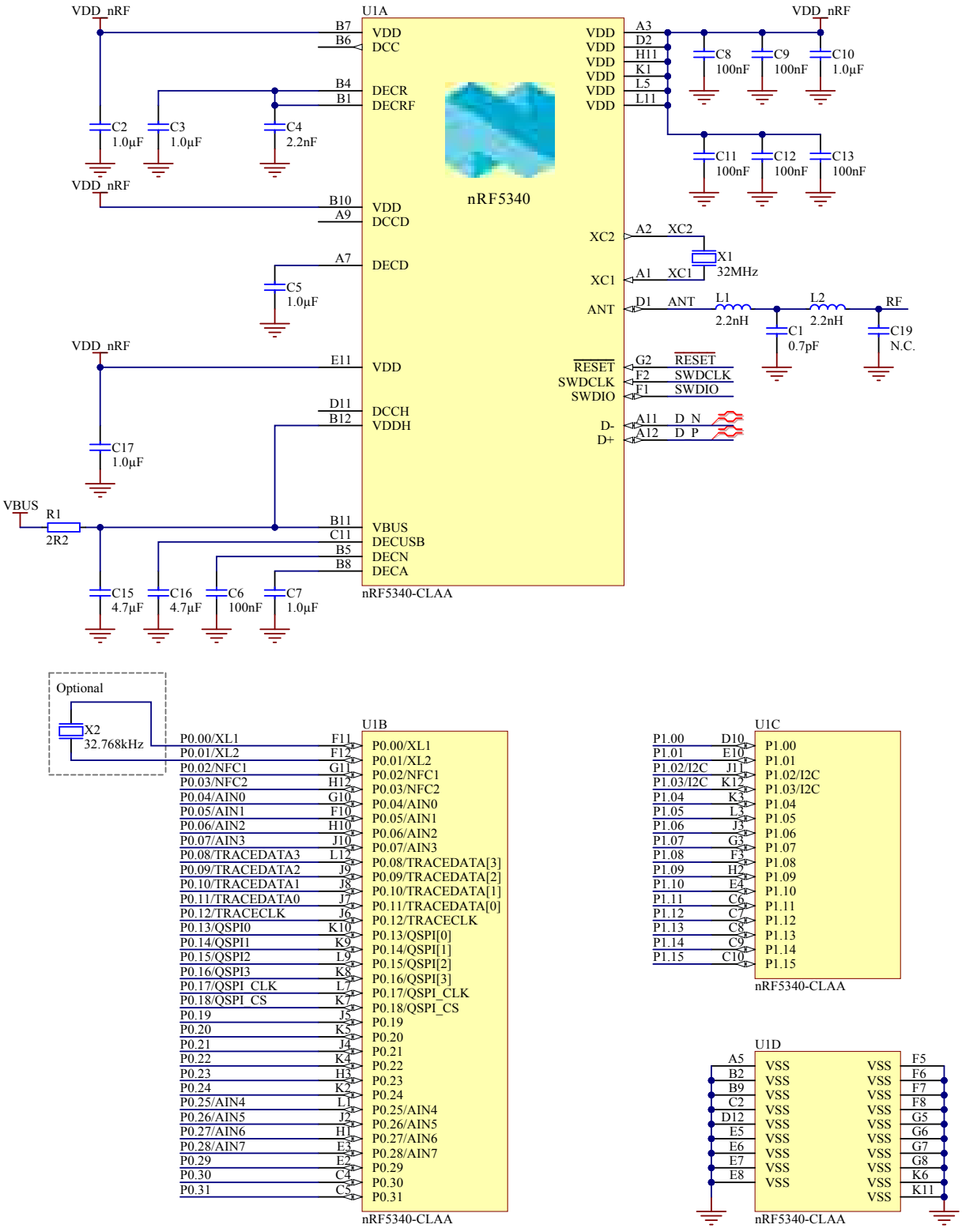


Figure 263: Circuit configuration no. 4 schematic

Note: For PCB reference layouts, see the product page for the nRF5340 on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1	0.7 pF	Capacitor, NP0, ±0.05 pF	0201
C2, C3, C5, C7, C10, C17	1.0 µF	Capacitor, X7S, ±10%	0402
C4	2.2 nF	Capacitor, X7R, ±10%	0201
C6, C8, C9, C11, C12, C13	100 nF	Capacitor, X7S, ±10%	0201
C15	4.7 µF	Capacitor, X7S, ±10%	0603
C16	4.7 µF	Capacitor, X7R, ±10%	0603
C19	N.C.	Not mounted	0201
L1, L2	2.2 nH	High frequency chip inductor, ±5%	0201
R1	2.2 Ω	Resistor, ±1%, 0.05 W	0201
U1	nRF5340-CLAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, ANT, and 2.4GHz proprietary System on Chip	WLCSP-95
X1	32 MHz	Crystal SMD 2016, 32 MHz, Cl=8 pF, Total Tol: ±30 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 98.	XTAL_2016
X2	32.768 kHz	Crystal SMD 2012, 32.768 kHz, Cl=9 pF, Total Tol: ±50 ppm	XTAL_2012

Table 231: Bill of material for circuit configuration no. 4

9.3.9 PCB layout example

The PCB layout in the following figure is a reference layout for Circuit configuration no. 1 for QKAA aQFN94.

Note: Pay attention to how the capacitor C1 is grounded. It is not directly connected to the ground plane, but grounded via pin J31 and to the VSS die pad. This is done to create additional filtering of harmonic components.

For all available reference layouts, see the product page for nRF5340 on www.nordicsemi.com.

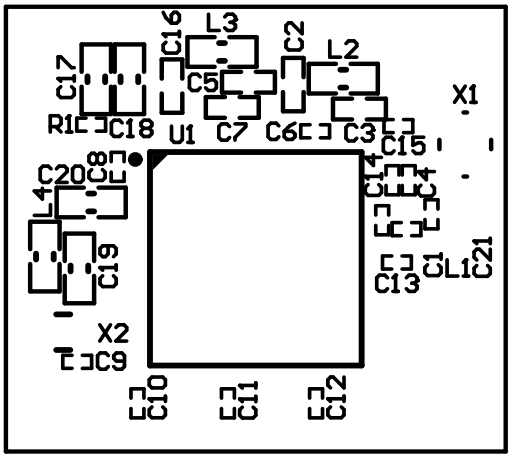


Figure 264: Top silk layer

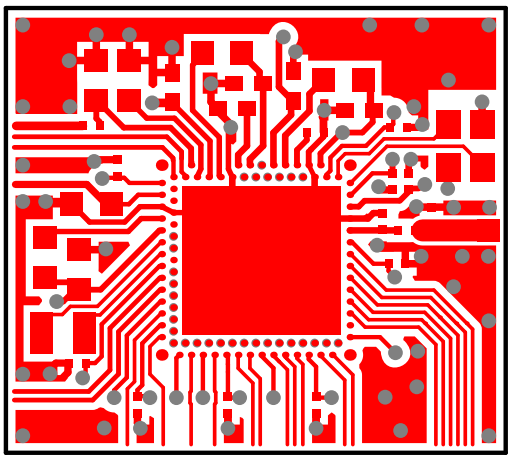


Figure 265: Top layer

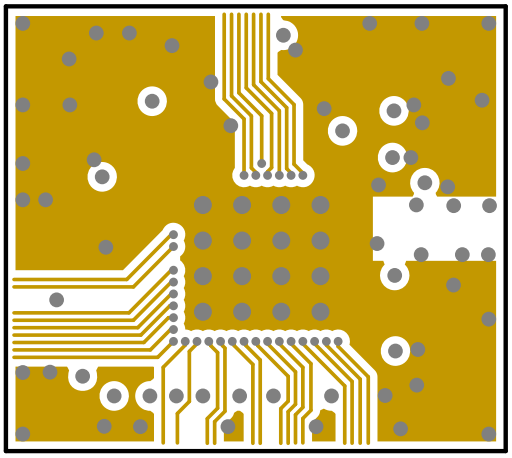


Figure 266: Mid layer 1

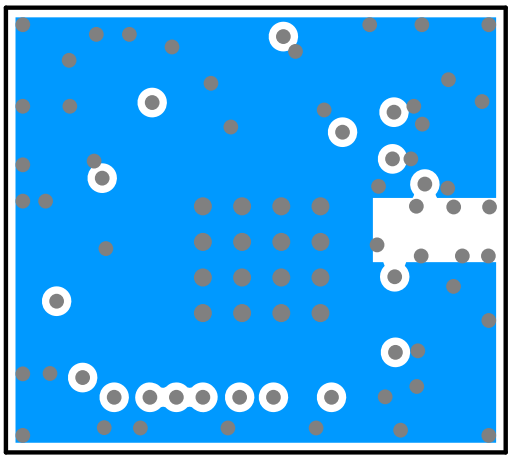


Figure 267: Mid layer 2

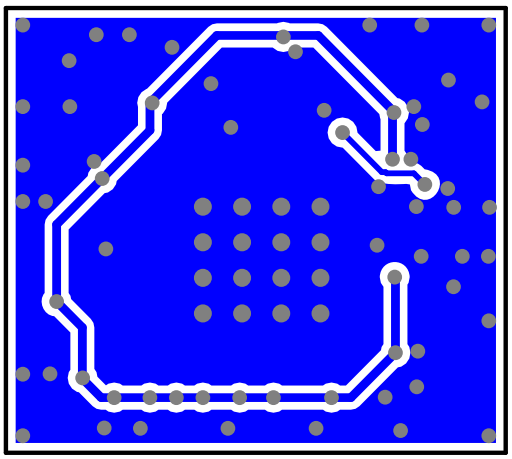


Figure 268: Bottom layer

10 Recommended operating conditions

The operating conditions are the physical parameters that the chip can operate within.

Symbol	Parameter	Min.	Nom.	Max.	Units
VDD	VDD supply voltage, independent of DCDC enable	1.7	3.0	3.6	V
VDDH	VDDH supply voltage, independent of DCDC enable	2.5	3.7	5.5	V
VBUS	VBUS USB supply voltage	4.35	5.0	5.5	V
TA	Operating temperature	-40	25	105	°C

Table 232: Recommended operating conditions

10.1 WLCSP light sensitivity

All WLCSP package variants are sensitive to visible and close-range infrared light. This means that a final product design must shield the chip properly, either by final product encapsulation or by shielding/coating of the WLCSP device.

Some WLCSP package variants have a backside coating, where the marking side of the device is covered with a light absorbing film, while the side edges and the ball side of the device are still exposed and need to be protected. Other WLCSP package variants do not have any such protection.

The WLCSP package variant CLAA has a backside coating.

11 Absolute maximum ratings

Maximum ratings are the extreme limits to which the chip can be exposed for a limited amount of time without permanently damaging it. Exposure to absolute maximum ratings for prolonged periods of time may affect the reliability of the device[33].

	Min.	Max.	Unit
Supply voltages			
VDD	-0.3	+3.9	V
VDDH	-0.3	+5.8	V
VBUS	-0.3	+5.8	V
VSS		0	V
I/O pin voltage			
$V_{I/O}$, VDD ≤ 3.6 V	-0.3	VDD + 0.3	V
$V_{I/O}$, VDD > 3.6 V	-0.3	3.9	V
Environmental aQFN package			
Storage temperature	-40	+125	°C
Moisture Sensitivity Level (MSL)		2	
ESD Human Body Model (HBM)		2 (all pins except DECR and DECN, rated at 1.4 kV)	kV
ESD Charged Device Model (CDM)		500	V
Flash memory			
Endurance	10 000 write/erase cycles		
Retention	10 years at 40°C		

Table 233: Absolute maximum ratings

[33] For accellerated life time testing (HTOL, etc) supply voltage should not exceed the recommended operating conditions max value, see Recommended operating conditions on page 812.

12 Ordering information

This chapter contains information on device marking, ordering codes, and container sizes.

12.1 Device marking

The nRF5340 package is marked as shown in the following figure.

N	5	3	4	0	
<P	P>	<V	V>	<H>	<P>
<Y	Y>	<W	W>	<L	L>

Figure 269: Device marking

12.2 Box labels

The following figures show the box labels used for nRF5340.

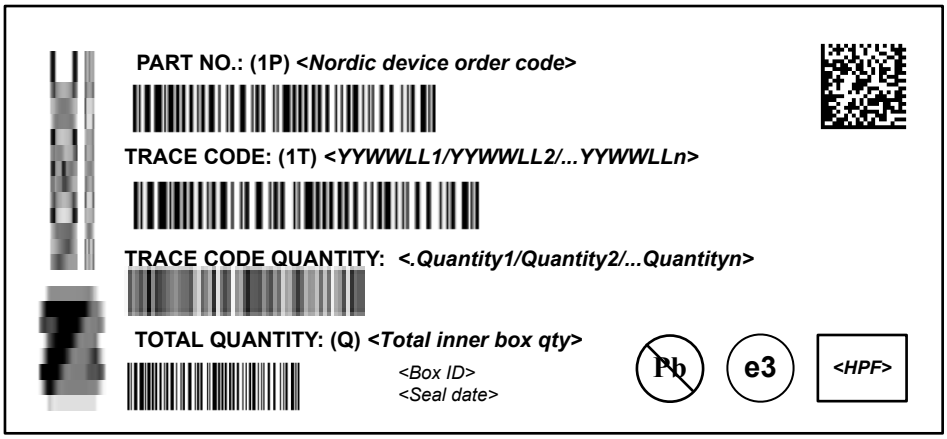


Figure 270: Inner box label

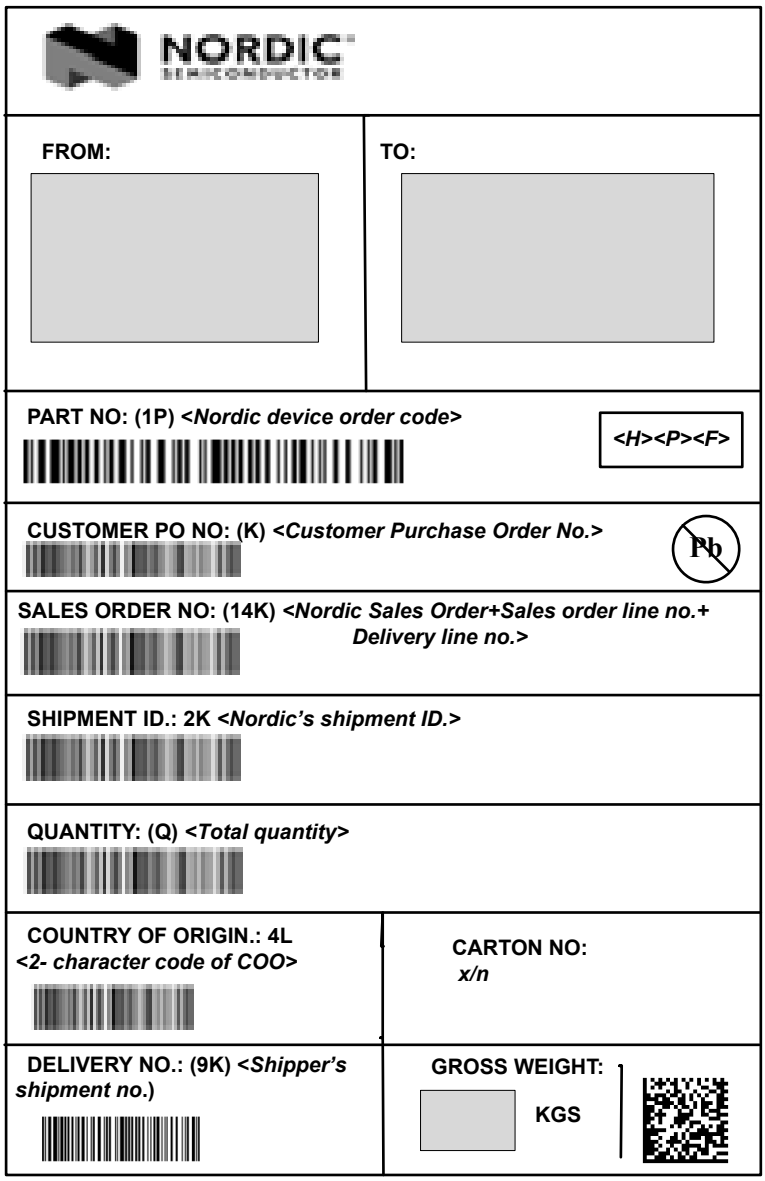


Figure 271: Outer box label

12.3 Order code

The following are the order codes and definitions for nRF5340.

n	R	F	5	3	4	0	-	<P	P>	<V	V>	-	<C	C>

Figure 272: Order code

Abbreviation	Definition and implemented codes
N53/nRF53	nRF53 series product
40	Part code
<PP>	Package variant code
<VV>	Function variant code
<H><P><F>	Build code H - Hardware version code P - Production configuration code (production site, etc.) F - Firmware version code (only visible on shipping container label)
<YY><WW><LL>	Tracking code YY - Year code WW - Assembly week number LL - Wafer lot code
<CC>	Container code

Table 234: Abbreviations

12.4 Code ranges and values

Defined here are nRF5340 code ranges and values.

<PP>	Package	Size (mm)	Pin/Ball count	Pitch (mm)
QK	AQFN	7 x 7	94	0.4
CL	WLCSP	4.390 x 3.994	95	0.35

Table 235: Package variant codes

<VV>	Flash (kB)	RAM (kB)
AA	1024	512

Table 236: Function variant codes

<H>	Description
[A . . Z]	Hardware version/revision identifier (incremental)

Table 237: Hardware version codes

<P>	Description
[0 . . 9]	Production device identifier (incremental)
[A . . Z]	Engineering device identifier (incremental)

Table 238: Production configuration codes

<F>	Description
[A . . N, P . . Z]	Version of preprogrammed firmware
[0]	Delivered without preprogrammed firmware

Table 239: Production version codes

<YY>	Description
[16 . . 99]	Production year: 2016 to 2099

Table 240: Year codes

<WW>	Description
[1 . . 52]	Week of production

Table 241: Week codes

<LL>	Description
[AA . . ZZ]	Wafer production lot identifier

Table 242: Lot codes

<CC>	Description
R7	7" Reel
R	13" Reel

Table 243: Container codes

12.5 Product options

Defined here are the nRF5340 product options.

The following table lists the ordering code, as well as the minimum ordering quantity (MOQ).

Order code	MOQ
nRF5340-QKAA-R7	800
nRF5340-QKAA-R	3000
nRF5340-CLAA-R7	1500
nRF5340-CLAA-R	7000

Table 244: nRF5340 order codes

Order code	Description
nRF5340-DK	nRF5340 Development Kit

Table 245: Development tools order code

13 Legal notices

By using this documentation you agree to our terms and conditions of use. Nordic Semiconductor may change these terms and conditions at any time without notice.

Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function, or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

Nordic Semiconductor ASA does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. If there are any discrepancies, ambiguities or conflicts in Nordic Semiconductor's documentation, the Product Specification prevails.

Nordic Semiconductor ASA reserves the right to make corrections, enhancements, and other changes to this document without notice.

Life support applications

Nordic Semiconductor ASA's products are not designed for use in life-critical medical equipment, support appliances, devices, or systems where malfunction of Nordic Semiconductor ASA's products can reasonably be expected to result in personal injury. Customer may not use any Nordic Semiconductor ASA's products in life-critical medical equipment unless adequate design and operating safeguards by customer's authorized officers have been made. Customer represents that, with respect to its applications, it has all the necessary expertise to create and implement safeguards that anticipate dangerous consequences of failures, monitor failures and their consequences, and lessen the likelihood of failures that might cause harm, and to take appropriate remedial actions. Customer agrees that prior to using or distributing any life-critical medical equipment that include Nordic Semiconductor ASA's products, customer will thoroughly test such systems and the functionality of such products as used in such systems.

Nordic Semiconductor ASA assumes no liability for applications assistance or the design of customers' products. Customers are solely responsible for the design, validation, and testing of its applications as well as for compliance with all legal, regulatory, and safety-related requirements concerning its applications.

Customer will fully indemnify Nordic Semiconductor ASA and its representatives against any damages, costs, losses, and/or liabilities arising out of customer's non-compliance with this section.

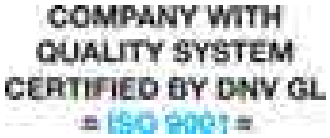
RoHS and REACH statement

Refer to www.nordicsemi.com for complete hazardous substance reports, material composition reports, and latest version of Nordic's RoHS and REACH statements.

Trademarks

All trademarks, service marks, trade names, product names, and logos appearing in this documentation are the property of their respective owners.

Copyright notice


COMPANY WITH
QUALITY SYSTEM
CERTIFIED BY DNV GL
= ISO 9001 =