

2.4GHz wireless audio streamer

nRF24Z1

FEATURES

- Low cost 0.18u CMOS process, 36 pin 6x6mm QFN package
- Single chip 2.4GHz RF transceiver
- 4Mbit/sec RF link
- Input sample rate up to 96kHz, 24 bits
- Output sample rate up to 48kHz, 24 bits
- Programmable latency
- Quality of Service engine supporting up to 1.536 Mbit/s LPCM audio
- S/PDIF interface for direct connection to PC soundcard and surround receivers
- I2S interface for glue-less audio support
- SPI or 2-wire interface for up to 12 kbit/sec peak bi-directional digital control/AUX data
- On chip optional compression
- On chip voltage regulators
- Few external components
- Uses global 2.4GHz band

APPLICATIONS

- Compact Disk, CD quality headset
- MP3 / Mini Disk headset
- Speakers
- Surround speakers
- Microphone
- Musical instruments
- Audio streaming from PC soundcard to HiFi system
- Download MP3 files from PC to MP3 player
- Compressed video streaming

GENERAL DESCRIPTION

nRF24Z1 provides a true single chip system for CD quality audio streaming of up to 16 bit 48 kHz audio with support of up to 24 bit 96 kHz input. I2S and S/PDIF interfaces are supported for audio I/O. Seamless interfacing of low cost A/D and D/A for analog audio input and output. SPI or 2-wire (I2C compatible) serial interfaces for control. The circuit has embedded voltage regulators, giving maximum noise immunity and operation from a single 2.0V to 3.6V supply.

QUICK REFERENCE DATA

| Parameter | Value | Unit |
|--|------------|------|
| Minimum supply voltage | 2.0 | V |
| Temperature range | -20 to +80 | °C |
| Peak supply current in transmit @ -5dBm output power | 15 | mA |
| Peak supply current in receive mode | 32 | mA |
| Supply current in power down mode | 4 | µA |
| Maximum transmit output power | 0 | dBm |
| Audio sample rate | 8 to 96 | kS/s |
| Audio resolution | 16 | bit |
| Receiver sensitivity | -80 | dBm |

Table 1-1 nRF24Z1 quick reference data.



ORDERING INFORMATION

| Type number | Description | Version |
|---------------|----------------|---------|
| nRF24Z1 | 36L QFN 6x6 mm | B |
| nRF24Z1-EVKIT | Evaluation kit | 1.0 |

Table 1-2 nRF24Z1 ordering information.



TABLE OF CONTENTS

| | |
|---|----|
| FEATURES | 1 |
| APPLICATIONS | 1 |
| GENERAL DESCRIPTION | 1 |
| QUICK REFERENCE DATA | 1 |
| ORDERING INFORMATION | 2 |
| TABLE OF CONTENTS | 3 |
| 1 Pin Assignment | 6 |
| 2 Pin Function | 7 |
| 3 Glossary of terms | 8 |
| 4 Architectural Overview | 9 |
| 4.1 Audio transmitter | 11 |
| 4.1.1 I2S audio input | 11 |
| 4.1.2 S/PDIF audio input | 12 |
| 4.1.3 Serial control (slave) interfaces | 12 |
| 4.1.4 Master interfaces | 12 |
| 4.1.5 Direct data input pins | 13 |
| 4.1.6 Interrupt output | 13 |
| 4.2 Audio Receiver | 13 |
| 4.2.1 I2S audio output | 14 |
| 4.2.2 S/PDIF audio output | 15 |
| 4.2.3 Master interfaces | 15 |
| 4.2.4 Serial control (slave) interfaces | 15 |
| 4.2.5 Parallel port and PWM | 15 |
| 4.3 Blocks common to audio transmitter and receiver | 16 |
| 4.3.1 XTAL Oscillator | 16 |
| 4.3.2 Radio Transceiver | 16 |
| 4.3.3 Quality of Service (QoS) engine | 16 |
| 4.3.4 Audio compression / decompression | 17 |
| 4.3.5 Power | 17 |
| 4.3.6 IREF / RESET | 17 |
| 5 Operation overview | 18 |
| 5.1 Power on / RESET sequence | 18 |
| 5.2 RF Link initialization | 18 |
| 5.2.1 Idle state | 18 |
| 5.2.2 Link-locate state | 19 |
| 5.2.2.1 Link-locate on ATX | 19 |
| 5.2.3 Synchronization state | 19 |
| 5.3 Audio streaming | 19 |
| 5.4 Audio receiver clock rate recovery | 20 |
| 5.5 Data link | 20 |
| 5.6 Power down mode | 21 |
| 6 nRF24Z1 register MAP | 22 |
| 6.1 Access from audio transmitter side | 22 |
| 6.2 Access from audio receiver side | 22 |
| 6.3 Register map in external EEPROM | 24 |



| | | |
|--------|---|----|
| 7 | Digital I/O | 25 |
| 7.1 | Digital I/O behaviour during RESET | 25 |
| 7.2 | Audio interfaces | 25 |
| 7.3 | I2S Audio Interface | 25 |
| 7.4 | S/PDIF Audio Interface | 26 |
| 7.5 | Audio interface functionality | 27 |
| 7.5.1 | ATX audio interface control | 28 |
| 7.5.2 | ARX audio interface control | 31 |
| 7.5.3 | I2S Audio interface timing | 33 |
| 7.6 | Serial master interfaces | 33 |
| 7.6.1 | Timing serial master interfaces | 37 |
| 7.7 | Control and GPIO interfaces | 38 |
| 7.7.1 | ATX control and GPIO pins | 38 |
| 7.7.2 | SPI slave interface | 39 |
| 7.7.3 | 2-wire slave interface | 39 |
| 7.7.4 | General purpose input pins D[2:0] | 41 |
| 7.7.5 | ATX Control interface timing | 41 |
| 7.7.6 | ARX control interface options | 42 |
| 7.7.7 | ARX GPIO pins | 44 |
| 7.8 | Data Channel Timing | 45 |
| 7.8.1 | Forward data channel, data transfer from ATX to ARX | 45 |
| 7.8.2 | Return data channel, data transfer from ARX to ATX | 46 |
| 8 | Quality of service (Qos) and RF | 48 |
| 8.1 | Link algorithm | 48 |
| 8.2 | RF protocol | 48 |
| 8.3 | Adaptive frequency hopping | 48 |
| 8.3.1 | Adapting to the RF environment | 50 |
| 8.4 | Link registers | 51 |
| 8.4.1 | RF link latency | 52 |
| 8.5 | RF output power | 53 |
| 9 | Interrupts | 54 |
| 10 | RESET outputs | 56 |
| 11 | Power down control | 57 |
| 11.1 | Enter power down | 57 |
| 11.2 | Wake up from power down | 57 |
| 11.2.1 | External pin change | 57 |
| 11.2.2 | Timer controlled wake up | 58 |
| 11.3 | nRF24Z1 power saving example | 60 |
| 12 | Register update over the data link | 61 |
| 12.1 | Register update and relink | 62 |
| 13 | Test mode | 64 |
| 14 | Electrical Specification | 65 |
| 15 | Absolute maximum ratings | 67 |
| 16 | Package outline | 68 |
| 17 | Application Information | 69 |
| 17.1 | Antenna I/O | 69 |
| 17.2 | Crystal Specification | 69 |
| 17.3 | Bias reference resistor | 69 |



nRF24Z1 wireless audio streamer

| | | |
|------|--|----|
| 17.4 | Internal digital supply de-coupling | 69 |
| 17.5 | PCB layout and de-coupling guidelines..... | 70 |
| 18 | Application example | 71 |
| 18.1 | nRF24Z1 schematics..... | 71 |
| 18.2 | nRF24Z1 layout | 73 |
| 18.3 | nRF24Z1 Bill of Material | 75 |
| 19 | References..... | 77 |
| 20 | Definitions..... | 78 |
| 21 | Your notes..... | 79 |



1 PIN ASSIGNMENT

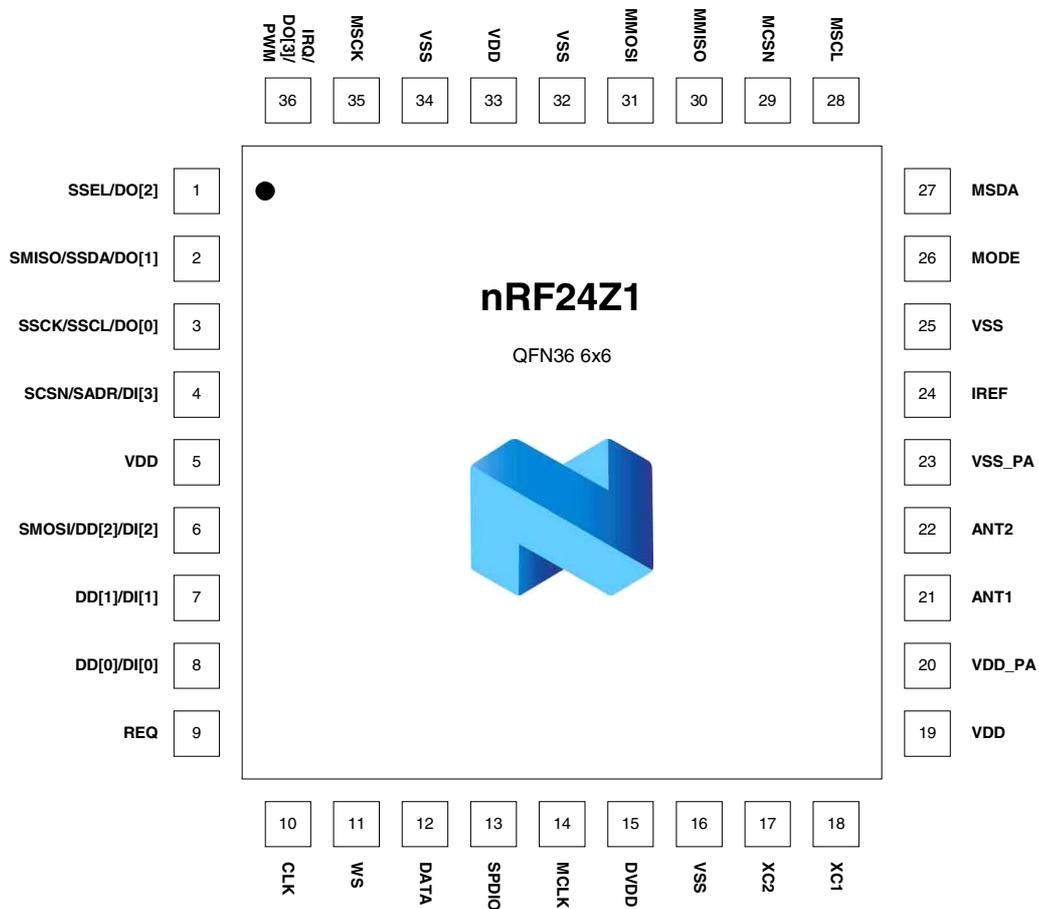


Figure 1-1: Pin assignment nRF24Z1



2 PIN FUNCTION

| Pin | Name | | Pin function | | Description | |
|-----|--------------|-------------|-----------------------------|-------------|---|------------------------------|
| | Slave IF | ARX GPIO | Slave IF | ARX GPIO IF | ATX/ARX with slave IF | ARX with GPIO IF |
| 1 | SSEL | DO[2] | Dig. In | Dig. Out | Slave interface select 1: 2-wire, 0: SPI | GPIO out bit #2 |
| 2 | SMISO /SSDA | DO[1] | Digital Output / Digital IO | | Slave SPI serial out / Slave 2-wire data (bidir) | GPIO out bit #1 |
| 3 | SSCK /SSCL | DO[0] | Dig. In | Dig. Out | Slave SPI clock / Slave 2-wire clock | GPIO out bit #0 |
| 4 | SCSN /SADR | DI3 | Digital Input | | Slave SPI slave select / Address select 2-wire slave | GPIO in bit #2 |
| 5 | VDD | | Power | | Power Supply (2.0-3.6 V DC) | |
| 6 | SMOSI /DD[2] | DI2 | Digital Input | | Slave SPI serial in / Direct data in bit #2 | GPIO in bit #3 |
| 7 | DD[1] | DI1 | Digital Input | | Direct data in bit #1 | GPIO in bit #1 |
| 8 | DD[0] | DI0 | Digital Input | | Direct data in bit #0 | GPIO in bit #0 |
| | | | ATX | ARX | | |
| 9 | REQ | | Dig. Out | Dig. In | I2S data request (programmable polarity) | |
| 10 | CLK | | Dig. IO | Dig. Out | I2S bit clock | |
| 11 | WS | | Dig. IO | Dig. Out | I2S word clock | |
| 12 | DATA | | Dig. In | Dig. Out | I2S data signal | |
| 13 | SPDIO | | Dig. In | Dig. Out | S/PDIF interface | |
| 14 | MCLK | | Dig. Out | | 256X sample rate clock to ADC or DAC | |
| 15 | DVDD | | Regulator output | | Digital voltage regulator output for decoupling | |
| 16 | VSS | | Power | | Ground (0V) | |
| 17 | XC2 | | Analog output | | Crystal Pin 2 | |
| 18 | XC1 | | Analog input | | Crystal Pin 1 | |
| 19 | VDD | | Power | | Power Supply (2.0-3.6 V DC) | |
| 20 | VDD_PA | | Regulator output | | DC output (+1.8V) for RF interface (ANT1, ANT2) | |
| 21 | ANT1 | | RF | | Antenna interface 1 | |
| 22 | ANT2 | | RF | | Antenna interface 2 | |
| 23 | VSS_PA | | Power | | Ground (0V) | |
| 24 | IREF | | Analog input | | Connection to external Bias reference resistor, or RESET if pulled to VDD | |
| 25 | VSS | | Power | | Ground (0V) | |
| 26 | MODE | | Digital Input | | nRF24Z1 function 1 : audio transmitter, 0: audio receiver | |
| 27 | MSDA | | Digital IO | | Master 2-wire bi-directional data | |
| 28 | MSCL | | Digital IO | | Master 2-wire bi-directional clock | |
| 29 | MCSN | | Digital Output | | Master SPI primary slave select (active low) | |
| 30 | MMISO | | Digital Input | | Master SPI serial input | |
| 31 | MMOSI | | Digital Output | | Master SPI serial output | |
| 32 | VSS | | Power | | Ground (0V) | |
| 33 | VDD | | Power | | Power Supply (2.0-3.6 V DC) | |
| 34 | VSS | | Power | | Ground (0V) | |
| 35 | MSCK | | Digital Output | | Master SPI clock | |
| 36 | IRQ | DO[3] / PWM | Digital Output | | Interrupt request | GPIO out bit #3 / PWM output |

Table 2-1 nRF24Z1 pin function



3 GLOSSARY OF TERMS

| Term | Description |
|--------|---|
| ADC | Analog to Digital Converter |
| ARX | audio receiver |
| ATX | audio transmitter |
| CD | Carrier Detect |
| CHPA | SPI clock phase |
| CLK | Clock |
| CPOL | SPI clock polarity |
| CRC | Cyclic Redundancy Check |
| DAC | Digital to Analog Converter |
| DSP | Digital Signal Processor |
| EEPROM | Electrical erasable programmable read only memory |
| Flash | Flash memory |
| GFSK | Gaussian Frequency Shift Keying |
| GPIO | General Purpose In Out |
| I2S | 3 wire audio serial interface |
| ISM | Industrial-Scientific-Medical |
| LPCM | Linear PCM (pulse code modulation) |
| MBZ | Must Be Zero (reserved for future extensions) |
| MCU | Micro Controller Unit |
| MP3 | MPEG3, audio compression format |
| PWM | Pulse Width Modulation |
| QOS | Quality Of Service |
| RX | Receive |
| S/PDIF | One wire serial digital audio format |
| SPI | Serial Peripheral Interface |
| TX | Transmit |
| 2-wire | 2-wire serial interface compatible with I2C |

Table 3-1 Glossary of terms nRF24Z1.



4 ARCHITECTURAL OVERVIEW

nRF24Z1 is a 4 Mbit/s single chip RF transceiver that operates in the world wide 2.4 GHz license free ISM band. The nRF24Z1 is based on the proven nRF24xx radio and ShockBurst™ platforms from Nordic Semiconductor.

The device offers a wireless channel for seamless streaming of LPCM or compressed audio in parallel with a low data rate control channel. To enable this, the device offers the following features in addition to the nRF24xx RF platform:

- Standard digital audio interfaces (I2S, S/PDIF)
- Fully embedded Quality of Service engine that handles all RF protocol and RF link tasks.
- SPI and 2-wire master and slave control interfaces
- GPIO pins

As all processing related to audio I/O, RF protocol and RF link management is embedded, the device offers a transparent audio channel with capacity of up to 1.54 Mbit/s, where no true time processing is needed. The nRF24Z1 can be utilized in systems without external microcontroller or used in conjunction with a simple microcontroller that only need to handle low speed tasks over the serial or parallel ports (ex: volume up/down).

A typical system using nRF24Z1 is illustrated in Figure 4-1

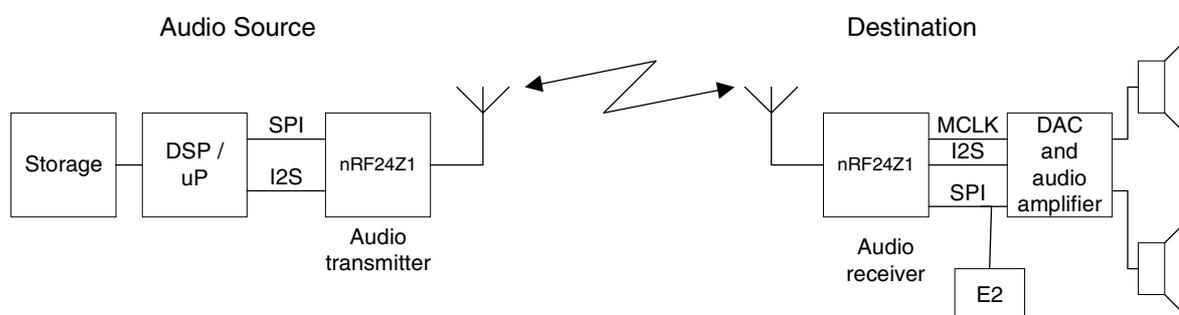


Figure 4-1 Typical audio application using nRF24Z1

In this system a DSP or micro controller feeds data from a storage device to an nRF24Z1 using standard audio format (I2S). An nRF24Z1 pair transfers the audio data and presents it to a stereo DAC on the other side. For other parts of the application, the nRF24Z1 link will in other words appear as an open channel (like a cable).

Initial configuration of nRF24Z1 is done by the micro controller through an SPI or 2-wire control interface. On the destination side, peripherals like a DAC can be controlled from the audio source side through the control channel offered by nRF24Z1. In designs without an external micro controller, configuration data can be loaded into the nRF24Z1 from an optional EEPROM/FLASH memory, enabling it to operate stand alone with limited feature set.



A wireless system streaming audio will have an asymmetrical load on the RF link since audio data is fed from an audio source (CD player) to a destination (loud speakers). From the destination back to the audio source only service and control communication is needed.

nRF24Z1 is used both on the audio source side (ex. in a CD player) transmitting audio data, and in the 'destination' (loud speaker) side receiving audio data. Due to the asymmetry, nRF24Z1 has two main modes set by external pin MODE, depending on whether it represents the transmitter or the receiver. The two modes have significant differences both in internal and I/O functionality.

To differentiate these two modes of operation, the following notation is introduced:

- Audio transmitter (ATX) – nRF24Z1 on the audio source side, transmitting audio data
- Audio receiver (ARX) – nRF24Z1 on the destination side, receiving audio data

Transmitter and receiver are here referring to the flow of the audio; the nRF24Z1 RF front end always runs a full two way link.

The nRF24Z1 control and data channel is a two way low data rate channel superimposed on the audio stream. The audio transmitter is designated master meaning that when a RF link is present the 2-wire, SPI, GPIO and internal registers in the audio receiver can be seen and controlled as a virtual extension of the audio transmitters own I/O and registers. The implications of this is that external devices like audio DAC or volume control components connected to the audio receiver effectively can be controlled by input to the audio transmitter. User actions (ex: push of a button) on the audio receiver side are similarly fed back to and can be processed on the audio transmitter side.

The following sections will give an overview of the I/O, main modules and functionality of nRF24Z1. Due to the differences in ATX and ARX, the overview will present the modes separately.



4.1 Audio transmitter

When nRF24Z1 is applied at the audio source side of the RF link, MODE must be high and nRF24Z1 becomes an audio transmitter (ATX). The block schematic of nRF24Z1 in ATX mode can be seen in Figure 4-2.

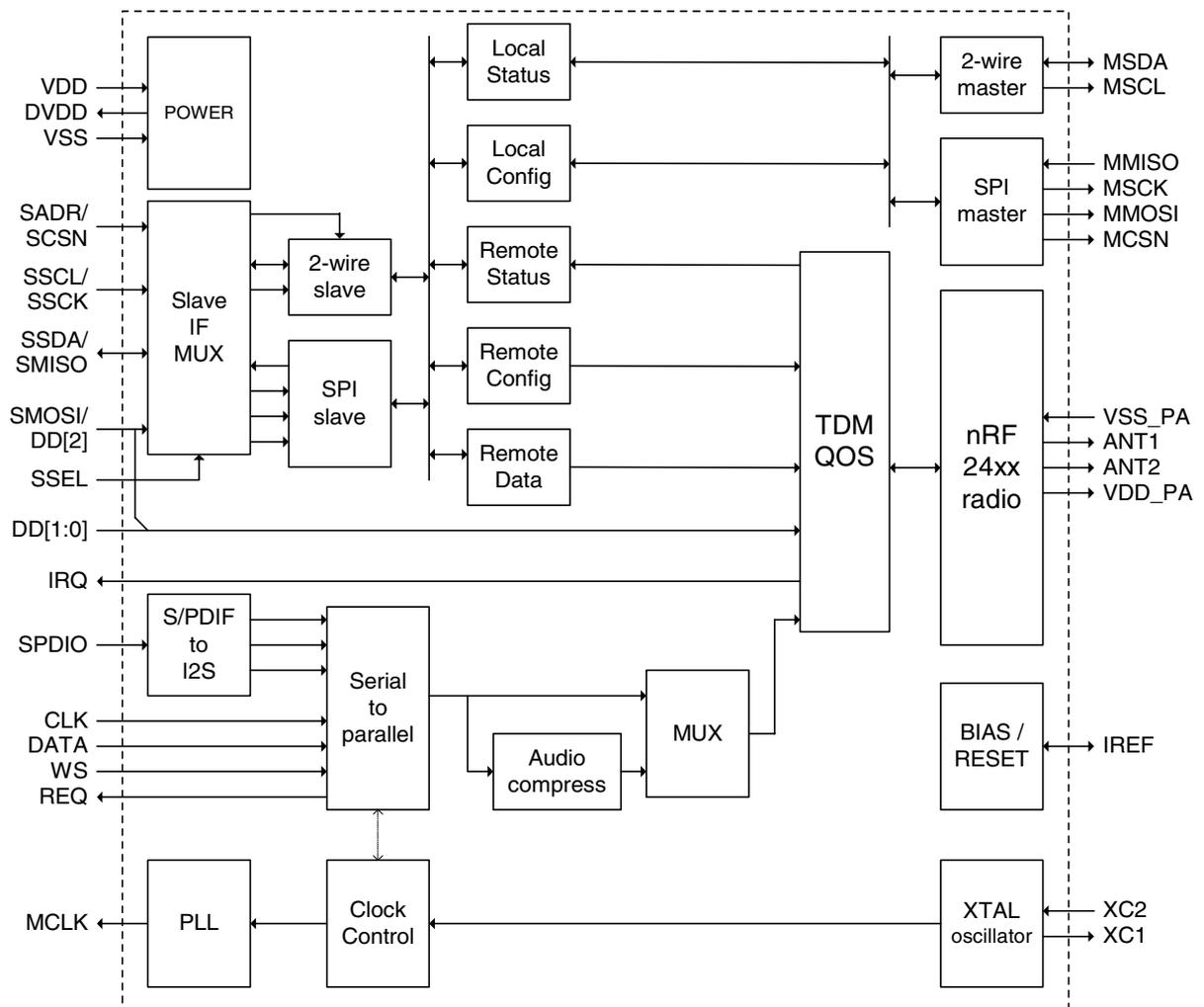


Figure 4-2 nRF24Z1 ATX mode block diagram

The I2S or S/PDIF interfaces can be used for audio data input, or alternatively, the device may stream other real-time data from a DSP over the I2S interface.

4.1.1 I2S audio input

For seamless input from audio sources physically close to nRF24Z1, I2S is the preferred interface. The I2S interface consists of pins CLK, DATA and WS. This interface supports the following sample rates: 8, 11.025, 12, 16, 22.05, 24, 32, 44.1, 48 and 96 kHz. Data may

**nRF24Z1 wireless audio streamer**

be in 16, 20 or 24 bit format.¹ The nRF24Z1 can be configured to automatically detect the applied data rate.

I2S may also be used with an external stereo ADC for analog audio sources. The nRF24Z1 offers a 256 times audio sampling rate clock (f_s) on the MCLK pin to be used as system clock for the ADC.

A REQ output is available for pacing the data-flow when streaming MP3 or other data streams over the I2S interface.

4.1.2 S/PDIF audio input

For audio sources physically more remote, the ATX offers a (CMOS level) S/PDIF input on pin SPDIO. This interface supports 32, 44.1 or 48 kHz sampling rates with resolution of 16, 20 or 24 bit. It supports both linear and nonlinear audio according to IEC standards, see Ch. 7.4 for details.

4.1.3 Serial control (slave) interfaces

When ATX is controlled by an external MCU, configuration and control data both for the audio transmitter itself and a linked audio receiver may be entered via a 2-wire or SPI slave serial interface. The same interface is used for reading back status information. The register map is identical for both interfaces, but only one of the interfaces, selected by SSEL pin, may be used in a given application.

The two interfaces are:

SSEL = 0; SPI (pins SCSN, SSCK, SMISO, SMOSI).

SSEL = 1; 2-wire (pins SADR, SSCL and SEDA)

Pin SADR is not part of a standard 2-wire interface but selects one of two possible bus addresses for the nRF24Z1.

4.1.4 Master interfaces

For standalone operation of nRF24Z1, a serial EEPROM or FLASH memory may be connected to an SPI or 2-wire master interface. If a memory is present at any of these interfaces during power up or reset, the device will read default configuration data from the memory.

The SPI master is found on pins MCSN, MMISO, MMOSI and MSCK and 2-wire master on pins MSDA and MSCL.

¹ This specification item is for the I2S input interface. Not all of these formats can be transferred within the available 1.54 Mbit/s data rate.



4.1.5 Direct data input pins

The ATX has 2 general purpose input pins, DD[1:0], which may be transmitted directly to the ARX without the help from external MCU. When SSEL is set high (2-wire interface selected), one additional direct data pin (DD[2]) is available. If the logic level on pins DD[2:0] are mirrored (copied) over the control link, pins DO[2:0] on the audio receiver will carry the mirrored signal.

These pins may hence be used to switch on/off audio receiver peripherals without microprocessor activity.

4.1.6 Interrupt output

The nRF24Z1 can interrupt the external application through pin IRQ based on a number of sources (no audio input detected, loss of RF communication etc.).

Once IRQ has triggered external MCU, interrupt status can be read through the serial slave interface.

4.2 Audio Receiver

When nRF24Z1 is put at the destination side of the RF link, MODE must be low and nRF24Z1 becomes the audio receiver (ARX). The block schematic of nRF24Z1 in ARX mode can be seen in Figure 4-3. I2S or S/PDIF are now used for audio or other real time data output.

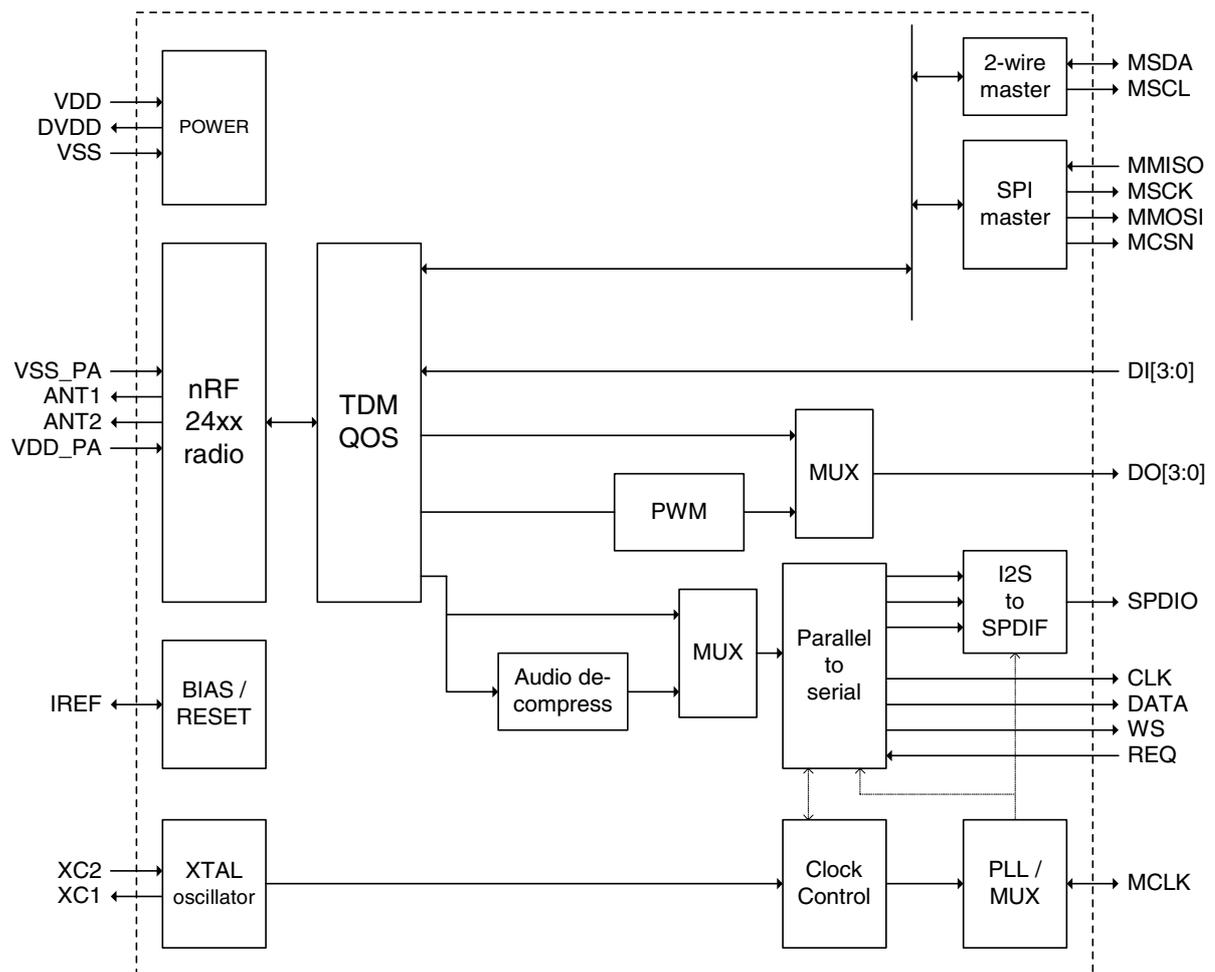


Figure 4-3 nRF24Z1 ARX mode block diagram

After a link is established, the user can control the SPI and 2 wire master on the ARX from the ATX. This feature enables the ATX to remotely control serial peripheral devices on the ARX (audio DACs, amplifiers etc.).

4.2.1 I2S audio output

Audio output to devices physically close to nRF24Z1 (typically a stereo DAC) are normally driven by the I2S output (pins CLK, DATA and WS). This interface supports the following sample rates: 8, 11.025, 12, 16, 22.05, 24, 32, 44.1 and 48 kHz. Data are in 16 bit format.

In audio receiver mode the MCLK pin provides 256 times f_s clock for an external DAC.

A REQ input is available for pacing the data-flow when streaming MP3 or other data streams over the I2S.



4.2.2 S/PDIF audio output

For physically more remote audio devices, the audio receiver provides an S/PDIF (full swing CMOS) output on pin SPDIO. This interface supports 32, 44.1 and 48 kHz, 16 or 24 bit data. It supports both linear and nonlinear audio according to IEC standards, see Ch. 7.4 for details.

4.2.3 Master interfaces

A serial EEPROM or FLASH memory may be connected to a SPI or 2-wire master interface. If a memory is present at any of these interfaces during power up or reset, the device will read default configuration data from that memory. If not hard coded default values will be used.

During audio receiver configuration, the SPI master (pins MMSCK, MMISO, MMOSI, MCSN) is operated at 1MHz or 0.5MHz with the SPI format set to CPOL=0,CHPA=0 for EEPROM/FLASH compatibility. After a link has been established, the user may control the SPI master from the audio transmitter. The available clock speed is up to 8 MHz over the full operational range of the device.

During start-up, the audio receiver operates the 2-wire master (MSDA, MSCL) interface at 100 kHz. After a link has been established, the user may control the 2-wire master from the audio transmitter to 100 kHz, 400 kHz or 1 MHz.

4.2.4 Serial control (slave) interfaces

When the ARX is controlled by an external MCU, configuration and control data for the audio receiver may be entered via a 2-wire or SPI slave serial interface. The same interface is used for reading back status information. The register map is identical for both interfaces, but only one of the interfaces, selected by the SSEL pin, may be used in a given application.

The two interfaces are:

SSEL = 0; SPI (pins SCSN, SSCK, SMISO, SMOSI).

SSEL = 1; 2-wire (pins SADR, SSCL and SSDA)

Pin SADR is not part of a standard 2-wire interface but selects one of two possible bus addresses for the nRF24Z1.

4.2.5 Parallel port and PWM

Alternatively to the serial slave interfaces, the ARX can be configured with an 8 bit parallel port, which can be controlled and read from the audio transmitter. There are 4 input pins DI[3:0] that are continuously monitored when a link is up. Changes on any of these inputs will be sent back to the audio transmitter where it can be accessed in a register (via the



serial control interface). The audio receiver can also be programmed to wake up from power down mode upon a change on any of these pins.

There are 4 outputs DO[3:0] that are controlled from the audio transmitter. Any of these may be programmed for high current in order to drive LEDs or for standard CMOS to control of other devices on the audio receiver board.

DO3 may be programmed to provide a PWM signal, where the output duty cycle is programmable with 8-bit resolution from the audio transmitter. Note that this PWM cannot be used as an audio DAC.

The output pins DO[3:0] may also function as slave select signals if multiple slaves are present on the ARX SPI master bus.

4.3 Blocks common to audio transmitter and receiver

4.3.1 XTAL Oscillator

The crystal oscillator will provide a stable reference frequency with low phase noise for the radio and audio functions. See section 17.2 for full Crystal Specification.

4.3.2 Radio Transceiver

The RF transceiver part of the circuit is a member of nRF24xx family of low power highly integrated 2.4GHz ShockBurst™ transceivers. The transceiver interface is optimized for high speed streaming of up to 4 Mbps. Output power and some radio protocol parameters can be controlled by the user via the quality of service module.

4.3.3 Quality of Service (QoS) engine

The primary function of the QoS engine is to ensure robust communication between the ATX and the ARX in an audio streaming application.

Various data streams with different properties are handled. The available bandwidth is shared between audio data, service data and remote data.

Data integrity is ensured through a number of RF protocol features:

1. Packets of data are sent in frames with each packet consisting of an *RF address*, *payload* and *CRC*.
2. Packets that are lost or received with errors are handled by the error correction level of the quality of service engine; a two way, acknowledge protocol:
When a packet is received by ARX, it is registered and CRC is verified. After ARX has received a frame, it sends a packet back to ATX acknowledging those packets which were successfully transferred. Packets lost or received with errors are re-sent from ATX in the next frame.



3. The information (audio data) is dispersed over the 2.4 GHz band by use of an adaptive frequency hopping algorithm. This enables the nRF24Z1 link to cope with RF propagation challenges like reflections, multi-path fading and avoid heavily trafficked areas of the 2.4 GHz band. Handling co-existence scenarios with contemporary RF systems such as Bluetooth, ZigBee, WLAN/WiFi as well as other nRF applications, is increasingly important.

The nRF24Z1 is constantly monitoring the quality of the RF link. The link quality information is available for external control devices in registers. nRF24Z1 can also be set to interrupt external controller devices upon poor link quality before RF link is lost. An external controller device can hence take further actions to improve link quality or warn end user if RF link margins are poor.

The secondary function of the QoS module is to run a link initialization algorithm which manages initial connect and re-connect if link is lost (ex: out of range) between paired nRF24Z1's. Several schemes are available to enable nRF24Z1 connection without end user involvement.

4.3.4 Audio compression / decompression

Default operation of the nRF24Z1 is streaming of uncompressed audio. However, some optional low delay audio compression options are available. This function can be enabled by the user in order to conserve power or to increase the dynamic range with a constant signal to noise ratio for 24-bit input signals.

4.3.5 Power

The power section of nRF24Z1 offers linear regulated supply to all internal parts of the device. This makes the device very robust towards external voltage supply noise and isolates (audio) devices in an application from noise generated by the nRF24Z1.

4.3.6 IREF / RESET

The IREF pin sets up the bias reference for the nRF24Z1 by use of an external resistor. Pulling IREF to VDD will reset the device. When IREF pin is released, nRF24Z1 runs a full configuration procedure.



5 OPERATION OVERVIEW

5.1 Power on / RESET sequence

When supply is applied, nRF24Z1 goes into power on reset. The reset is held until supply voltage is kept above minimum supply voltage for a few milliseconds. Pulling IREF to VDD will also put the device into reset.

When reset (power on or IREF high) is released the device needs to be configured. There are 2 ways nRF24Z1 can be configured:

1. After reset nRF24Z1 will look for an external EEPROM/FLASH memory on the SPI master interface. If such a memory is present, configuration data is loaded, which means that all registers values are read from the external memory. If no memory is present on the SPI master interface, the procedure is repeated on the 2-wire master interface. These data will override any initial contents of nRF24Z1 registers.
2. If no external memory is present:
For both ATX and ARX an external micro processor must configure the nRF24Z1 through the slave SPI or 2-wire serial interface, otherwise hard coded initial register contents is used.

NOTE:

A combination of the two power-up sequences may well be used. One likely scenario is that ATX is configured by external MCU and ARX is configured from an external EEPROM/FLASH memory.

nRF24Z1 will now start a link initialization procedure based on the link configuration data. The value of the MODE pin determines whether it will be in ATX or ARX mode.

5.2 RF Link initialization

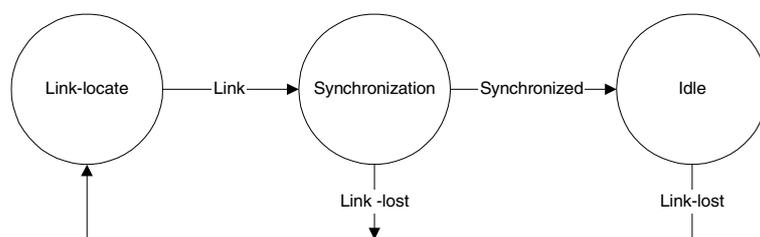


Figure 5-1 : link initialization algorithm

5.2.1 Idle state

The nRF24Z1 link initialization algorithm will be in its idle state when a link is established, and the channel hopping engine is initiated and synchronized.



5.2.2 Link-locate state

When the link between ATX and ARX is broken, a special link-locate routine is initiated on both sides to establish a new link, see Figure 5-1. During initialization nRF24Z1 uses the NLCH first channels of the frequency hopping table (see ch. 8.3).

5.2.2.1 Link-locate on ATX

ATX tries to establish a link with ARX by iteratively sending short search packages on all available channels until acknowledge is received from ARX. ATX will send one package on each channel and wait for acknowledge for a specific time which is long enough to secure that ARX has time to respond. The accumulated time used by ATX while looping through all available channels is here defined as the ATX-loop-time. After receiving an acknowledge package from ARX, ATX will enter the synchronization state as described in Figure 5-1.

5.2.2.2 Link-locate on ARX

ARX tries to establish a link with ATX by listening for incoming search packages on all available channels until such is received. When a search package is received, ARX will proceed by sending one acknowledge package to confirm a feasible link. ARX will listen for incoming search packages on each channel for a fixed time which is larger than the ATX-loop-time, which will guarantee at least one search package to get through on each available channel used by ARX, as long as this channel is not being occupied by another radio device. After sending the acknowledge package, ARX will enter the synchronization state.

5.2.3 Synchronization state

This state takes care of synchronizing the channel hopping engine on ATX and ARX, to secure that both parts follows the same hopping sequence. Which channel to start from is implicitly found during the link-locate state.

5.3 Audio streaming

The audio data fed to the audio interfaces on an nRF24Z1 in ATX mode can be one of several different common digital audio stream formats:

I2S (audio serial) interface:

- Left justified, I2S and right justified.

S/PDIF interface:

- Consumer Linear PCM Audio described in IEC 60958-3. nRF24Z1 has a single ended CMOS interface, so to fulfil the electrical requirements external adaptation circuitry is needed.

**nRF24Z1 wireless audio streamer**

- **Non-Linear PCM Audio.** As described in IEC 61937-1 (General) and IEC 61937-2 (Burst-info). The nRF24Z1 is transparent to the specific audio compression algorithms used, so it covers all the described formats in IEC 61937-3 to 61937-7.

In ATX the audio stream formats are converted to the nRF24Z1 RF protocol and transferred over the air.

In the ARX the data are validated and converted back to audio stream format and fed to the corresponding audio output interface.

5.4 Audio receiver clock rate recovery

In all RF systems streaming ‘true time’ data, maintaining equal data rates on both sides of RF link is a big challenge. In other words; keeping the master clock (MCLK) for the DAC on the receiving side, equal to the clock used to feed data into the RF device on the transmitter side.

If these two clocks are not equal the receiving end will either run out of samples for the DAC or overflow hence need to skip some.

Usually this problem is solved by use of very tight tolerance crystals (expensive) or extensive digital filtering (high current consumption) only masking or interpolating the bits missing in the stream.

nRF24Z1 solves this problem without tight tolerance crystal or extensive digital filtering.

As long as the nRF24Z1 quality of service engine is able to maintain an RF link, the ARX (audio receiver) locks its master clock output (MCLK) to the speed of the audio stream which is fed into the ATX. The MCLK signal on the ARX side is hence locked to the reference (crystal) of the device (DSP, MCU, DECODER) feeding the audio data to the ATX and not the crystal of the nRF24Z1 devices (ATX or ARX) themselves.

One exception; if the MCLK output option is used in audio transmitter (clocking an external ADC for instance) the crystal on the nRF24Z1 in ATX mode is the reference for the audio speed on the entire nRF24Z1 link.

This offers the end application a true loss less audio channel.

5.5 Data link

There is a 2-way, low bit rate, robust, control and data link running in parallel with the audio stream. This data link is a part of the quality of service overhead, i.e. difference between on the air data rate (4 MBit/s) and audio data rate 1.5 MBit/s. Data link rate can hence not be traded for higher audio data rate. The functionality of the control and data link is illustrated in Figure 5-2.

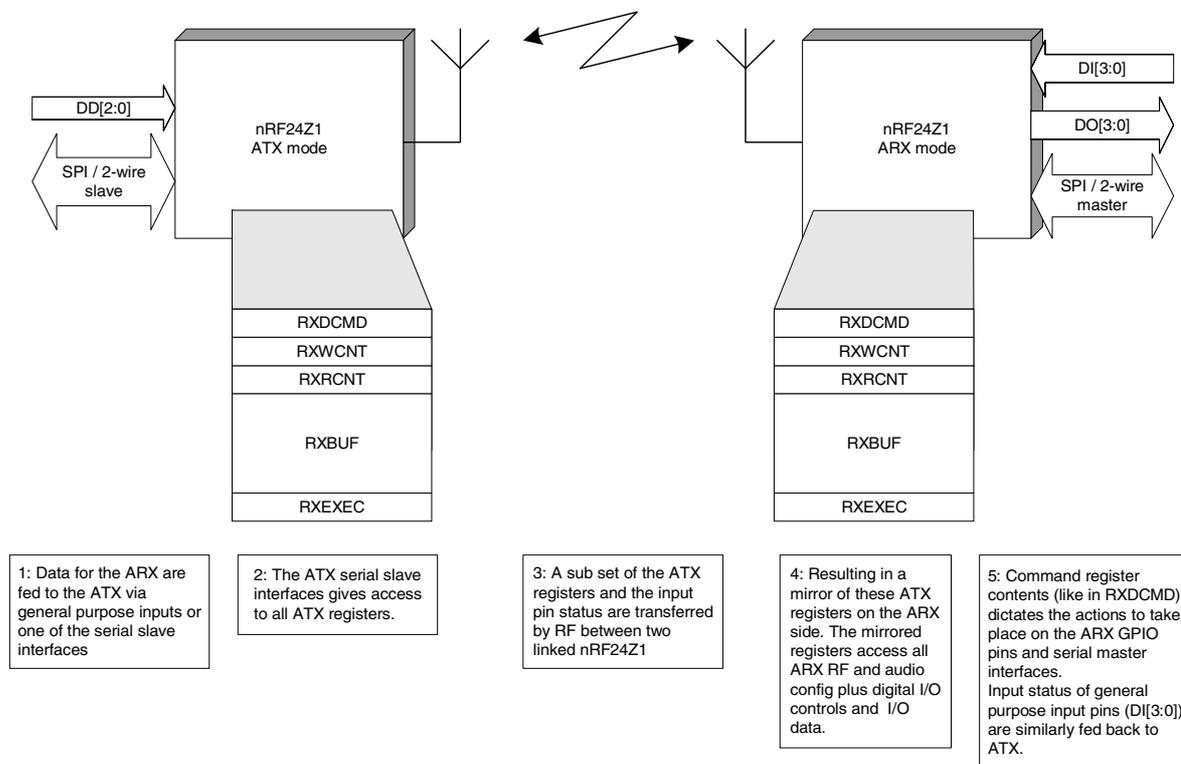


Figure 5-2 nRF24Z1 control and data link

Through the control and data link the ATX has write access to a majority of registers which are related to ARX configuration. ATX can thereby access ARX GPIO (for LED's etc.) and the ARX 2-wire and SPI master interface for configuring of DAC's, volume control and other peripheral functions.

5.6 Power down mode

nRF24Z1 has a power saving mode called "Power down". In this mode, the quality of service engine is shut down, and only a low frequency oscillator and some timers are running. The power down mode can be left upon sleep timer time out or on external pin event. The ATX and ARX will now go up and start the link initialization routine as described in section 5.2. The sleep timers also enable the nRF24Z1 to shut down again if no counterpart is found on the air or no audio input is detected. ARX may also be put in and out of power down mode by toggling a pin.



6 NRF24Z1 REGISTER MAP

nRF24Z1 contains several control and status registers, which are listed in the table below. The registers may be accessed by an external MCU via the (SPI or 2-wire) slave interface. The registers are organized functionally into 6 groups; ATX, Link and ARX control and status, and Data link. All registers are present both in audio transmitter and audio receiver. The initial value of all registers is read from EEPROM (if present) immediately after reset, otherwise the initial values in Table 6-1 apply.

6.1 Access from audio transmitter side

If an MCU on the ATX side writes to a register, only the ATX version of the register is updated. Registers TXCSTATE, LNKSTATE, RXCSTAT, RXEXEC controls whether also the ARX version of the register is updated via the data link.

- If it is an ATX control register, register TXCSTATE controls if also the audio receiver version of the register is updated via the data link
- If it is a Link control register, register LNKSTATE controls if also the audio receiver version of the register is updated via the data link
- If it is an ARX control register, register RXCSTATE controls if also the audio receiver version of the register is updated via the data link

See ch. 12 for details about how control registers are updated via the data link, and Table 7-10 about data link registers.

A MCU on the audio transmitter side can read all registers on its side, plus the Link status, ARX status and data link registers, which are read from audio receiver via the data link.

6.2 Access from audio receiver side

If an MCU on the audio receiver side writes to a register, only the audio receiver version of the register is written, and it is not sent via the data link to the audio transmitter. This implies that an MCU on the audio transmitter will not know about it, but as mentioned above, ATX MCU may read status registers via the link anytime.

An MCU on the audio receiver side can read all registers on its side, but it cannot read anything via the link. In brief ARX MCU only has local access, while ATX MCU controls the data link.



| Address Hex | Register | R/W | Initial Hex | Description |
|--------------------------------------|------------|-----|-------------|---------------------|
| <i>ATX registers</i> | | | | |
| 0x01 | TXSTA | R/W | 0x40 | Table 7-4, page 28 |
| 0x75 | TXDD | R | 0x00 | Table 7-15, page 41 |
| 0x02 | INTSTA | R | 0x00 | Table 9-1, page 55 |
| 0x5A | TXMOD | R/W | 0x01 | Table 7-6, page 30 |
| 0x51 | TXFMT | R/W | 0x00 | Table 7-5, page 29 |
| 0x52 | TXLAT | R/W | 0x00 | Table 8-6, page 52 |
| 0x53 | INTCF | R/W | 0x00 | Table 9-1, page 55 |
| 0x54 | I2SCNF_IN | R/W | 0x00 | Table 7-6, page 30 |
| 0x55 | I2SRAT | R/W | 0x00 | Table 7-7, page 31 |
| 0x56 | TXPWR | R/W | 0x03 | Table 8-7, page 53 |
| 0x57 - 0x58 | TXSTI[0:1] | R/W | 0x00 | Table 11-2, page 59 |
| 0x59 | TXWTI | R/W | 0x00 | Table 11-2, page 59 |
| 0x4D | TXLTI | R/W | 0x00 | Table 11-2, page 59 |
| 0x50 | TXRESO | R/W | 0x00 | Table 10-1, page 56 |
| 0x5B | TXCSTATE | R/W | 0x00 | Table 12-1, page 62 |
| <i>LINK status registers</i> | | | | |
| 0x03 | LNKSTA | R | 0x00 | Table 8-5, page 52 |
| 0x04 | LNKQ | R | 0x00 | Table 8-4, page 51 |
| 0x05 | LNKERR | R | 0x00 | Table 8-4, page 51 |
| <i>LINK control registers</i> | | | | |
| 0x0C- 0x31 | CHTAB | R/W | 0x00 | Table 8-2, page 49 |
| 0x32 | BCHD | R/W | 0x00 | Table 8-3, page 50 |
| 0x33 | NBCH | R/W | 0x00 | Table 8-3, page 50 |
| 0x34 | NACH | R/W | 0x00 | Table 8-3, page 50 |
| 0x35 | NLCH | R/W | 0x00 | Table 8-3, page 50 |
| 0x36 | LNKMOD | R/W | 0x00 | Table 8-5, page 52 |
| 0x37 | LNKWTH | R/W | 0xff | Table 8-4, page 51 |
| 0x38 | LNKETH | R/W | 0xff | Table 8-4, page 51 |
| 0x39 - 0x3D | ADDR[0:4] | R/W | 0x01 | Table 8-1, page 48 |
| 0x3E | LNKCSTATE | R/W | 0x00 | Table 12-1, page 62 |
| <i>ARX status registers</i> | | | | |
| 0x06 | RXSTA | R | 0x00 | Table 7-16, page 43 |
| 0x07 | RXPIN | R | 0x00 | Table 7-21, page 45 |
| <i>ARX control registers</i> | | | | |
| 0x4A | RXMOD | R/W | 0x00 | Table 7-8, page 32 |
| 0x41 | RXPIO | R/W | 0x00 | Table 7-19, page 44 |
| 0x42 | RXPWME | R/W | 0x00 | Table 7-20, page 45 |
| 0x43 | RXPWMD | R/W | 0x00 | Table 7-20, page 45 |
| 0x44 | I2SCNF_OUT | R/W | 0x00 | Table 7-8, page 32 |
| 0x45 | RXWAKE | R/W | 0x28 | Table 11-1, page 58 |



nRF24Z1 wireless audio streamer

| | | | | |
|----------------------------|------------|-----|------|---------------------|
| 0x49 | RXPWR | R/W | 0x00 | Table 8-7, page 53 |
| 0x46 - 0x47 | RXSTI[0:1] | R/W | 0x00 | Table 11-2, page 59 |
| 0x48 | RXWTI | R/W | 0x00 | Table 11-2, page 59 |
| 0x4C | RXLTI | R/W | 0x00 | Table 11-2, page 59 |
| 0x40 | RXRESO | R/W | 0x00 | Table 10-1, page 56 |
| 0x4B | RXCSTATE | R/W | 0x00 | Table 12-1, page 62 |
| Data link registers | | | | |
| 0x70 | RXDCMD | R/W | 0x82 | Table 7-10, page 35 |
| 0x71 | RXWCNT | R/W | 0x00 | Table 7-11, page 36 |
| 0x72 | RXRCNT | R/W | 0x00 | |
| 0x60-0x6f | RXBUF | R/W | 0x00 | |
| 0x74 | RXEXEC | R/W | 0x00 | |
| Test registers | | | | |
| 0x7E | TESTREG | R/W | 0x00 | Table 13-1, page 64 |
| 0x7F | TESTCH | R/W | 0x00 | |
| 0x7D | REVBYT | R | 0x00 | Revision byte |

Table 6-1 nRF24Z1 register listing

In the following chapters, the nRF24Z1 registers are described sorted by function.

6.3 Register map in external EEPROM

The table below shows the layout of the first few bytes of the EEPROM image.

| Register address | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------|--------------------------------|---|---|---|---|------------------------|---|---|
| 0: | Reserved (must be 00) | | | | | Reserved (must be 011) | | |
| 1: | Reserved | | | | | | | |
| 2: | Reserved | | | | | | | |
| 3 | value of register address 0x00 | | | | | | | |
| 4 | value of register address 0x01 | | | | | | | |
| ... | ... | | | | | | | |
| 127 | value of register address 0x7C | | | | | | | |

Table 6-2 : EEPROM layout

Bytes 3 to 127 are used to set initial values into all nRF24Z1 registers, so that initial value of register with address n, must be put in byte n+3 of EEPROM.



7 DIGITAL I/O

This section describes the digital I/O pins, control registers and important interface timing of the nRF24Z1.

The digital I/O pins of the nRF24Z1 are divided into three groups:

1. Audio interfaces
2. Serial master interfaces
3. Control and GPIO interfaces

7.1 Digital I/O behaviour during RESET

During reset all digital pins, except the master SPI interface output pins, are set as inputs to avoid driving conflicts. The master SPI pins MCSN, MSCK and MMOSI are set to output high, that is, in inactive SPI state, ready to read EEPROM configuration data, which is the first thing to happen after reset. All pins will remain in their respective direction until one of the configuration read routines described in section 5.1 is finished, and thereafter I/O pins are set according to new configuration data.

7.2 Audio interfaces

The audio interface group are defined as the I2S and S/PDIF interfaces plus the MCLK and REQ pins.

| Pin name | Function |
|----------|---|
| CLK | bit clock |
| WS | word sync clock |
| DATA | audio data |
| MCLK | 256 * audio fundamental sample rate output, see Table 7-4 . |
| REQ | data request, used for burst type data, see Table 7-7. |
| SPDIO | S/PDIF serial input or output, see Ch. 7.4. |

Table 7-1 serial audio port pins

7.3 I2S Audio Interface

The nRF24Z1 has a three-wire serial audio port which can be configured to be compatible with different serial audio formats. In ATX mode, the audio port is in slave (receive) mode. In ARX mode, the audio port is in master (transmit) mode. The audio port consists altogether of 6 pins, see Table 7-1.

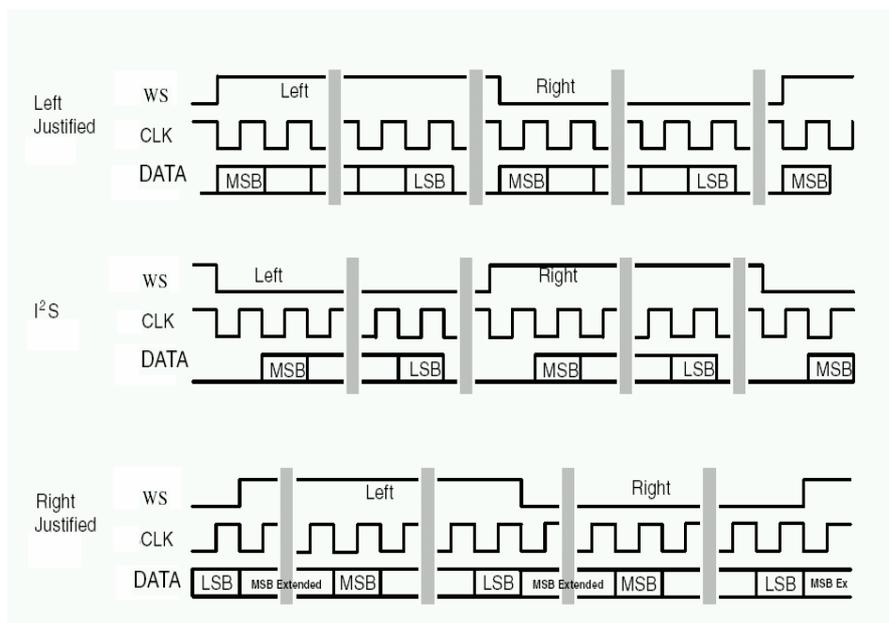


Figure 7-1 : some serial audio formats

| audio format | I2SCNF[3:0] value |
|-----------------|-------------------|
| Left justified | 0xA |
| I2S | 0x0 |
| Right Justified | 0xB |
| | |

Table 7-2 : Example of I2SCNF setting for some common serial audio formats applies to both I2SCNF_IN and I2SCNF_OUT registers.

See also Table 7-7 and Table 7-8.

7.4 S/PDIF Audio Interface

nRF24Z1 supports the following formats of the S/PDIF interface:

- Consumer Linear PCM Audio described in IEC 60958-3. nRF24Z1 has a single ended CMOS interface, so to fulfil the electrical requirements external adaptation circuitry is needed.
- Non-Linear PCM Audio, as described in IEC 61937-1 (General) and IEC 61937-2 (Burst-info). nRF24Z1 is transparent to the specific audio compression algorithms used, it just transfers what comes in on the input side to the receiver side. Except that only the 32 first bits of channel status information bits are transferred, and no user data bits.


nRF24Z1 wireless audio streamer

Only one of the audio interfaces can be active at one time. The active audio interface for ATX is chosen by bit 2 in register TXMOD (0= I2S is default, see Table 7-6), and for ARX by bit 2 in register RXMOD.

7.5 Audio interface functionality

The functionality and direction of the pins in the audio interfaces are listed in Table 7-3.

| nRF24Z1 Audio interfaces | | MODE=0 (ARX) | | | | MODE =1 (ATX) | | | | |
|--------------------------------|-------------|---------------------|-----------|------------------------|-----------|---------------------|----------------|-----------------|------------------------|-----------|
| | | TXMOD[2]=0 (I2S) | | TXMOD[2]=1 (S/PDIF) | | TXMOD[2]=0 (I2S) | | | TXMOD[2]=1 (S/PDIF) | |
| Pin # | Pin name | Function | Direction | Function | Direction | Function | Audio slave | Audio master | Function | Direction |
| | | | | | | | Direction | | | |
| 9 | REQ | REQ | IN | X | IN | REQ | OUT | OUT | X | OUT |
| 10 | CLK | CLK | OUT | X | IN | CLK | IN | OUT | X | IN |
| 11 | WS | WS | OUT | X | IN | WS | IN | OUT | X | IN |
| 12 | DATA | DATA | OUT | X | IN | DATA | IN | IN | X | IN |
| 13 | SPDIO | X | IN | SPDIO | OUT | X | IN | IN | SPDIO | IN |
| | | RESET* | OUT | | | RESET* | OUT | OUT | | |
| 14 | MCLK | MCLK | OUT | X | OUT | MCLK | OUT | OUT | X | OUT |

Table 7-3 Audio interface pin functions

* If S/PDIF is not used for audio, the SPDIO pin can be used as RESET (output) to external devices. Please see chapter 10 for further details.



7.5.1 ATX audio interface control

The nRF24Z1 in ATX mode can be configured to automatically detect the rate of a digital audio stream from an external master on the I2S interface (except 16 kHz). Register 0x01 holds the status of this detection for optional read back to an external MCU.

| Address Hex | Register | R/W | Initial Hex | Description | |
|-------------|---|-------------|-------------|-------------------------------|---|
| 0x01 | TXSTA | R/W | 0x40 | ATX audio input rate register | |
| | | | | Bit | Interpretation |
| | | | | 7 | Reserved |
| | | | | 6:5 | Audio rate scale factor (I2S only) if TXSTA bit 0 is set : audio rate is auto detected by hardware if TXSTA bit 0 is not set : audio rate must be set by user |
| | | | | 00 | 0.25 |
| | | | | 01 | 0.5 |
| | | | | 10 | 1 |
| | | | | 11 | Reserved |
| | | | | 4:3 | Audio fundamental rate if TXSTA bit 0 is set : audio rate is auto detected by hardware if TXSTA bit 0 is not set : audio rate must be set by user |
| | | | | 00 | 48 kHz |
| | | | | 01 | 44.1 kHz |
| | | | | 10 | 32 kHz |
| | | | | 11 | Illegal or no input detected |
| 2:1 | Reserved | | | | |
| 0 | Value | Description | | | |
| 1 | Audio input rate is auto detected by hardware | | | | |
| 0 | Audio input rate must be set by user | | | | |

Table 7-4 Audio input status

New values in TXSTA will take effect after ATX and ARX is reconfigured, this is done by setting LNKMOD[4] to “1”.

Default nRF24Z1 transfers uncompressed audio, but it has some options to transfer companded audio data. This can be used to save power, or to accommodate transfer of 24 bit audio, which uncompanied would exceed the radio bandwidth. This is a lossy



nRF24Z1 wireless audio streamer

compression/decompression in the way that superfluous LSB's are lost. Here is an example of how 24 to 16 bit compand works:

If the sample values within an audio frame are within +/- 200000, 3 LSB are cut, and if sample values of the next frame are within +/- 60000, only 1 LSB is lost. So the number of LSB lost varies dynamically with maximum sample value for each frame, and this limits the relative error.

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | |
|-------------|-----------------------------------|-----|-------------|---|-------|----------------|---|-------------------|---|-----------------------------------|---|----------------------------------|----|------------------|-------|---------------|
| 0x51 | TXFMT | R/W | 0x00 | Transmit data format | | | | | | | | | | | | |
| | | | | <table border="1"> <thead> <tr> <th>Value</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>16 bit linear PCM</td> </tr> <tr> <td>1</td> <td>24 bit linear companded to 16 bit</td> </tr> <tr> <td>4</td> <td>16 bit linear companded to 8 bit</td> </tr> <tr> <td>64</td> <td>I2S carries data</td> </tr> <tr> <td>other</td> <td>Reserved, MBZ</td> </tr> </tbody> </table> | Value | Interpretation | 0 | 16 bit linear PCM | 1 | 24 bit linear companded to 16 bit | 4 | 16 bit linear companded to 8 bit | 64 | I2S carries data | other | Reserved, MBZ |
| Value | Interpretation | | | | | | | | | | | | | | | |
| 0 | 16 bit linear PCM | | | | | | | | | | | | | | | |
| 1 | 24 bit linear companded to 16 bit | | | | | | | | | | | | | | | |
| 4 | 16 bit linear companded to 8 bit | | | | | | | | | | | | | | | |
| 64 | I2S carries data | | | | | | | | | | | | | | | |
| other | Reserved, MBZ | | | | | | | | | | | | | | | |

Table 7-5 : register TXFMT.

New values in TXFMT take effect after ATX and ARX are reconfigured, this is done by setting LNKMOD[4] to "1".

The Audio interfaces in ATX mode are controlled by registers listed in Table 7-6.

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | | | | | | | | | | | |
|-------------|--|-----|-------------|--|---|-----------------------|---|------------------------------|---|---------------------------------|---|-------------------------------------|---|--|---|--------------------------------------|-----|-----------------------|--|-----------------------|--|------------------------|--|--------------------------|--|------------------------|
| 0x5A | TXMOD | R/W | 0x00 | Operation modes for audio transmitter | | | | | | | | | | | | | | | | | | | | | | |
| | | | | <table border="1"> <tbody> <tr> <td>7</td> <td>RF transmitter enable</td> </tr> <tr> <td>6</td> <td>Audio transmitter power down</td> </tr> <tr> <td>5</td> <td>Enable wakeup on changing DD[1]</td> </tr> <tr> <td>4</td> <td>REQ pin polarity (0 for active low)</td> </tr> <tr> <td>3</td> <td>Enable direct data from pins DD[2:0]. DD[2] is only available if SSEL=1</td> </tr> <tr> <td>2</td> <td>S/PDIF enable (default input is I2S)</td> </tr> <tr> <td>1:0</td> <td>MCLK output control**</td> </tr> <tr> <td></td> <td>00 MCLK off (logic 0)</td> </tr> <tr> <td></td> <td>01 Output 256 x 48 kHz</td> </tr> <tr> <td></td> <td>10 Output 256 x 44.1 kHz</td> </tr> <tr> <td></td> <td>11 Output 256 x 32 kHz</td> </tr> </tbody> </table> | 7 | RF transmitter enable | 6 | Audio transmitter power down | 5 | Enable wakeup on changing DD[1] | 4 | REQ pin polarity (0 for active low) | 3 | Enable direct data from pins DD[2:0]. DD[2] is only available if SSEL=1 | 2 | S/PDIF enable (default input is I2S) | 1:0 | MCLK output control** | | 00 MCLK off (logic 0) | | 01 Output 256 x 48 kHz | | 10 Output 256 x 44.1 kHz | | 11 Output 256 x 32 kHz |
| 7 | RF transmitter enable | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | Audio transmitter power down | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | Enable wakeup on changing DD[1] | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | REQ pin polarity (0 for active low) | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | Enable direct data from pins DD[2:0]. DD[2] is only available if SSEL=1 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | S/PDIF enable (default input is I2S) | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1:0 | MCLK output control** | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 00 MCLK off (logic 0) | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 01 Output 256 x 48 kHz | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 10 Output 256 x 44.1 kHz | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 11 Output 256 x 32 kHz | | | | | | | | | | | | | | | | | | | | | | | | | |



| | | | | | | |
|------|--------------------------|--------------------------|------|---|---|---|
| 0x54 | I2SCNF_IN | R/W | 0x00 | I2S interface configuration (on ATX side), see also Table 7-2 | | |
| | | | | 7 | I2S audio in clock mode | |
| | | | | | 0 | Slave mode, WS,CLK,DATA are input |
| | | | | | 1 | Master mode, WS,CLK are output, DATA is input |
| | | | | 6 | Reserved, MBZ | |
| | | | | 5:4 | Sample length | |
| | | | | | 00 | 16-bit samples |
| | | | | | 10 | 24-bit samples |
| | | | | | other | Reserved |
| | | | | 3 | WS Polarity | |
| | | | | | 0 | WS=0: Left sample (std) |
| | | | | | 1 | WS=1: Left sample |
| | | | | 2 | Data to Bit Clock relation (data valid at clock edge) | |
| | | | | | 0 | Rising Edge (standard) |
| 1 | Reserved | | | | | |
| 1 | WS to MSB delay | | | | | |
| | 0 | 1 clock cycle (standard) | | | | |
| | 1 | 0 clock cycles | | | | |
| 0 | Audio word justification | | | | | |
| | 0 | Left justified | | | | |
| | 1 | Right justified | | | | |

Table 7-6 ATX audio input control

New values in TXMOD and I2SCNF_IN take effect after ATX and ARX are reconfigured, this is done by setting LNKMOD[4] to “1”.

****IMPORTANT NOTE!**

For S/PDIF audio input, MCLK output is disabled, but the MCLK control value TXMOD[1:0] should generally be set to the expected sampling rate. This is mandatory if 32 kHz sampling rate is expected, and recommended otherwise. Setting MCLK to the expected sampling rate gives the best phase margin and hence sampling quality when the input S/PDIF signal has much jitter.

To pace I2S data a REQ output signal is used. Polarity is set in TXMOD[4]. In I2S data mode (I2SCNF_IN[7]=1) I2S DATA and CLK input must be stopped within 32 CLK cycles when REQ goes inactive, and must start again when REQ goes active.

For analog audio sources, the nRF24Z1 offers a 256x clock output on pin MCLK (TXMOD[1:0]). This clock can be used as system clock for an external stereo ADC.



nRF24Z1 wireless audio streamer

If I2S is used for data transfers ($I2SCNF_IN[7] = 1$), average data rate is set in register I2SRAT.

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | |
|-------------|------------------|-----|-------------|---|---------|----------------|---|----------|---|-----------|-------|------------------|-----|------------|------|---------|
| 0x55 | I2SRAT | R/W | 0x00 | I2S interface speed for digital input streams that are not interpreted as audio by nRF24Z1. <table border="1"> <thead> <tr> <th>n (dec)</th> <th>Digital stream</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>8 kbit/s</td> </tr> <tr> <td>1</td> <td>16 kbit/s</td> </tr> <tr> <td>2-190</td> <td>$(n+1)*8$ kbit/s</td> </tr> <tr> <td>191</td> <td>1536kbit/s</td> </tr> <tr> <td>>191</td> <td>Illegal</td> </tr> </tbody> </table> | n (dec) | Digital stream | 0 | 8 kbit/s | 1 | 16 kbit/s | 2-190 | $(n+1)*8$ kbit/s | 191 | 1536kbit/s | >191 | Illegal |
| n (dec) | Digital stream | | | | | | | | | | | | | | | |
| 0 | 8 kbit/s | | | | | | | | | | | | | | | |
| 1 | 16 kbit/s | | | | | | | | | | | | | | | |
| 2-190 | $(n+1)*8$ kbit/s | | | | | | | | | | | | | | | |
| 191 | 1536kbit/s | | | | | | | | | | | | | | | |
| >191 | Illegal | | | | | | | | | | | | | | | |

Table 7-7 ATX I2S data transfer control

7.5.2 ARX audio interface control

In ARX mode the audio interfaces are controlled by the registers 0x4A and 0x44 listed below.

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | | | | | |
|-------------|---------------------|-----|-------------|--|-----|----------------|---|---------------------|---|---------------|---|--------------------|---|------------------|---|---------------|---|---------------|-----|---------------|
| 0x4A | RXMOD | R/W | 0x00 | Set operation modes for audio receiver <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>Receiver power down</td> </tr> <tr> <td>6</td> <td>Reserved, MBZ</td> </tr> <tr> <td>5</td> <td>RF receiver enable</td> </tr> <tr> <td>4</td> <td>REQ pin polarity</td> </tr> <tr> <td>3</td> <td>Reserved, MBZ</td> </tr> <tr> <td>2</td> <td>S/PDIF enable</td> </tr> <tr> <td>1:0</td> <td>Reserved, MBZ</td> </tr> </tbody> </table> | Bit | Interpretation | 7 | Receiver power down | 6 | Reserved, MBZ | 5 | RF receiver enable | 4 | REQ pin polarity | 3 | Reserved, MBZ | 2 | S/PDIF enable | 1:0 | Reserved, MBZ |
| Bit | Interpretation | | | | | | | | | | | | | | | | | | | |
| 7 | Receiver power down | | | | | | | | | | | | | | | | | | | |
| 6 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | |
| 5 | RF receiver enable | | | | | | | | | | | | | | | | | | | |
| 4 | REQ pin polarity | | | | | | | | | | | | | | | | | | | |
| 3 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | |
| 2 | S/PDIF enable | | | | | | | | | | | | | | | | | | | |
| 1:0 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | |



| Address Hex | Register | R/W | Initial Hex | Description | |
|-------------|--------------------------|-----|-------------|--|---|
| 0x44 | I2SCNF_OUT | R/W | 0x00 | I2S interface configuration for audio output (on ARX side), see also Table 7-2 | |
| | | | | Bit | Interpretation |
| | | | | 7 | Reserved, MBZ |
| | | | | 6 | Mute sound output |
| | | | | 5:4 | Reserved MBZ |
| | | | | 3 | WS Polarity |
| | | | | 0 | WS=0: Left sample (std) |
| | | | | 1 | WS=1: Left sample |
| | | | | 2 | Data to Bit Clock relation (data valid at clock edge) |
| | | | | 0 | Rising edge (standard) |
| | | | | 1 | Falling edge |
| | | | | 1 | WS to MSB delay |
| 0 | 1 clock cycle (standard) | | | | |
| 1 | 0 clock cycles | | | | |
| 0 | Audio word justification | | | | |
| 0 | Left justified | | | | |
| 1 | Right justified | | | | |

Table 7-8 ARX audio interface control registers

RXMOD is automatically sent from ARX to ATX during link-locate mode. New values in I2SCNF_OUT take effect after ATX and ARX are reconfigured, this is done by setting LNKMOD[4] to “1”.

In I2S data mode (I2SCNF_IN[7]=1) I2S DATA and CLK out will be stopped within 32 CLK cycles when REQ input goes inactive, and will start again when REQ goes active.

NOTE:

After reset, the ARX registers can be accessed by the audio transmitter (nRF24Z1 in ATX mode) through the control and data channel set up between two linked nRF24Z1.



7.5.3 I2S Audio interface timing

I2S input (ATX) timing

The I2S input protocol is configurable to handle several I2S formats. In addition, the interface will automatically detect sample size and word length for the most common formats. This section shows the detailed bit, clock and word timing requirements.

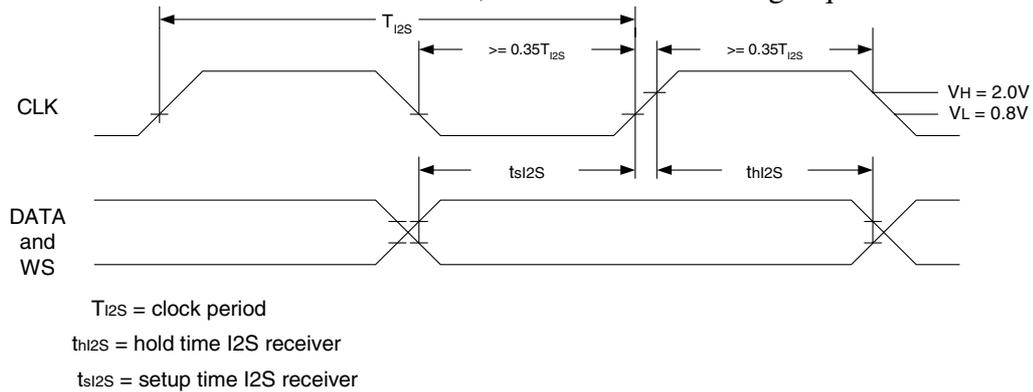


Figure 7-2. Timing for I2S input on nRF24Z1, for values see Table 14-1

I2S output (ARX) timing

I2S output is protocol compatible with most I2S DACs and CODECs.

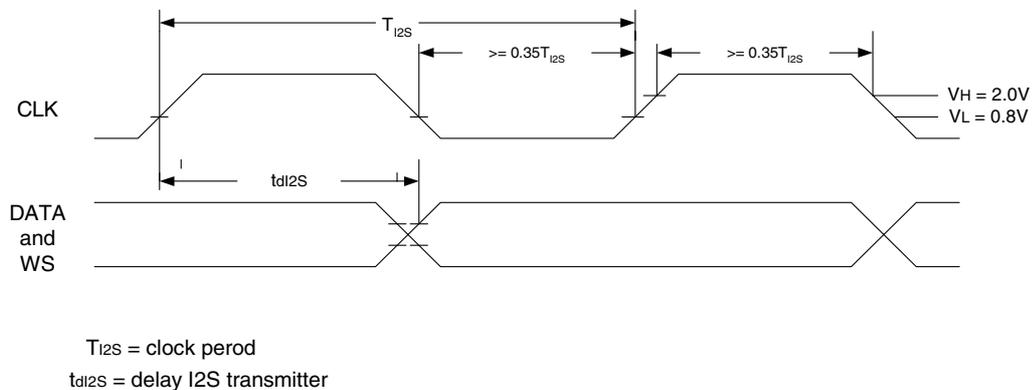


Figure 7-3. Timing for I2S output of nRF24Z1, for values see Table 14-1

7.6 Serial master interfaces

The nRF24Z1 serial master interfaces group contains an SPI and a 2-wire master interface available at all times.

After RESET the nRF24Z1 will look for a serial memory on the SPI master interface. If no memory is present, the process is repeated on the 2-wire interface. If no memories are found the nRF24Z1 assumes an external microcontroller will configure it over one of the


nRF24Z1 wireless audio streamer

slave control interfaces and will hence be idle until that happens. Link initialization will start as soon as the necessary control register bits are set.

During the configuration the SPI master is operated at 1MHz. SPI format is CPOL=0, CHPA=0 as used by industry standard EEPROM/FLASH memories.

The nRF24Z1 is protocol compatible with SPI memory with sizes ranging from 1 Kbyte to 64 Kbytes with 16-bit sub-address used.

The slave(s) that are connected determines the protocol on the 2-wire master interface. For the case of configuration, nRF24Z1 is protocol compatible with “industry standard” 2-wire memory with sizes ranging from 128 bytes to 4 Kbytes (with 3 address pins and one byte sub-address used). During configuration this interface is operated at 100 kHz for compatibility with most serial 2-wire memories.

The pin out and functionality of the serial master I/O pins are shown in Table 7-9.

| nRF24Z1 I/O: serial masters | | ARX mode | | | | ATX mode | | | |
|--------------------------------------|-------|-------------------------|-----------|-------------------------|-----------|--|-----------|--|-----------|
| | | MODE=0 RXDCMD[7] = 0 | | MODE=0 RXDCMD[7] = 1 | | MODE=1 and EEPROM connected to SPI | | MODE =1 and EEPROM connected to 2 wire | |
| Pin | Name | Function | Direction | Function | Direction | Function | Direction | Function | Direction |
| | | | | X | IN | X | IN | | |
| 27 | MSDA | MSDA | IN/OUT | RESET* | OUT | RESET* | OUT | MSDA | IN/OUT |
| 28 | MSCL | MSCL | OUT | X | IN | X | IN | MSCL | OUT |
| 29 | MSCN | X | IN | MSCN | OUT | MSCN | OUT | X | IN |
| 30 | MMISO | X | IN | MMISO | IN | MMISO | IN | X | IN |
| | | X | IN | | | | | X | IN |
| 31 | MMOSI | RESET* | OUT | MMOSI | OUT | MMOSI | OUT | RESET* | OUT |
| 35 | MSCK | X | IN | MSCK | OUT | MSCK | OUT | X | IN |

Table 7-9 Serial masters functionality

* A pin in the serial interface that is NOT used for external memory and/or controlling external circuitry can be configured to act as reset for external devices such as an ADC or DAC. Please refer to chapter 10 for further details.

After configuration, the master interfaces in ATX goes idle with pins still active, while the interfaces in ARX becomes an extended arm of a linked ATX through the control channel between two nRF24Z1.

The set-up of the serial master interfaces of the ARX after configuration is loaded is controlled by register RXDCMD (0x70) as shown in Table 7-10.



| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|--|-----------------------|--------------------|---|------------------------------|----------------|--------------------------------|---|--------------------------------|----------------|--------------------------------|--------------------------|------------------------------|-----------------------|---------------------------------|--|---------------------------------|--------------------|---------------------------------|------------------------------|-----------------------|--------------------------------|-------------------|--------------------------------|---|--------------------------------|----------------------------|------------------------------|-----|---------------------------------|-----|---------------------------------|-----|---------------------------------|-------|-----------------------|-----|-------------------|-----|---|-----|---|-------------|--------------------|-----------------------|-----|----------|---------|-----|----------|------------|-----|----------|------------|-----|----------|----------|-----|----------|---------|-----|------------|---------|-----|------------|---------|-----|----------------------------|--|---|---------------|--|--|
| 0x70 | RXDCMD | R/W | 0x82 | <p>Data “command”. Specifies master interface and speed</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>Interface select <table border="1"> <thead> <tr> <th>Value</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Use ARX 2-wire interface</td> </tr> <tr> <td>1</td> <td>Use ARX SPI interface</td> </tr> </tbody> </table> </td> </tr> <tr> <td>6:4</td> <td>SPI slave select control. Set which signal to be used as slave select, and associated polarity. Or 2-wire access type. <table border="1"> <thead> <tr> <th>Value</th> <th>SPI interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>CSN, active low²</td> </tr> <tr> <td>001</td> <td>DO[0], active low¹</td> </tr> <tr> <td>010</td> <td>DO[1], active low¹</td> </tr> <tr> <td>011</td> <td>DO[2], active low¹</td> </tr> <tr> <td>100</td> <td>CSN, active low²</td> </tr> <tr> <td>101</td> <td>DO[0], active high¹</td> </tr> <tr> <td>110</td> <td>DO[1], active high¹</td> </tr> <tr> <td>111</td> <td>DO[2], active high¹</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Value</th> <th>2-wire interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>start stop access</td> </tr> <tr> <td>001</td> <td>start only access (a start only access followed by a start-stop access will be a start-start-stop access)</td> </tr> </tbody> </table> </td> </tr> <tr> <td>3:1</td> <td>Speed select <table border="1"> <thead> <tr> <th>Value (bin)</th> <th>SPI Interpretation</th> <th>2-wire Interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>8 Mbit/s</td> <td>Illegal</td> </tr> <tr> <td>001</td> <td>8 Mbit/s</td> <td>100 kbit/s</td> </tr> <tr> <td>010</td> <td>4 Mbit/s</td> <td>400 kbit/s</td> </tr> <tr> <td>011</td> <td>2 Mbit/s</td> <td>1 Mbit/s</td> </tr> <tr> <td>100</td> <td>1 Mbit/s</td> <td>Illegal</td> </tr> <tr> <td>101</td> <td>500 kbit/s</td> <td>Illegal</td> </tr> <tr> <td>110</td> <td>250 kbit/s</td> <td>Illegal</td> </tr> <tr> <td>111</td> <td colspan="2">No master interface access</td> </tr> </tbody> </table> </td> </tr> <tr> <td>0</td> <td colspan="3">Reserved, MBZ</td> </tr> </tbody> </table> <p>Notes:</p> <ol style="list-style-type: none"> In order to use any of DO[2:0] as SPI slave selects, the corresponding bit in RXPIO should be set to the SPI slave select inactive state. I.e. if DO[0] is to be used as active low slave select, RXPIO[0] must be set to 1. MCSN is always active low | Bit | Interpretation | 7 | Interface select <table border="1"> <thead> <tr> <th>Value</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Use ARX 2-wire interface</td> </tr> <tr> <td>1</td> <td>Use ARX SPI interface</td> </tr> </tbody> </table> | Value | Interpretation | 0 | Use ARX 2-wire interface | 1 | Use ARX SPI interface | 6:4 | SPI slave select control. Set which signal to be used as slave select, and associated polarity. Or 2-wire access type. <table border="1"> <thead> <tr> <th>Value</th> <th>SPI interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>CSN, active low²</td> </tr> <tr> <td>001</td> <td>DO[0], active low¹</td> </tr> <tr> <td>010</td> <td>DO[1], active low¹</td> </tr> <tr> <td>011</td> <td>DO[2], active low¹</td> </tr> <tr> <td>100</td> <td>CSN, active low²</td> </tr> <tr> <td>101</td> <td>DO[0], active high¹</td> </tr> <tr> <td>110</td> <td>DO[1], active high¹</td> </tr> <tr> <td>111</td> <td>DO[2], active high¹</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Value</th> <th>2-wire interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>start stop access</td> </tr> <tr> <td>001</td> <td>start only access (a start only access followed by a start-stop access will be a start-start-stop access)</td> </tr> </tbody> </table> | Value | SPI interpretation | 000 | CSN, active low ² | 001 | DO[0], active low ¹ | 010 | DO[1], active low ¹ | 011 | DO[2], active low ¹ | 100 | CSN, active low ² | 101 | DO[0], active high ¹ | 110 | DO[1], active high ¹ | 111 | DO[2], active high ¹ | Value | 2-wire interpretation | 000 | start stop access | 001 | start only access (a start only access followed by a start-stop access will be a start-start-stop access) | 3:1 | Speed select <table border="1"> <thead> <tr> <th>Value (bin)</th> <th>SPI Interpretation</th> <th>2-wire Interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>8 Mbit/s</td> <td>Illegal</td> </tr> <tr> <td>001</td> <td>8 Mbit/s</td> <td>100 kbit/s</td> </tr> <tr> <td>010</td> <td>4 Mbit/s</td> <td>400 kbit/s</td> </tr> <tr> <td>011</td> <td>2 Mbit/s</td> <td>1 Mbit/s</td> </tr> <tr> <td>100</td> <td>1 Mbit/s</td> <td>Illegal</td> </tr> <tr> <td>101</td> <td>500 kbit/s</td> <td>Illegal</td> </tr> <tr> <td>110</td> <td>250 kbit/s</td> <td>Illegal</td> </tr> <tr> <td>111</td> <td colspan="2">No master interface access</td> </tr> </tbody> </table> | Value (bin) | SPI Interpretation | 2-wire Interpretation | 000 | 8 Mbit/s | Illegal | 001 | 8 Mbit/s | 100 kbit/s | 010 | 4 Mbit/s | 400 kbit/s | 011 | 2 Mbit/s | 1 Mbit/s | 100 | 1 Mbit/s | Illegal | 101 | 500 kbit/s | Illegal | 110 | 250 kbit/s | Illegal | 111 | No master interface access | | 0 | Reserved, MBZ | | |
| Bit | Interpretation | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | Interface select <table border="1"> <thead> <tr> <th>Value</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Use ARX 2-wire interface</td> </tr> <tr> <td>1</td> <td>Use ARX SPI interface</td> </tr> </tbody> </table> | Value | Interpretation | 0 | Use ARX 2-wire interface | 1 | Use ARX SPI interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | Interpretation | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | Use ARX 2-wire interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | Use ARX SPI interface | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6:4 | SPI slave select control. Set which signal to be used as slave select, and associated polarity. Or 2-wire access type. <table border="1"> <thead> <tr> <th>Value</th> <th>SPI interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>CSN, active low²</td> </tr> <tr> <td>001</td> <td>DO[0], active low¹</td> </tr> <tr> <td>010</td> <td>DO[1], active low¹</td> </tr> <tr> <td>011</td> <td>DO[2], active low¹</td> </tr> <tr> <td>100</td> <td>CSN, active low²</td> </tr> <tr> <td>101</td> <td>DO[0], active high¹</td> </tr> <tr> <td>110</td> <td>DO[1], active high¹</td> </tr> <tr> <td>111</td> <td>DO[2], active high¹</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Value</th> <th>2-wire interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>start stop access</td> </tr> <tr> <td>001</td> <td>start only access (a start only access followed by a start-stop access will be a start-start-stop access)</td> </tr> </tbody> </table> | Value | SPI interpretation | 000 | CSN, active low ² | 001 | DO[0], active low ¹ | 010 | DO[1], active low ¹ | 011 | DO[2], active low ¹ | 100 | CSN, active low ² | 101 | DO[0], active high ¹ | 110 | DO[1], active high ¹ | 111 | DO[2], active high ¹ | Value | 2-wire interpretation | 000 | start stop access | 001 | start only access (a start only access followed by a start-stop access will be a start-start-stop access) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | SPI interpretation | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 000 | CSN, active low ² | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 001 | DO[0], active low ¹ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 010 | DO[1], active low ¹ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 011 | DO[2], active low ¹ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 100 | CSN, active low ² | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 101 | DO[0], active high ¹ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110 | DO[1], active high ¹ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 111 | DO[2], active high ¹ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | 2-wire interpretation | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 000 | start stop access | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 001 | start only access (a start only access followed by a start-stop access will be a start-start-stop access) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3:1 | Speed select <table border="1"> <thead> <tr> <th>Value (bin)</th> <th>SPI Interpretation</th> <th>2-wire Interpretation</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>8 Mbit/s</td> <td>Illegal</td> </tr> <tr> <td>001</td> <td>8 Mbit/s</td> <td>100 kbit/s</td> </tr> <tr> <td>010</td> <td>4 Mbit/s</td> <td>400 kbit/s</td> </tr> <tr> <td>011</td> <td>2 Mbit/s</td> <td>1 Mbit/s</td> </tr> <tr> <td>100</td> <td>1 Mbit/s</td> <td>Illegal</td> </tr> <tr> <td>101</td> <td>500 kbit/s</td> <td>Illegal</td> </tr> <tr> <td>110</td> <td>250 kbit/s</td> <td>Illegal</td> </tr> <tr> <td>111</td> <td colspan="2">No master interface access</td> </tr> </tbody> </table> | Value (bin) | SPI Interpretation | 2-wire Interpretation | 000 | 8 Mbit/s | Illegal | 001 | 8 Mbit/s | 100 kbit/s | 010 | 4 Mbit/s | 400 kbit/s | 011 | 2 Mbit/s | 1 Mbit/s | 100 | 1 Mbit/s | Illegal | 101 | 500 kbit/s | Illegal | 110 | 250 kbit/s | Illegal | 111 | No master interface access | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value (bin) | SPI Interpretation | 2-wire Interpretation | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 000 | 8 Mbit/s | Illegal | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 001 | 8 Mbit/s | 100 kbit/s | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 010 | 4 Mbit/s | 400 kbit/s | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 011 | 2 Mbit/s | 1 Mbit/s | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 100 | 1 Mbit/s | Illegal | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 101 | 500 kbit/s | Illegal | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110 | 250 kbit/s | Illegal | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 111 | No master interface access | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Table 7-10 RXDCMD register



Values in RXDCMD are used when 0x01 is written to RXEXEC.

Number of bytes data to read from/write to master interface is set in ‘write and read count’ registers RXWCNT (0x71) and RXRCNT (0x72) and the actual data are transferred through the data buffers RXBUF. Once the above registers are set, writing to RXEXEC will initiate a SPI or 2-wire operation on the ARX serial master interfaces. Operation finished is also reported in RXEXEC and can be coupled to an interrupt in the ATX. See also Figure 5-2 nRF24Z1 control and data link. Values in RXWCNT, RXRCNT, and RXBUF are used when 0x01 is written to RXEXEC. If no data is to be exchanged on the master interface, configure the “Speed select” In RXDCMD to “No master interface access”.

| Address Hex | Register | R/W | Initial Hex | Description |
|-------------|----------|-----|-------------|---|
| 0x71 | RXWCNT | R/W | 0x00 | Number of bytes to write (max 16) to master interface |
| 0x72 | RXRCNT | R/W | 0x00 | Number of bytes to read (max 16) from master interface |
| 0x60-0x6f | RXBUF | R/W | 0x00 | Data to be written to the interface specified by RXDCMD, or Data read from audio receiver on the interface specified by RXDCMD |
| 0x74 | RXEXEC | R/W | 0x00 | Write to this register will execute a command on the audio receiver. The interface and speed are specified by RXDCMD. The audio receiver will first write RXWCNT bytes from RXBUF to the selected interface, then read RXRCNT bytes and transmit back to be stored in RXBUF. An interrupt may be delivered upon successful completion of the command. Returned values are : 0 : idle, and last transfer was successful 1 : busy with a write or read command 2 : idle, but last transfer was unsuccessful MCU must set RXEXEC=1 to perform command, and can thereafter poll RXEXEC to see if command is finished (idle) |

Table 7-11 ARX master data registers



7.6.1 Timing serial master interfaces

SPI master timing

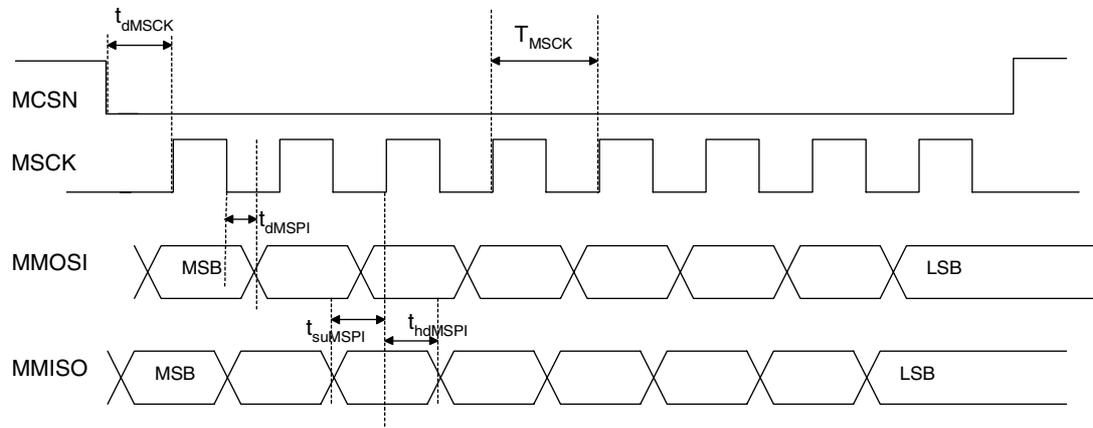


Figure 7-4 : SPI master interface timing, here is shown a one byte transaction.

T_{MSCK} : MSCK cycle time, as defined by RXDCMD register.

t_{dMSCK} : time from MCSN active to first SCK pulse, $t_{dSCK} = T_{MSCK} / 2$

t_{dMSPI} : delay from negedge MSCK to new MMOSI output data

t_{suMSPI} : MMISO setup time to posedge MSCK.

t_{hdMSPI} : MMISO hold time to posedge MSCK.

for values see Table 14-1

2-wire master timing

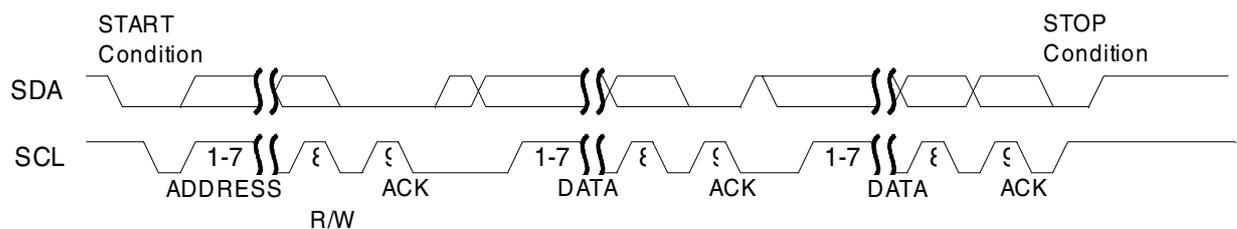


Figure 7-5 : 2-wire data transfer

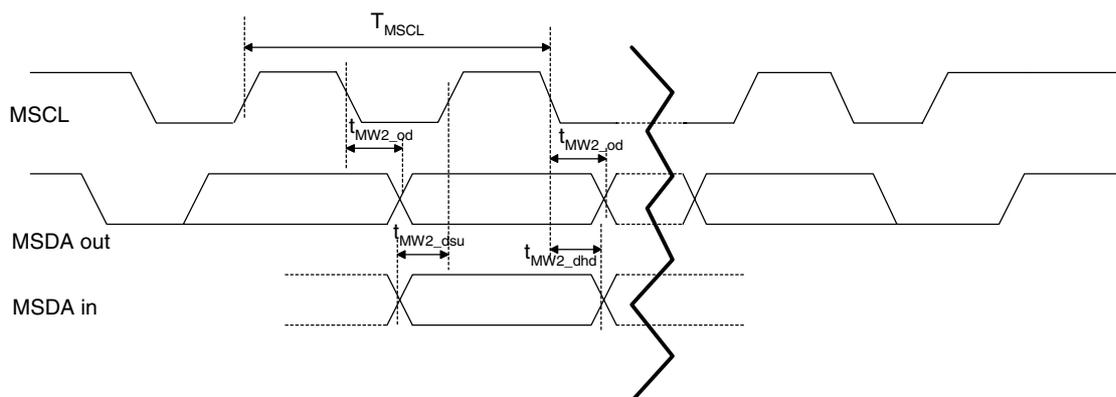


Figure 7-6 : 2-wire master timing, for values see Table 14-1

7.7 Control and GPIO interfaces

The following control and GPIO interfaces are found:

- ATX: SPI or 2-wire slave interfaces and general purpose inputs(DD [2:0])
- ARX: SPI or 2-wire slave interfaces as for ATX, or GPIO pins DI[3:0], DO[3:0] with alternative functionality: PWM or master SPI chip select signals

The ATX serial slave interfaces are the main channel for external applications or devices to control the nRF24Z1 and the RF communication. All register access and all but the most basic control and data transfers' takes place through one of these interfaces.

If an external controlling device (microcontroller) is not present in the audio transmitter (ATX), nRF24Z1 configuration can only be done once directly after RESET (section 7.6). All audio and RF link set-up is fixed and the data/control channel offered can only carry simple button push/interrupt signals.

7.7.1 ATX control and GPIO pins

In ATX mode the pins are mainly carrying the serial slave control interfaces that enables access and control of all the nRF24Z1 registers both in the device physically connected and a linked ARX (through the nRF24Z1 data/control RF channel).

One of two interfaces can be chosen based on the level of input pin SSEL.

SSEL = 0; SPI (pins SCSN, SSCK, SMISO, SMOSI).

SSEL = 1; 2-wire (pins SADR, SSCL and SSDA)

The functionality and signal direction of the pins in ATX mode are listed in the table below.



| nRF24Z1 GPIO and serial slave interface pins | | ATX mode | | | |
|--|-------------|--|-----------|---|-----------|
| | | Device control: SPI MODE, SSEL = 10 | | Device control: 2-wire MODE, SSEL =11 | |
| pin | name | Function | Direction | Function | Direction |
| 1 | SSEL | SSEL | IN | SSEL | IN |
| 2 | SMISO/SSDA | SMISO | OUT | SSDA | IN/OUT |
| 3 | SSCK/SSCL | SSCK | IN | SSCL | IN/OUT |
| 4 | SCSN/SADR | SCSN | IN | SADR | IN |
| | | | | | |
| 6 | SMOSI/DD[2] | SMOSI | IN | DD[2] | IN |
| 7 | DD[1] | DD[1] | IN | DD[1] | IN |
| 8 | DD[0] | DD[0] | IN | DD[0] | IN |
| | | | | | |
| 26 | MODE | MODE | IN | MODE | IN |
| 36 | IRQ | IRQ | OUT | IRQ | OUT |

Table 7-12 ATX Control and GPIO pins functionality

7.7.2 SPI slave interface

The first byte of the SPI transaction is a special command which specifies the register address and whether it is a read or a write access. The seven least significant bits in the first byte is the nRF24Z1 register address, while the most significant bit is the read/write indicator (read=1, write=0), see Table 7-13

| B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|-----|------------------|----|----|----|----|----|----|
| R/W | Register address | | | | | | |

Table 7-13 SPI command byte encoding

Write transaction: The next byte on SMOSI will be put into register with address as set in the first byte. Writing more bytes will auto increment the register address.

Read transaction: The next byte on SMISO will be value of register with address as set in first byte. Reading more bytes will auto increment the register address.

Consecutive accesses with SCSN low will auto-increment the address. This means that all registers may theoretically be accessed with one SPI transfer.

7.7.3 2-wire slave interface

This interface is similar to what is found on serial memories and data converter devices. The 7-bit device address of nRF24Z1 is 'a101001', where 'a' is the logic level of the SADR input pin (read during power up and reset only).

Each 2-wire transaction is started with the "Start condition" followed by the first byte which contains the 7 bit long device address and one read/write bit, this byte is hereafter referred to as the "address/read command byte" or the "address/write command byte"



depending on the state of the read/write bit, read=1, write=0. The second byte contains the register address which specifies which nRF24Z1 register is to be accessed. This address will be written into the ATX chip, it is therefore necessary that the first byte after the first start condition is an address/write command. Further actions on the 2-wire interface depend on whether the access is a read or write access. W2 command byte is illustrated in Table 7-14.

| B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|----|----|----|----|----|----|----|-----|
| a | 1 | 0 | 1 | 0 | 0 | 1 | R/W |

Table 7-14 : 2-wire command byte encoding

7.7.3.1 2-Wire write access.

Figure 7-7 illustrates a simple write operation, where one byte is written to the ATX chip.

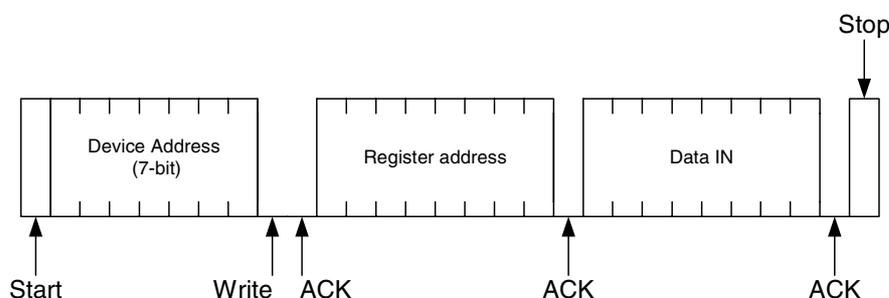


Figure 7-7 : example of a 2-wire write operation

A write access is composed by a start condition, an address/write command byte, a register address byte and a data byte which will be written to the register specified in the previous register address byte. Each byte will be acknowledged by the 2-wire slave by pulling the data line (SDA) low. To stop the write access a stop condition should be applied on the 2-wire interface. See Figure 7-9 for an example. Consecutive write access is performed by postponing the stop condition.

7.7.3.2 2-Wire read access

Figure 7-8 illustrates a simple read operation, where one byte is read back from the ATX chip.

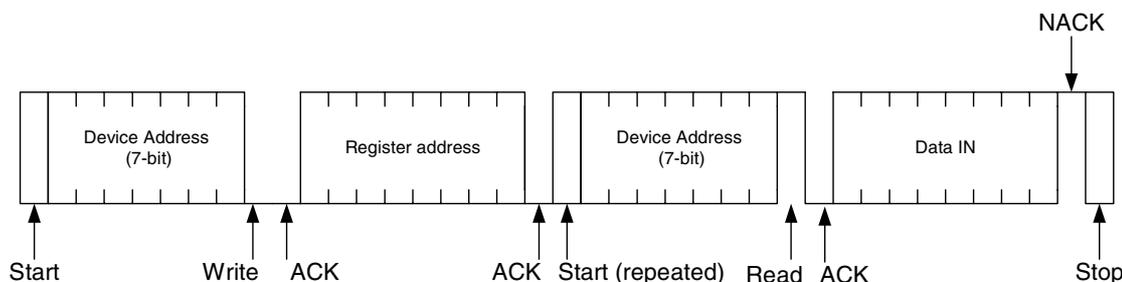


Figure 7-8 : example of a 2-wire read operation


nRF24Z1 wireless audio streamer

A read access is composed by a start condition, an address/write command byte, and a register address byte. These two bytes are acknowledged by the 2-wire slave. This scenario is followed by a repeated start condition and an address/read control byte. This byte is also acknowledged by the 2-wire slave. After the acknowledge bit has been sent from the 2-wire slave, the register value which corresponds to the register address byte will be supplied by the 2-wire slave as well. This byte should be acknowledged by the 2-wire master if consecutive register read is wanted. The read access is stopped by not acknowledging the last byte read, followed by a stop condition. See Figure 7-9 for an example.

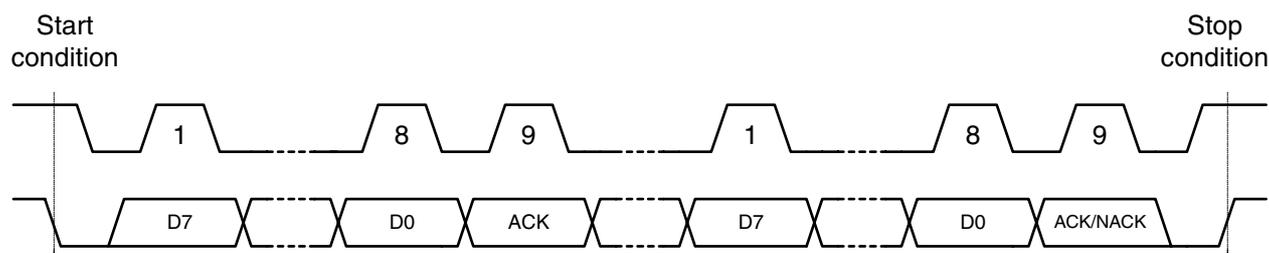


Figure 7-9 : example of 2-wire waveform

7.7.4 General purpose input pins D[2:0]

Three (2 if SPI slave is used, SSEL=0) general purpose inputs are also available. The status on these pins can be read in register 0x20

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | |
|-------------|-------------------------------|-----|-------------|--|-----|----------------|-----|----------------------|---|-------------------------------|---|--------------|---|--------------|
| 0x75 | TXDD | R | 0x00 | Value of ATX DD input pins | | | | | | | | | | |
| | | | | <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7:3</td> <td>Reserved, do not use</td> </tr> <tr> <td>2</td> <td>Value of DD2 (only if SSEL=1)</td> </tr> <tr> <td>1</td> <td>Value of DD1</td> </tr> <tr> <td>0</td> <td>Value of DD0</td> </tr> </tbody> </table> | Bit | Interpretation | 7:3 | Reserved, do not use | 2 | Value of DD2 (only if SSEL=1) | 1 | Value of DD1 | 0 | Value of DD0 |
| Bit | Interpretation | | | | | | | | | | | | | |
| 7:3 | Reserved, do not use | | | | | | | | | | | | | |
| 2 | Value of DD2 (only if SSEL=1) | | | | | | | | | | | | | |
| 1 | Value of DD1 | | | | | | | | | | | | | |
| 0 | Value of DD0 | | | | | | | | | | | | | |

Table 7-15 ATX DD[2:0] status

If bit TXMOD[3] is set the levels of pins DD[2:0] are mirrored on pins DO[2:0] on a linked ARX device directly. See Table 7-6

The IRQ pin can act as an interrupt signal to the outside application. There are a number of interrupt sources available; this is described further in chapter 9.

7.7.5 ATX Control interface timing

7.7.5.1 SPI slave timing

The SPI slave can operate with up to 8 MHz clock speed over the full operation range of the device. With a 3V +/-10% supply the maximum clock speed is 16MHz. However note that there is a minimum pause interval t_{SRD} between writing or reading a byte.

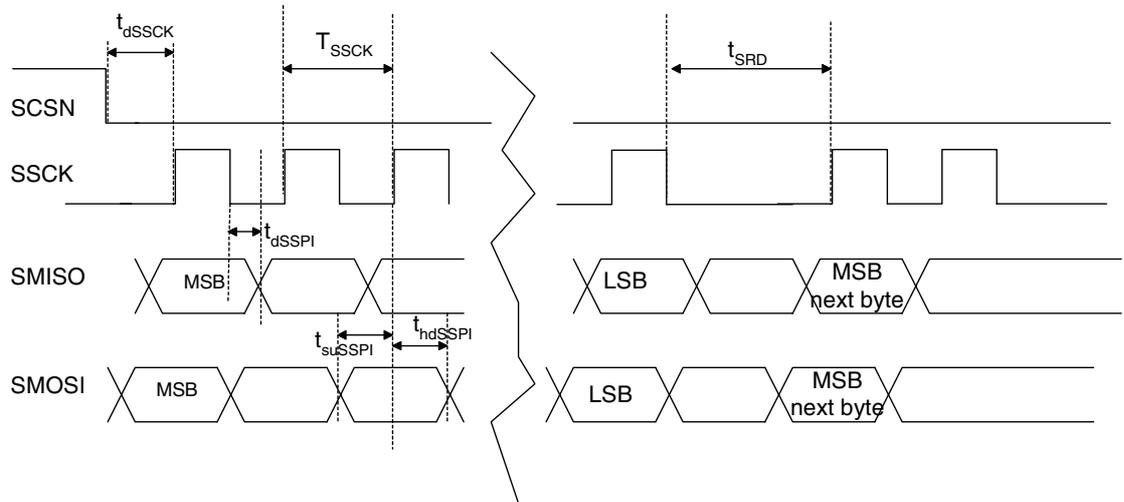


Figure 7-10 : SPI slave timing diagram

T_{SSCK} : SSCK cycle time.

t_{dSSCK} : time from SCSN active to first SSCK pulse

t_{dSSPI} : delay from negedge SSCK to new SMISO output data

t_{suSSPI} : SMOSI setup time to posedge SSCK

t_{hdSSPI} : SMOSI hold time to posedge SSCK

t_{SRD} : minimum pause between each byte read from or written to slave SPI

for values see Table 14-1

7.7.5.2 2-wire slave timing

The interface supports 100 kHz, 400 kHz and 1MHz over the operating range of the device.

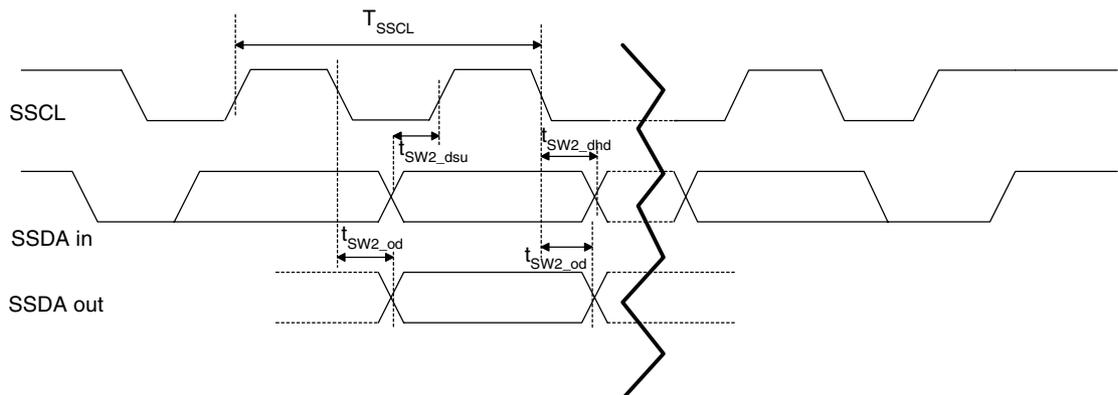


Figure 7-11 : 2-wire slave timing diagram, for values see Table 14-1

7.7.6 **ARX control interface options**

ARX is default configured with a serial slave interface, identical to ATX serial slave interface, SPI or 2-wire as selected by pin SSEL, in the same way as for ATX. The serial slave interface gives local access to all registers in ARX, but not to any registers on the



nRF24Z1 wireless audio streamer

ATX side, except for RXPIN[3:0] and indirectly RXBUF. Alternatively if RXSTA[6] is set in the configuration EEPROM, ARX will be configured with GPIO pins instead of serial slave interface. The pin out and functionality of the slave interface or GPIO pins are shown in Table 7-17

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | |
|-------------|---|-----|-------------|--|-----|----------------|---|------------------------------------|---|---|
| 0x06 | RXSTA | R | 0x00 | ARX status register. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>RXEXEC status, 1 is OK, 0 is error</td> </tr> <tr> <td>6</td> <td>0 : ARX has serial slave interface 1 : ARX has GPIO interface, which implies no serial slave interface</td> </tr> </tbody> </table> | Bit | Interpretation | 7 | RXEXEC status, 1 is OK, 0 is error | 6 | 0 : ARX has serial slave interface 1 : ARX has GPIO interface, which implies no serial slave interface |
| Bit | Interpretation | | | | | | | | | |
| 7 | RXEXEC status, 1 is OK, 0 is error | | | | | | | | | |
| 6 | 0 : ARX has serial slave interface 1 : ARX has GPIO interface, which implies no serial slave interface | | | | | | | | | |

Table 7-16: ARX status register

RXSTA[6] may not be updated from ATX. It must be set by EEPROM or MCU in ARX.

| | ARX with SPI slave interface RXSTA6 = 0, SSEL=0 | | ARX with 2-wire slave interface RXSTA6 = 0, SSEL=1 | | ARX with GPIO interface RXSTA6 =1 | |
|-----|--|-----------|---|-----------|--------------------------------------|-----------|
| Pin | Function | Direction | Function | Direction | Function | Direction |
| 1 | SSEL | IN | SSEL | IN | DO2 | OUT |
| 2 | SMISO | OUT | SSDA | IN/OUT | DO1 | OUT |
| 3 | SSCK | IN | SSCL | IN/OUT | DO0 | OUT |
| 4 | SCSN | IN | SADR | IN | DI3 | IN |
| 6 | SMOSI | IN | DI2 | IN | DI2 | IN |
| 36 | IRQ ¹ | OUT | IRQ | OUT | DO3/PWM ² | OUT |

Table 7-17: ARX serial slave interface or GPIO pins.

Note that the GPIO functions described for registers RXPIO, RXPWME, RXPWMD, RXPIN bits 3:2, RXWAKE bits 3:2, are only available if ARX is configured with GPIO interface (RXSTA[6]=1). However these registers may always be read or written to, but if ARX is configured with slave interface the registers will be disconnected from their corresponding GPIO pins.

Note also that if ARX is configured with slave interface, many ARX registers can be accessed both over the air from ATX, and locally via slave interface, and it is the sole responsibility of the external MCU to avoid setting conflicting values to a register.

¹ ATX and ARX has each its own local instance of the INTSTA register.

² General purpose output (DO[3]) or PWM functionality set in register RXPWME (0x42)



7.7.7 ARX GPIO pins

ARX mode has general purpose inputs DI[3:0] and outputs DO[3:0].

OUTPUTS

General purpose pins DO[3:0] can be controlled in several ways:

| Pins | Functionality | Controlling register | Description |
|---------|---------------------------|--------------------------------|-------------------------|
| DO[3:0] | General purpose output | RXPIO (0x41) | See below |
| DO[2:0] | Mirror of DD[2:0] | TXMOD (0x1A) | Section 7.7.1 |
| DO[2:0] | ARX SPI master bus enable | RXDCMD (0x70) | Table 7-10, Section 7.6 |
| DO[3] | PWM output | RXPWME (0x42) RXPWMD (0x43) | See below |

Table 7-18 DO[3:0] alternate functions

Used as general purpose outputs DO[3:0] are controlled by register RXPIO (0x41). On all general purpose outputs of the ARX, high current drive capabilities can be enabled to drive LED's for instance. The register contents of 0x41 is given in Table 7-19.

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | | | | | | | |
|-------------|----------------------------------|-----|-------------|--|-----|------------------------|---|----------------------------------|---|---------------------------------|---|----------------------------------|---|---------------------------------|---|----------------|---|----------------|---|----------------|---|----------------|
| 0x41 | RXPIO | R/W | 0x00 | Receiver GPIO output and drive strength <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bit</th> <th>Logic 1 Interpretation</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>High drive high enable for DO[3]</td> </tr> <tr> <td>6</td> <td>High drive low enable for DO[2]</td> </tr> <tr> <td>5</td> <td>High drive high enable for DO[1]</td> </tr> <tr> <td>4</td> <td>High drive low enable for DO[0]</td> </tr> <tr> <td>3</td> <td>Data for DO[3]</td> </tr> <tr> <td>2</td> <td>Data for DO[2]</td> </tr> <tr> <td>1</td> <td>Data for DO[1]</td> </tr> <tr> <td>0</td> <td>Data for DO[0]</td> </tr> </tbody> </table> | Bit | Logic 1 Interpretation | 7 | High drive high enable for DO[3] | 6 | High drive low enable for DO[2] | 5 | High drive high enable for DO[1] | 4 | High drive low enable for DO[0] | 3 | Data for DO[3] | 2 | Data for DO[2] | 1 | Data for DO[1] | 0 | Data for DO[0] |
| Bit | Logic 1 Interpretation | | | | | | | | | | | | | | | | | | | | | |
| 7 | High drive high enable for DO[3] | | | | | | | | | | | | | | | | | | | | | |
| 6 | High drive low enable for DO[2] | | | | | | | | | | | | | | | | | | | | | |
| 5 | High drive high enable for DO[1] | | | | | | | | | | | | | | | | | | | | | |
| 4 | High drive low enable for DO[0] | | | | | | | | | | | | | | | | | | | | | |
| 3 | Data for DO[3] | | | | | | | | | | | | | | | | | | | | | |
| 2 | Data for DO[2] | | | | | | | | | | | | | | | | | | | | | |
| 1 | Data for DO[1] | | | | | | | | | | | | | | | | | | | | | |
| 0 | Data for DO[0] | | | | | | | | | | | | | | | | | | | | | |

Table 7-19 Register 0x41 RXPIO

Values in RXPIO are sent to ARX when 0x01 is written to RXCSTATE

Pin DO[3] can also be used as a PWM output. PWM enable and PWM frequency is controlled by register RXPWME (0x42) while PWM duty cycle is controlled by register RXPWMD (0x43) as shown in Table 7-20.



| Address Hex | Register | R/W | Initial Hex | Description | | | | | | |
|-------------|--|-----|-------------|--|-----|----------------|-----|--|-----|--|
| 0x42 | RXPWME | R/W | 0x00 | Enables audio receiver PWM onto DO[3] and set PWM frequency <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7:6</td> <td>00 : PWM not enabled 11 : Enable PWM on DO[3] 01,10 : reserved, do not use</td> </tr> <tr> <td>5:0</td> <td>PWM frequency (repetition rate) $f_{PWM} = 16MHz / (255 * (1 + RXPWME[5:0]))$</td> </tr> </tbody> </table> | Bit | Interpretation | 7:6 | 00 : PWM not enabled 11 : Enable PWM on DO[3] 01,10 : reserved, do not use | 5:0 | PWM frequency (repetition rate) $f_{PWM} = 16MHz / (255 * (1 + RXPWME[5:0]))$ |
| Bit | Interpretation | | | | | | | | | |
| 7:6 | 00 : PWM not enabled 11 : Enable PWM on DO[3] 01,10 : reserved, do not use | | | | | | | | | |
| 5:0 | PWM frequency (repetition rate) $f_{PWM} = 16MHz / (255 * (1 + RXPWME[5:0]))$ | | | | | | | | | |
| 0x43 | RXPWMD | R/W | 0x00 | Set audio receiver PWM duty cycle | | | | | | |

Table 7-20 Registers RXPWME (0x42) and RXPWMD (0x43)

Values in RXPWME and RXPWMD are sent to ARX when 0x01 is written to RXCSTATE

INPUTS

DI[3:0] are in ARX general purpose inputs. Status on these pins is monitored in register RXPIN (0x07) in the ATX. If the ARX is configured with a slave interface, a connected MCU may simulate general purpose inputs by performing a local write to RXPIN

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | |
|-------------|-----------------------|-----|-------------|---|-----|----------------|-----|-----------------------|---|-----------|---|-----------|---|-----------|---|-----------|
| 0x07 | RXPIN | R | 0x00 | Current state of audio receiver GPIO inputs <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7:4</td> <td>Reserved, do not use.</td> </tr> <tr> <td>3</td> <td>DI3 value</td> </tr> <tr> <td>2</td> <td>DI2 value</td> </tr> <tr> <td>1</td> <td>DI1 value</td> </tr> <tr> <td>0</td> <td>DI0 value</td> </tr> </tbody> </table> | Bit | Interpretation | 7:4 | Reserved, do not use. | 3 | DI3 value | 2 | DI2 value | 1 | DI1 value | 0 | DI0 value |
| Bit | Interpretation | | | | | | | | | | | | | | | |
| 7:4 | Reserved, do not use. | | | | | | | | | | | | | | | |
| 3 | DI3 value | | | | | | | | | | | | | | | |
| 2 | DI2 value | | | | | | | | | | | | | | | |
| 1 | DI1 value | | | | | | | | | | | | | | | |
| 0 | DI0 value | | | | | | | | | | | | | | | |

Table 7-21 Register RXPIN (0x07)

7.8 Data Channel Timing

7.8.1 Forward data channel, data transfer from ATX to ARX

Data communication from ATX to ARX is performed through a dedicated data channel which is superimposed on the audio stream. Maximum data rate on this channel can be calculated with formula 1 and formula 2.

$$BRR = AFR / (8 * 16) \quad \text{frame/sec} \quad (1)$$

$$DR = BRR * NPB * NBP * 8 \text{bits} \quad \text{bits/sec} \quad (2)$$


nRF24Z1 wireless audio streamer

BRR: Frame Repetition Rate, time interval between start of data frames.

AFR: Audio Fundamental Rate (sampling frequency).

DR: Average Data Rate on data channel.

NPB: Number of Packets per Frame.

NBP: Number of data Bytes per Packet.

nRF24Z1 has the following parameters for the ATX to ARX data channel:

AFR: 48 kHz, 44.1 kHz or 32 kHz

NPB: 4

NBP: 1

Table 7-22 lists resulting data rates.

| Fundamental rate of audio signal | Maximum data rate on data channel |
|---|--|
| 48kHz | 12000 bits/sec |
| 44.1kHz | 11025 bits/sec |
| 32 kHz | 8000 bits/sec |

Table 7-22 : forward data channel rate

7.8.1.1 Limitations in forward data channel, effective data rate.

The data rate shown in Table 7-22 is shared by the different inputs. Data from all interfaces are fed into a FIFO queue, so that high activity on one source will increase the link delay for the other sources.

Effective data rate on forward data channel depends mainly on the link quality, a poor link can reduce data rate.

7.8.2 **Return data channel, data transfer from ARX to ATX**

Data communication from ARX to ATX is performed through a dedicated data channel which is built into the acknowledge data stream. Maximum data rate on this data channel can be calculated with Formula 1 and Formula 2 with NPB = 1 and NBP = 5, see Table 7-23.

| Fundamental rate of audio signal | Maximum data rate on return data channel |
|---|---|
| 48kHz | 15000 bits/sec |
| 44.1kHz | 13781 bits/sec |
| 32 kHz | 10000 bits/sec |

Table 7-23 : return data channel rate

7.8.2.1 Limitations in return data channel

Data transfer from ARX to ATX can be divided into three groups:

1. Link error monitoring.
2. Parallel port monitoring.
3. ARX master interface communication.


nRF24Z1 wireless audio streamer

These three groups share the same bandwidth of the return channel. As can be seen in the following sections, each function is assigned a fraction of return channel data rate shown in Table 7-23.

7.8.2.2 Link error monitoring

Refresh rate of link error register, LNKERR depends on the data rate shown in Table 7-24.

| Fundamental rate of audio signal | LNKERR data rate on return data channel |
|---|--|
| 48kHz | 3000 bits/sec |
| 44.1kHz | 2756 bits/sec |
| 32 kHz | 2000 bits/sec |

Table 7-24 :LNKERR data rate

7.8.2.3 Parallel port monitoring

The parallel port is updated according to the data rate shown in Table 7-25.

| Fundamental rate of audio signal | Parallel port data rate on return data channel |
|---|---|
| 48kHz | 1500 bits/sec |
| 44.1kHz | 1378 bits/sec |
| 32 kHz | 1000 bits/sec |

Table 7-25 : Parallel port data rate

7.8.2.4 Master interface communication

Effective data rate of master interface communication when transferring data from ARX to ATX can be seen in Table 7-26.

| Fundamental rate of audio signal | Master interface data rate on return data channel |
|---|--|
| 48kHz | 3000 bits/sec |
| 44.1kHz | 2756 bits/sec |
| 32 kHz | 2000 bits/sec |

Table 7-26 : serial master interface data rate



8 QUALITY OF SERVICE (QOS) AND RF

8.1 Link algorithm

The link algorithm of the nRF24Z1 is fully managed on-chip. Time spent searching for counterparts can be controlled by the power down counters presented in chapter 11.

8.2 RF protocol

The RF protocol of nRF24Z1 is fully controlled on-chip, the only parameter configurable by the end system is the address length and address to be used in the RF protocol. This enables numerous nRF24Z1 to be identified and accessed independently in the same physical area. The RF protocol address length is 5 bytes and the address bytes are set in registers ADDR[0:4], listed in Table 8-1.

| Address Hex | Register | R/W | Initial Hex | Description |
|-------------|----------|-----|-------------|-----------------------|
| 0x39 | ADDR[0] | R/W | 0x01 | Address byte #0 (LSB) |
| 0x3A | ADDR[1] | R/W | 0x02 | Address byte #1 |
| 0x3B | ADDR[2] | R/W | 0x03 | Address byte #2 |
| 0x3C | ADDR[3] | R/W | 0x04 | Address byte #3 |
| 0x3D | ADDR[4] | R/W | 0x05 | Address byte #4 |

Table 8-1 RF protocol address

Values in ADDR[0-4] are sent to ARX when 0x01 is written to LINKCSTATE.

8.3 Adaptive frequency hopping

nRF24Z1 features adaptive frequency hopping. This enables the nRF24Z1 link to handle RF reflections, multi-path fading and avoid heavily trafficked areas of the 2.4 GHz band. This enables stable operation both in challenging environments physically and in co-existence with other 2.4 GHz systems.

The frequency hopping is implemented in 38 channel registers. The hopping is sequentially after the contents of the frequency hopping table registers CH0, CH1, CH02.... CH36, CH37, CH0, CH1 and so on. See Table 8-2. Values in CH0-37 are not sent from ATX to ARX. They must be configured locally by EEPROM or MCU.



| Address Hex | Register | R/W | Initial Hex | Description |
|-------------|----------|-----|-------------|--|
| 0x0C | CH0 | R/W | 0x00 | Channel numbers for the frequency hopping sequence. Channel number is in 1 MHz unit relative to 2400MHz. Example: To define a frequency hopping scheme starting at f=2420MHz, and then hopping to f=2440MHz, the following values must be set : CH0=20, CH1=40. And so on for the other registers. |
| 0x0D | CH1 | R/W | 0x00 | |
| 0x0E | CH2 | R/W | 0x00 | |
| 0x0F | CH3 | R/W | 0x00 | |
| 0x10 | CH4 | R/W | 0x00 | |
| 0x11 | CH5 | R/W | 0x00 | |
| 0x12 | CH6 | R/W | 0x00 | |
| 0x13 | CH7 | R/W | 0x00 | |
| 0x14 | CH8 | R/W | 0x00 | |
| 0x15 | CH8 | R/W | 0x00 | |
| 0x16 | CH10 | R/W | 0x00 | |
| 0x17 | CH11 | R/W | 0x00 | |
| 0x18 | CH12 | R/W | 0x00 | |
| 0x19 | CH13 | R/W | 0x00 | |
| 0x1A | CH14 | R/W | 0x00 | |
| 0x1B | CH15 | R/W | 0x00 | |
| 0x1C | CH16 | R/W | 0x00 | |
| 0x1D | CH17 | R/W | 0x00 | |
| 0x1E | CH18 | R/W | 0x00 | |
| 0x1F | CH19 | R/W | 0x00 | |
| 0x20 | CH20 | R/W | 0x00 | |
| 0x21 | CH21 | R/W | 0x00 | |
| 0x22 | CH22 | R/W | 0x00 | |
| 0x23 | CH23 | R/W | 0x00 | |
| 0x24 | CH24 | R/W | 0x00 | |
| 0x25 | CH25 | R/W | 0x00 | |
| 0x26 | CH26 | R/W | 0x00 | |
| 0x27 | CH27 | R/W | 0x00 | |
| 0x28 | CH28 | R/W | 0x00 | |
| 0x29 | CH29 | R/W | 0x00 | |
| 0x2A | CH30 | R/W | 0x00 | |
| 0x2B | CH31 | R/W | 0x00 | |
| 0x2C | CH32 | R/W | 0x00 | |
| 0x2D | CH33 | R/W | 0x00 | |
| 0x2E | CH34 | R/W | 0x00 | |
| 0x2F | CH35 | R/W | 0x00 | |
| 0x30 | CH36 | R/W | 0x00 | |
| 0X31 | CH37 | R/W | 0x00 | |

Table 8-2 : Frequency hopping table registers.



8.3.1 Adapting to the RF environment

In a clean RF environment, which means few or no other RF transmitters nearby, the nRF24Z1 will use all the channels in Table 8-2. If another RF system appears in the same area, the nRF24Z1 system is likely to collide with the other system on some of the channels and lose one or more RF packets.

If the poor quality on a certain channel prevails, ATX removes it from the channels used by the frequency hopping algorithm. A synchronized frequency hopping table is kept in the ARX by use of the service channel superimposed on the audio stream. The communication problems on this channel will hence not affect the nRF24Z1 link quality thereafter.

Channels that are taken out of the channel hopping sequence are inserted into a banning list.

The length of the banning list and duration of banning are configurable, as seen in Table 8-3. Maximum number of banned channels in the banning list is 18.

With this list the nRF24Z1 can in all refrain from using 36 MHz of the 2.4 GHz band which means that it is capable of masking out an entire WLAN channel for instance.

| Address Hex | Register | R/W | Initial Hex | Description |
|-------------|----------|-----|-------------|---|
| 0x32 | BCHD | R/W | 0x01 | Banned channel duration. This says how long the channel shall be banned, in number of hops. Time (in number of frequency hops) before a banned channel is released is BCHD*NBCH |
| 0x33 | NBCH | R/W | 0x13 | Number of banned channels. The number of channels the user want to be banned at the same time. Max value for this register is 18. |
| 0x34 | NACH | R/W | 0x05 | Number of Channels used in audio mode (normal stream mode). Only the first NACH lines of the frequency table will be used. |
| 0x35 | NLCH | R/W | 0x00 | Number of Channels used in link mode. This is the number of channels the user wants to use in linking mode. Only the first NLCH lines of the frequency table will be used. |

Table 8-3 Frequency hopping configuration registers.

Values in BCHD, NBCH, NACK, and NLCH are sent to ARX when 0x01 is written to RXCSTATE

To minimise linking time between two nRF24Z1, the same basic frequency hopping scheme must be set on both ATX and ARX side.



8.4 Link registers

The QoS engine by use of the input from the registers presented above will maintain a RF link. The nRF24Z1 has a number of link registers which primary function is to monitor the overall quality on the link and enable actions to be taken based on it. The link monitoring registers are listed in Table 8-4.

| Address Hex | Register | R/W | Initial Hex | Description |
|-------------|----------|-----|-------------|--|
| 0x04 | LNKQ | R | 0x00 | Link quality register. Number of successfully transferred packets per unit of time. A value of 0 means all packets out of 256 are lost, a value of 128 means that 50% of the packets were lost and a value of 255 means that no packet out of 256 is lost. This number always represents the status over the last 256 packets transmitted. |
| 0x05 | LNKERR | R | 0x00 | Link error register. Number of audio packets permanently lost per unit of time. A packet is permanently lost if it's not successfully received at the time its required at the audio output. A permanently lost will be audible! |
| 0x37 | LNKWTH | R/W | 0x00 | Link warning threshold limit. A LNKQ value \leq LNKWTH will cause IRQ to be activated (if enabled by INTCF.5) |
| 0x38 | LNKETH | R/W | 0xff | Link error threshold limit. A LNKERR value \geq LNKETH will cause IRQ to be activated (if enabled by INTCF.2) |

Table 8-4 Link quality monitoring registers

LNKQ and LNKERR are status registers directly available in ATX. However, only LNKERR is available in ARX.

Registers LNKQ shows how hard the quality of service engine is working to maintain the link and the LNKERR shows how many times it has actually failed. LNKWTH and LNKETH enable the main system to be interrupted if one or the other of these quality indicators drops below an acceptable level.

The overall status of the link can be monitored in register LNKSTA and direct actions to be taken upon the quality requirements set in LNKWTH and LNKETH can be set in register LNKMOD. Registers LNKSTA and LNKMOD are listed in Table 8-5. LNKSTA register is only available in ATX.



| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | | | | | |
|-------------|---|-----|-------------|---|-----|----------------|-----|----------------------|---|---|---|---------------|---|---|---|-------------------------|---|------------------------------------|-----|---------------|
| 0x03 | LNKSTA | R | 0x00 | Link status register <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7:1</td> <td>Reserved, do not use</td> </tr> <tr> <td>0</td> <td>Link established</td> </tr> </tbody> </table> | Bit | Interpretation | 7:1 | Reserved, do not use | 0 | Link established | | | | | | | | | | |
| Bit | Interpretation | | | | | | | | | | | | | | | | | | | |
| 7:1 | Reserved, do not use | | | | | | | | | | | | | | | | | | | |
| 0 | Link established | | | | | | | | | | | | | | | | | | | |
| 0x36 | LNKMOD | R/W | 0x00 | Link mode register. <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>Reserved, MBZ</td> </tr> <tr> <td>6</td> <td>ATX and ARX will reset to initial register contents if no counterpart is found on next link initialization.</td> </tr> <tr> <td>5</td> <td>Reserved, MBZ</td> </tr> <tr> <td>4</td> <td>Force reconfiguration with new configuration data</td> </tr> <tr> <td>3</td> <td>Mute when LNKERR>LNKETH</td> </tr> <tr> <td>2</td> <td>Disable adaptive frequency hopping</td> </tr> <tr> <td>1:0</td> <td>Reserved, MBZ</td> </tr> </tbody> </table> | Bit | Interpretation | 7 | Reserved, MBZ | 6 | ATX and ARX will reset to initial register contents if no counterpart is found on next link initialization. | 5 | Reserved, MBZ | 4 | Force reconfiguration with new configuration data | 3 | Mute when LNKERR>LNKETH | 2 | Disable adaptive frequency hopping | 1:0 | Reserved, MBZ |
| Bit | Interpretation | | | | | | | | | | | | | | | | | | | |
| 7 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | |
| 6 | ATX and ARX will reset to initial register contents if no counterpart is found on next link initialization. | | | | | | | | | | | | | | | | | | | |
| 5 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | |
| 4 | Force reconfiguration with new configuration data | | | | | | | | | | | | | | | | | | | |
| 3 | Mute when LNKERR>LNKETH | | | | | | | | | | | | | | | | | | | |
| 2 | Disable adaptive frequency hopping | | | | | | | | | | | | | | | | | | | |
| 1:0 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | |

Table 8-5 Link registers

8.4.1 RF link latency

It is possible to trade link robustness versus link latency. In systems where latency is not important, like CD player headsets, high latency option should be used. Latency is set in TXLAT register as shown below. Note that delay depends on audio sampling rate F_s .

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | | | | | | | | | |
|-------------|----------------|----------------------|------------------------|--|-------|----------------|----------------------|------------------------|----------------------|---|---------|--------|---------|---------|---|------|-------|---------|--------|---|-------|------|------|------|
| 0x52 | TXLAT | R/W | 0x00 | ATX to ARX Latency <table border="1"> <thead> <tr> <th>Value</th> <th>Interpretation</th> <th>$F_s = 48\text{kHz}$</th> <th>$F_s = 44.1\text{kHz}$</th> <th>$F_s = 32\text{kHz}$</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>Nominal</td> <td>12.5ms</td> <td>13.4 ms</td> <td>17.8 ms</td> </tr> <tr> <td>4</td> <td>High</td> <td>20 ms</td> <td>19.2 ms</td> <td>25.8ms</td> </tr> <tr> <td>0</td> <td>Short</td> <td>7 ms</td> <td>7 ms</td> <td>8 ms</td> </tr> </tbody> </table> | Value | Interpretation | $F_s = 48\text{kHz}$ | $F_s = 44.1\text{kHz}$ | $F_s = 32\text{kHz}$ | 2 | Nominal | 12.5ms | 13.4 ms | 17.8 ms | 4 | High | 20 ms | 19.2 ms | 25.8ms | 0 | Short | 7 ms | 7 ms | 8 ms |
| Value | Interpretation | $F_s = 48\text{kHz}$ | $F_s = 44.1\text{kHz}$ | $F_s = 32\text{kHz}$ | | | | | | | | | | | | | | | | | | | | |
| 2 | Nominal | 12.5ms | 13.4 ms | 17.8 ms | | | | | | | | | | | | | | | | | | | | |
| 4 | High | 20 ms | 19.2 ms | 25.8ms | | | | | | | | | | | | | | | | | | | | |
| 0 | Short | 7 ms | 7 ms | 8 ms | | | | | | | | | | | | | | | | | | | | |

Table 8-6 Register TXLAT



8.5 RF output power

The only configurable parameter in the RF sub system is the RF output power from the device. The output power is set in registers TXPWR for ATX and RXPWR for ARX.

| Address Hex | Register | R/W | Initial Hex | Description | |
|-------------|----------|-----|-------------|--------------------------------|----------------|
| 0x56 | TXPWR | R/W | 0x03 | Audio transmitter output power | |
| | | | | Value | Interpretation |
| | | | | 0 | -20 dBm |
| | | | | 1 | -10 dBm |
| | | | | 2 | -5 dBm |
| 0x49 | RXPWR | R/W | 0x00 | Audio receiver output power | |
| | | | | Value | Interpretation |
| | | | | 0 | -20 dBm |
| | | | | 1 | -10 dBm |
| | | | | 2 | -5 dBm |
| | | | | 3 | 0 dBm |

Table 8-7 Registers TXPWR and RXPWR

Note that both the output power registers are accessible from the ATX an external system may hence tune the output power based on the result of the RF link quality registers presented above. Reducing the output power from a RF device both saves current and eases the co-existence with this device. Remember that the other RF system your system are colliding with may well be another (of your) nRF24Z1 systems.



9 INTERRUPTS

An nRF24Z1 with serial slave interface can be configured to deliver interrupts to the external system on pin IRQ. Interrupt sources are set in register INTCF (0x13) and interrupt status flags are available in register INTSTA (0x02) After an interrupt occurs, IRQ will stay active (“0” with INTCF[7]=0, “1” with INTCF[7]=”1”) until the corresponding interrupt flag in INTSTA is cleared by writing a “1” to it. All interrupt flags may be cleared by writing 0x7F to INTSTA.

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | | | | | | | |
|-------------|---|-----|-------------|--|-----|----------------|---|--------------------------------------|---|------------------------------|---|--|---|---|---|---|---|------------------------------------|---|--|---|----------------------|
| 0x02 | INTSTA | R | 0x00 | <p>Read interrupt status. Clear interrupt source bits that are read out.</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>Reserved, do not use</td> </tr> <tr> <td>6</td> <td>link broken status flag</td> </tr> <tr> <td>5</td> <td>poor link quality status flag (LNKQ<=LNKWTH)</td> </tr> <tr> <td>4</td> <td>remote transfer done status flag , which is set when a RXEXEC, TXCSTATE, LNKCSTATE or RXCSTATE command has finished</td> </tr> <tr> <td>3</td> <td>for ATX : remote input changed status flag for ARX with slave interface : DO[3:0] register value changed status flag</td> </tr> <tr> <td>2</td> <td>Link error flag (LNKERR >= LNKETH)</td> </tr> <tr> <td>1</td> <td>Wakeup from power down status flag</td> </tr> <tr> <td>0</td> <td>Reserved, do not use</td> </tr> </tbody> </table> <p>See INTCF for interrupt enabling</p> | Bit | Interpretation | 7 | Reserved, do not use | 6 | link broken status flag | 5 | poor link quality status flag (LNKQ<=LNKWTH) | 4 | remote transfer done status flag , which is set when a RXEXEC, TXCSTATE, LNKCSTATE or RXCSTATE command has finished | 3 | for ATX : remote input changed status flag for ARX with slave interface : DO[3:0] register value changed status flag | 2 | Link error flag (LNKERR >= LNKETH) | 1 | Wakeup from power down status flag | 0 | Reserved, do not use |
| Bit | Interpretation | | | | | | | | | | | | | | | | | | | | | |
| 7 | Reserved, do not use | | | | | | | | | | | | | | | | | | | | | |
| 6 | link broken status flag | | | | | | | | | | | | | | | | | | | | | |
| 5 | poor link quality status flag (LNKQ<=LNKWTH) | | | | | | | | | | | | | | | | | | | | | |
| 4 | remote transfer done status flag , which is set when a RXEXEC, TXCSTATE, LNKCSTATE or RXCSTATE command has finished | | | | | | | | | | | | | | | | | | | | | |
| 3 | for ATX : remote input changed status flag for ARX with slave interface : DO[3:0] register value changed status flag | | | | | | | | | | | | | | | | | | | | | |
| 2 | Link error flag (LNKERR >= LNKETH) | | | | | | | | | | | | | | | | | | | | | |
| 1 | Wakeup from power down status flag | | | | | | | | | | | | | | | | | | | | | |
| 0 | Reserved, do not use | | | | | | | | | | | | | | | | | | | | | |
| 0x53 | INTCF | R/W | 0x00 | <p>Interrupt configuration. Select events that can generate interrupt on the IRQ pin.</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>Interrupt polarity, 1 is active high</td> </tr> <tr> <td>6</td> <td>Enable link broken interrupt</td> </tr> <tr> <td>5</td> <td>Enable poor link quality interrupt</td> </tr> <tr> <td>4</td> <td>Enable remote transfer done interrupt</td> </tr> <tr> <td>3</td> <td>Enable remote input changed interrupt</td> </tr> <tr> <td>2</td> <td>Enable link error interrupt</td> </tr> <tr> <td>1</td> <td>Enable wakeup from power down interrupt”</td> </tr> <tr> <td>0</td> <td>Reserved, MBZ</td> </tr> </tbody> </table> | Bit | Interpretation | 7 | Interrupt polarity, 1 is active high | 6 | Enable link broken interrupt | 5 | Enable poor link quality interrupt | 4 | Enable remote transfer done interrupt | 3 | Enable remote input changed interrupt | 2 | Enable link error interrupt | 1 | Enable wakeup from power down interrupt” | 0 | Reserved, MBZ |
| Bit | Interpretation | | | | | | | | | | | | | | | | | | | | | |
| 7 | Interrupt polarity, 1 is active high | | | | | | | | | | | | | | | | | | | | | |
| 6 | Enable link broken interrupt | | | | | | | | | | | | | | | | | | | | | |
| 5 | Enable poor link quality interrupt | | | | | | | | | | | | | | | | | | | | | |
| 4 | Enable remote transfer done interrupt | | | | | | | | | | | | | | | | | | | | | |
| 3 | Enable remote input changed interrupt | | | | | | | | | | | | | | | | | | | | | |
| 2 | Enable link error interrupt | | | | | | | | | | | | | | | | | | | | | |
| 1 | Enable wakeup from power down interrupt” | | | | | | | | | | | | | | | | | | | | | |
| 0 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | | | |



Table 9-1 Registers INTCF and INTSTA

ATX always has an interrupt pin, ARX has interrupt pin only when configured with slave interface. In this case the interrupt registers are local to ATX and ARX respectively. If the nRF24Z1 is configured to give interrupt on “Wakeup from power down”, no other detected interrupts will trigger the IRQ pin, since the IRQ pin in this configuration will constantly be set active in power up and inactive in power down. This feature can be utilized by external MCU to track the power down state of the nRF24Z1. If other interrupts should trigger an event on the IRQ pin, remember to disable this feature first.

“Poor link quality interrupt”, “Remote transfer done interrupt” and “Link broken interrupt” is only available on ATX.

Values in INTSTA and INTCF are local to ATX or ARX. All interrupts must be cleared after power-on-reset.



10 RESET OUTPUTS

Some pins of the nRF24Z1 not used for other functionality can provide reset pulses to external peripherals.

The need for external RESET happens when nRF24Z1 is operated standalone and no 'intelligent' devices such as a microcontroller are available to provide RESET to peripherals (such as ADC or DAC).

External RESET will be executed as a part of the configuration routine which is performed after power-on-reset and after each reconfiguration.

Which nRF24Z1 I/O pin to use and the polarity of the reset signal are controlled by registers TXRESO (0x10) for ATX and RXRESO (0x40) for ARX peripherals.

The registers are described in the table below.

| Address Hex | Register | R/W | Initial Hex | Description | |
|-------------|---|-----|-------------|--|--|
| 0x50 | TXRESO | R/W | 0x00 | Optional RESET pulse output from ATX may be enabled. | |
| | | | | Bit | Interpretation |
| | | | | 7:3 | Reserved, MBZ |
| | | | | 2:1 | 0 : no RESET output 1 : RESET output on MSDA pin 2 : RESET output on MOSI pin 3 : RESET output on SPDIO pin |
| 0 | ATX RESET output polarity 0 : active low 1 : active high length of reset pulse is ca 285us | | | | |
| 0x40 | RXRESO | R/W | 0x00 | Optional RESET pulse output from ARX may be enabled. | |
| | | | | Bit | Interpretation |
| | | | | 7:3 | Reserved, MBZ |
| | | | | 2:1 | 0 : no RESET output 1 : RESET output on MSDA pin 2 : RESET output on MOSI pin 3 : RESET output on SPDIO pin |
| 0 | ARX RESET output polarity 0 : active low 1 : active high length of reset pulse is ca 285us | | | | |

Table 10-1 TXRESO and RXRESO registers

Values in TXRESO and RXRESO should be stored in a configuration EEPROM.



11 POWER DOWN CONTROL

The nRF24Z1 power down mode enables nRF24Z1 applications to set up a number of power saving routines.

11.1 Enter power down

nRF24Z1 can enter power down mode in two different ways, either by being forced in power down by external MCU, or by an automatic power down mechanism which is activated if no link is established. The normal way to initiate power down is for an external MCU connected to the ATX to first initiate power down on the ARX (register RXMOD[7]=0) and then initiate power down on the ATX (register TXMOD[6]=0). Register TXMOD is shown in Table 7-6, and register RXMOD is shown in Table 7-8. Note that *the nRF24Z1s will not enter power down mode unless a wake time is specified*. For ARX to enter power down RXWTI must be having a value that is different from 0x00. Similarly ATX must have a value bigger than 0x00 in TXWTI. See 11.2.2 for timer control registers. ARX will enter power down shortly after settings have been transferred to it. This transfer is initiated by writing 0x01 to RXCSTATE. If no link is established automatic power down will be activated after a period of $_XWTI * (_XLTI + 1)$. This feature can however be deactivated by configuring $_XWTI = 0x00$.

11.2 Wake up from power down

To emerge from power down there are two options; change on external pin and timer controlled. To deactivate power down mode, external MCU is required.

11.2.1 External pin change

To minimise current consumption the nRF24Z1 can be put in power down until the level on an external pin is changed to initiate wake up and new link initialisation.

In ATX there is one pin, DD[1], that can enable a wakeup. This feature is enabled in register TXMOD[5]. In ARX a wake-up can be initiate by a change on any of the input pins DI[3:0] wake-up enable is controlled by register RXWAKE. The external wake up control register is shown in Table 11-1. Register TXMOD is shown in Table 7-6. Wakeup on sleep timer has priority in this register. Therefore RXWAKE[4] must be cleared to enable external pin change wakeup. Any number, but at least one, of RXWAKE[0:3] can be set.

After a change of the specified input pin the nRF24Z1 will wake up and attempt to re-link for the specified wake time. If the power down bit is not reset during this interval it will return to power down. Power down timing is explained in 11.2.2 and the example in 11.3. For external pin wakeup the total wakeup time is: $_XWTI * (_XLTI + 1) * 10 \text{ ms}$.

Shortest pulse width for wakeup is 1 ms.



| Address Hex | Register | R/W | Initial Hex | Description | | | | | | | | | | | | | | | | |
|-------------|-----------------------|-----|-------------|---|-----|----------------|-----|---------------|---|----------|---|-----------------------|---|----------------------|---|----------------------|---|----------------------|---|----------------------|
| 0x45 | RXWAKE | R/W | 0x28 | Wakeup sources for audio receiver | | | | | | | | | | | | | | | | |
| | | | | <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7:6</td> <td>Reserved, MBZ</td> </tr> <tr> <td>5</td> <td>Reserved</td> </tr> <tr> <td>4</td> <td>Wakeup on sleep timer</td> </tr> <tr> <td>3</td> <td>Wakeup on DI3 change</td> </tr> <tr> <td>2</td> <td>Wakeup on DI2 change</td> </tr> <tr> <td>1</td> <td>Wakeup on DI1 change</td> </tr> <tr> <td>0</td> <td>Wakeup on DI0 change</td> </tr> </tbody> </table> | Bit | Interpretation | 7:6 | Reserved, MBZ | 5 | Reserved | 4 | Wakeup on sleep timer | 3 | Wakeup on DI3 change | 2 | Wakeup on DI2 change | 1 | Wakeup on DI1 change | 0 | Wakeup on DI0 change |
| Bit | Interpretation | | | | | | | | | | | | | | | | | | | |
| 7:6 | Reserved, MBZ | | | | | | | | | | | | | | | | | | | |
| 5 | Reserved | | | | | | | | | | | | | | | | | | | |
| 4 | Wakeup on sleep timer | | | | | | | | | | | | | | | | | | | |
| 3 | Wakeup on DI3 change | | | | | | | | | | | | | | | | | | | |
| 2 | Wakeup on DI2 change | | | | | | | | | | | | | | | | | | | |
| 1 | Wakeup on DI1 change | | | | | | | | | | | | | | | | | | | |
| 0 | Wakeup on DI0 change | | | | | | | | | | | | | | | | | | | |

Table 11-1 external wake-up control registers

11.2.2 Timer controlled wake up

The nRF24Z1 can also be run in a power saving mode where timers controls occasional wake up of the device to search for counter parts on the air. In this case the average current consumption will be higher than the power down current itself, but nRF24Z1 will be able to re-initialise the RF link without any user interaction. If the nRF24Z1 don't find any counterparts within a certain time interval after the wake-up it can go back to sleep.

When the nRF24Z1 enters power down the link is broken so ATX and ARX timing are no longer synchronised, a separate set of timers are hence found in ATX and ARX.

The timer wake up sequence is controlled by a 16 bit sleep interval timer (ATX: TXSTI, ARX: RXSTI) and an 8 bit wake interval timer (ATX: TXWTI, ARX: RXWTI). The contents of the sleep timer control registers sets how many 10 ms intervals the nRF24Z1 will sleep before it wakes up and tries to establish a RF link. The wake interval timers correspondingly sets how many 10 ms intervals the nRF24Z1 should attempt to re-link before it goes back to power down.

If the two paired nRF24Z1's are on the air at the same time, a link is established and the controlling device (MCU) on the ATX side must reset timers to prevent a return to power down again.

The sleep and wake timers are controlled by the registers shown in Table 11-2.



| Address Hex | Register | R/W | Initial Hex | Description |
|-------------|----------|-----|-------------|---|
| 0x57 | TXSTI[0] | R/W | 0x00 | Audio transmitter (and link) sleep timer byte #0 TXSTI is a 16-bit number specifying the number of 10 ms (nominal) periods the audio transmitter should sleep between attempting to establish a new link. |
| 0x58 | TXSTI[1] | R/W | 0x00 | Audio transmitter sleep timer byte #1 |
| 0x59 | TXWTI | R/W | 0x00 | Audio transmitter wake timer. With TXWTI set to 0, the audio transmitter will not go back to power down mode. A number larger than 0 will specify the number of 10ms (nominal) periods before the audio transmitter will reenter power down mode. |
| 0x4D | TXLTI | R/W | 0x00 | Audio transmitter wake time extension register, used to extend ATX wake time when woken by a pin change or in link mode. Total wake time will then be $(TXLTI+1)*TXWTI*10ms$ |
| 0x46 | RXSTI[0] | R/W | 0x00 | Audio receiver sleep timer byte #0 RXSTI is a 16-bit number specifying the number of 10 ms (nominal) periods the audio receiver should sleep between attempting to establish a new link. Only used if sleep timer is enabled, see bit 4 of RWAKE register. |
| 0x47 | RXSTI[1] | R/W | 0x00 | Audio receiver sleep timer byte #1 |
| 0x48 | RXWTI | R/W | 0x00 | Audio receiver wake timer. With RXWTI set to 0, the audio receiver will not go back to power down mode. A number larger than 0 will specify the number of 10ms (nominal) periods before the audio receiver will reenter power down mode. |
| 0x4C | RXLTI | R/W | 0x00 | Audio receiver wake time extension register, used to extend ARX wake time when woken by a pin change or in link mode. Total wake time will then be $(RXLTI+1)*RXWTI*10ms$ |

Table 11-2 sleep and wake timer registers



11.3 nRF24Z1 power saving example

Combining the two power down control methods described above is of course no problem. Figure 11-1 shows an example of timer controlled power down in ATX combined with power down with wake up on external pin event in ARX. In this example the ARX is shut down completely until a user activates the receiver (headset or speaker).

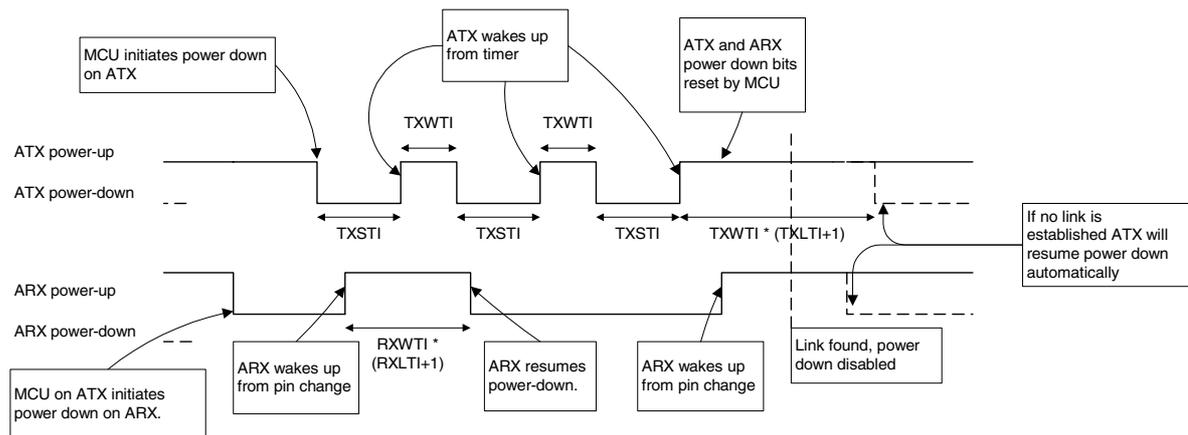


Figure 11-1 : nRF24Z1 power saving

In this example the ARX will wake up by pin change and stay awake for a time interval attempting to link. It will resume power down if the status still is set to power down.

ATX will wake up from power down at regular intervals. In order to wake up ATX, this interval must be used to reset the power down bit.

Now, while ATX has reset the ARX power down bit, if the ARX wakes up and establish a link, the new setting will cause the player and receiver to stay awake. ATX will resume power down if no link is found after the extended wake time interval is over; even when the power down bit was reset.



12 REGISTER UPDATE OVER THE DATA LINK

The registers TXCSTATE, LNKSTATE, RXCSTATE can be used by an ATX MCU to update audio receiver control registers via the data link. Write to these registers from an ARX MCU is illegal. When an MCU at ATX has to cause a change to one or more registers in ARX, the register value(s) are first buffered in ATX. The registers are transferred to ARX and employed there when the corresponding *update register* is written to by the ATX MCU. Prior to any transfer to ARX, the transfer register should be polled to see if the ATX is ready to transfer new register values to ARX.

Setting an update register to 0x01 tells the ATX to send the corresponding register values to ARX. The update register will be reset to 0x00 after the registers values are successfully sent. When an update register reads 0x01, the ATX is busy sending the register values to ARX. When an update register reads 0x02, the last transfer was unsuccessful. The MCU at ATX should poll the update register for a value 0x00 or 0x02 before starting to buffer new register values in the nRF24Z1 ATX. A value 0x02 may indicate a problem in the radio link.

| Address Hex | Register | R/W | Initial Hex | Description |
|-------------|-----------|-----|-------------|--|
| 0x5B | TXCSTATE | R/W | 0x00 | Controls when to send ATX side registers TXFMT, TXLAT, I2SCNF_IN, I2SRAT over the data link to ARX. Status values are : 0 : idle, and last transfer was successful 1 : busy, the registers may not be accessed. 2 : idle, but last transfer was unsuccessful when idle, ATX side registers are free to be written. Setting TXCSTATE=1 tells ATX to send the registers values to ARX, and TXCSTATE will be reset to 0 by ATX upon successful transfer to ARX. An external MCU should poll this register before accessing these register. |
| 0x3E | LNKCSTATE | R/W | 0x00 | Controls when to send ATX side Link control registers over the data link to ARX. Status values are : 0 : idle, and last transfer was successful 1 : busy, the registers may not be accessed. 2 : idle, but last transfer was unsuccessful when idle, Link control registers are free to be written. Setting LNKSTATE=1 tells ATX to send Link control registers values to ARX, and LNKSTATE will be reset to 0 by ATX upon successful transfer to ARX. An external MCU should poll this register before accessing any Link control register. |



| | | | | |
|------|----------|-----|------|--|
| 0x4B | RXCSTATE | R/W | 0x00 | <p>Controls when to send ATX side ARX control registers over the data link to ARX. Status values are :</p> <p>0 : idle, and last transfer was successful 1 : busy, the registers may not be accessed. 2 : idle, but last transfer was unsuccessful when idle, ARX control registers are free to be written.</p> <p>Setting RXCSTATE=1 tells ATX to send all ARX control registers values to ARX, and RXSTATE will be reset to 0 by ATX upon successful transfer to ARX. An external MCU should poll this register before accessing any ARX control register.</p> |
|------|----------|-----|------|--|

Table 12-1 : register update registers

12.1 Register update and relink

Some register updates require that the ARX and ATX are reconfigured, these are listed below. ATX and ARX will always be reconfigured after a successful link is established; it is therefore possible to force ATX and ARX to be reconfigured by forcing them to re-link. This can be done by setting the “Force reconfiguration” bit, LNKMOD[4].

ATX registers:

- TXSTA[6:3]
- TXFMT
- TXMOD[2:0]
- I2SCNF_IN
- TXLAT

ARX registers:

- RXMOD[2]
- I2SCNF_OUT[5:0]

LINK registers:

- ADDR0, ADDR1, ADDR2, ADDR3, ADDR4
- NBCH
- NACH

TEST_MODE registers:

- TESTREG
- TESTCH

Some registers have additional or special functionality. These are listed below:



- **RXSTA[6]**
The ARX serial slave interface is only possible to disable by setting RXSTA[6] in an external configuration EEPROM, if no EEPROM is present on ARX serial slave interface will be enabled by default.
- **CH0-37**
The channel table is not transferred from ATX to ARX. These registers must therefore be configured in EEPROM or local MCU.
- **TXSTA**
Sampling rates are automatically sent to ARX from ATX during link-locate mode.
- **RXMOD**
ARX operation mode is automatically sent to ARX from ATX during link-locate mode.
- **RXPIN**
Information about ARX general-purpose input pins is automatically sent from ARX to ATX from ARX during link-locate mode in addition to being continuously transferred from ARX to ATX during normal audio streaming mode.
- **TXCSTATE**
This is a transfer register employed by ATX to send data to ARX.
- **RXCSTATE**
This is a transfer register employed by ATX to send data to ARX.
- **LNKCSTATE**
This is a transfer register employed by ATX to send data to ARX.
- **RXEXEC**
This is a transfer register employed by ATX to send and receive data to/from the SPI and 2-wire master interfaces of ARX.



13 TEST MODE

To set nRF24Z1 into test mode you have to write to two test registers, TESTREG and TESTCH, and thereafter set bit 4 of LNKMOD register. This will force the chip to restart in test mode according to the values of TESTREG and TESTCH registers. nRF24Z1 will stay in test mode until the chip is reset. So the only way to abort a test mode is to reset the chip, and it is not allowed to change test mode without a reset. This approach shall be used for both ATX and ARX.

For ATX the test registers can be accessed in the normal way via the (SPI or 2-wire) slave interface. And so also for ARX configured with slave interface (RXSTA[6]=0). But for ARX configured from external EEPROM with GPIO interface (EEPROM/RXSTA[6]=1)., there is no normal way of setting the test registers, since there is no slave interface.

A way to avoid reading EEPROM is to force EEPROM data output line high, which will abort reading from EEPROM, and RXSTA[6] will not be set, and ARX will be configured with slave interface.

| Address Hex | Register | R/W | Initial Hex | Description | | | | | | |
|-------------|---|-----|-------------|--|----------------|----------------|---|---|-----|---|
| 0x7E | TESTREG | W | 0x00 | Test mode register: Code 1: 0110 0011 – Single channel test. Code 2: 0111 0011 – Channel sweep test. Sweeps all channels from frequencies from 2400 MHz to 2480 MHz in steps of 1 MHz. | | | | | | |
| 0x7F | TESTCH | W | 0x00 | <table border="1"> <thead> <tr> <th>Bit</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>1: TX, 0: RX Decide if the two tests described in TESTREG is to be in RX or TX mode.</td> </tr> <tr> <td>6:0</td> <td>Channel number when TESTREG is set to Code 1 (single channel), number is in 1MHz step relative to 2400MHz</td> </tr> </tbody> </table> | Bit | Interpretation | 7 | 1: TX, 0: RX Decide if the two tests described in TESTREG is to be in RX or TX mode. | 6:0 | Channel number when TESTREG is set to Code 1 (single channel), number is in 1MHz step relative to 2400MHz |
| | | | | Bit | Interpretation | | | | | |
| 7 | 1: TX, 0: RX Decide if the two tests described in TESTREG is to be in RX or TX mode. | | | | | | | | | |
| 6:0 | Channel number when TESTREG is set to Code 1 (single channel), number is in 1MHz step relative to 2400MHz | | | | | | | | | |

Table 13-1 : Test mode registers.



14 ELECTRICAL SPECIFICATION

| Symbol | Parameter (condition) | Notes | Min. | Typ. | Max. | Units |
|---|--|-------|----------|--------|---------|-------|
| Operating conditions | | | | | | |
| VDD | Supply voltage | | 2.0 | 3.0 | 3.6 | V |
| TEMP | Operating temperature | | -20 | 27 | 80 | °C |
| Digital input pins | | | | | | |
| V _{IH} | HIGH level input voltage | | 0.7 *VDD | | VDD | V |
| V _{IL} | LOW level input voltage | | VSS | | 0.3*VDD | V |
| Digital output pins | | | | | | |
| V _{OH} | HIGH level output voltage (I _{OH} =-0.5mA) | | VDD-0.3 | | VDD | V |
| V _{OL} | LOW level output voltage (I _{OL} =0.5mA) | | VSS | | 0.3 | V |
| I _{OL_HD} | High drive sink current for DO[0] and DO[2] @ V _{OL} = 0.4V | 1 | | 10 | | mA |
| I _{OH_HD} | High drive source current for DO[1] and DO[3] @ V _{OH} = VDD-0.4V | 1 | | 10 | | mA |
| General electrical specification | | | | | | |
| I _{PD} | Supply current in power down mode | | | 4 | | µA |
| General RF conditions | | | | | | |
| f _{OP} | Operating frequency | 2 | 2400 | | 2521 | MHz |
| Δf | Frequency deviation | | | +/-640 | | kHz |
| R _{GFSK} | GFSK data rate | | | 4000 | | kbps |
| f _{CHANNEL} | Channel spacing | | | 4 | | MHz |
| f _{XTAL} | Crystal frequency | 3 | | 16 | | MHz |
| C _{load} | Crystal load capacitance | 3 | 8 | 12 | 16 | pF |
| Δ f _{XTAL} | Crystal frequency tolerance + temperature drift | 3 | | | +/-30 | ppm |
| RF Transmit mode | | | | | | |
| P _{RF_0dBm} | Maximum output power (TXPWR=3) | 4 | | 0 | 3 | dBm |
| P _{RF_-5dBm} | Maximum output power (TXPWR=2) | 4 | | -5 | 0 | dBm |
| P _{RF_-10dBm} | Maximum output power (TXPWR=1) | 4 | | -10 | -5 | dBm |
| P _{RF_-20dBm} | Maximum output power (TXPWR=0) | 4 | | -20 | -12 | dBm |
| P _{RFC} | RF power control range | | 16 | 20 | | dB |
| P _{RFCR} | RF power range control resolution | | | | +/-3 | dB |
| P _{BW} | 20dB bandwidth for modulated carrier | | | 4000 | | kHz |
| I _{TX_0dBm} | Supply current @ 0dBm output power | | | 17 | | mA |
| I _{TX_-5dBm} | Supply current @ -5dBm output power | | | 15 | | mA |
| I _{TX_-10dBm} | Supply current @ -10dBm output power | | | 14 | | mA |
| I _{TX_-20dBm} | Supply current @ -20dBm output power | | | 13 | | mA |
| RF Receive mode | | | | | | |
| I _{RX} | Supply current in receive mode | | | 32 | | mA |
| RX _{SENS} | Sensitivity at 0.1%BER | | | -80 | | dBm |
| RX _{MAX} | Maximum received signal | | 0 | | | dBm |

¹ Output pin programmed for high current (register RXPIO)

² Device operates in the 2400 MHz ISM band (2400-2483 MHz).

³ For further details on crystal specification, please see section 17.2

⁴ Antenna load impedance = 100Ω+j175Ω, please see chapter 17 application information



| I2S interface timing | | (see also Figure 7-2 and Figure 7-3) | | | | |
|--|---|--------------------------------------|------|-----|------|---------|
| T_{I2S} | I2S clock period | | 150 | | | ns |
| t_{sI2S} | DATA and WS (input) setup time to CLK | | 20 | | | ns |
| t_{hI2S} | DATA and WS (input) hold time from CLK | | 20 | | | ns |
| t_{dI2S} | DATA and WS (output) delay from CLK | | | | 40 | ns |
| MCLK (256 F_s) output clock | | | | | | |
| Δf_{MCLK} | Locking range vs nominal MCLK frequency | 1 | -500 | | +500 | ppm |
| J_{RMS} | RMS jitter 0-25kHz | | | 250 | 310 | ps |
| Slave SPI interface timing | | (see also Figure 7-10) | | | | |
| T_{SSCK} | SSCK clock period | 2 | 62 | | | ns |
| t_{suSSPI} | SMOSI setup time to SSCK | | 10 | | | ns |
| t_{hdSSPI} | SMOSI hold time from SSCK | | 10 | | | ns |
| t_{dSSPI} | SMISO delay from SSCK | | | | 55 | ns |
| t_{dSSCK} | SCSN setup time to SSCK | | 500 | | | μ s |
| t_{SRD} | SPI slave ready | | 500 | | | μ s |
| Master SPI interface timing | | (see also Figure 7-4) | | | | |
| T_{MSCK} | MSCK clock period | | 125 | | | ns |
| t_{suMSPI} | MMISO setup time to MSCK | | 55 | | | ns |
| t_{hdMSPI} | MMISO hold time from MSCK | | 10 | | | ns |
| t_{dMSPI} | MMOSI delay from MSCK | | | | 20 | ns |
| t_{dMSCK} | MCSN setup to MSCK | | 30 | 500 | | ns |
| Slave 2-wire interface timing | | (see also Figure 7-11) | | | | |
| T_{SSCL} | 2-wire clock period | | 1000 | | | ns |
| t_{SW2_dsu} | SSDA setup time to SSCL | | 50 | | | ns |
| t_{SW2_dhd} | SSDA hold time from SSCL | | 65 | | | ns |
| t_{SW2_od} | SSDA 1->0 delay from SSCL | | | | 170 | ns |
| Master 2-wire interface timing | | (see also Figure 7-6) | | | | |
| T_{MSCL} | 2-wire clock period | | 1000 | | | ns |
| t_{MW2_dsu} | MSDA setup time to MSCL | | 60 | | | ns |
| t_{MW2_dhd} | MSDA hold time from MSCL | | 50 | | | ns |
| t_{MW2_od} | MSDA 1->0 delay from MSCL | | | | 50 | ns |

Table 14-1 nRF24Z1 electrical specification.

¹ Nominal MCLK frequency is 256 times f_s for f_s in [32kHz, 44.1kHz, 48kHz] programmable

² For VDD 3.0V +/-10%, otherwise minimum T_{SSCK} is 124ns (8MHz)



15 ABSOLUTE MAXIMUM RATINGS

Supply Voltages

VDD - 0.3V to + 3.6V

VSS 0V

Input Voltage

V_I - 0.3V to VDD + 0.3V

Output Voltage

V_O - 0.3V to VDD + 0.3V

Total Power Dissipation

P_D (T_A=85°C) 115mW

Temperatures

Operating temperature - 20°C to + 80°C

Storage temperature - 40°C to + 125°C

Note: Stress exceeding one or more of the limiting values may cause permanent damage to the device.

ATTENTION!

Electrostatic sensitive device.

Observe precaution for handling.





16 PACKAGE OUTLINE

nRF24Z1 is packaged in a 36 pin 6 by 6 QFN (all dimensions in mm) matt tin plating.

| Package Type | | A | A ₁ | A ₂ | b | D/E | D1/E1 | e | J | K | L | R |
|----------------------------|------|-----|----------------|----------------|------|-------|----------|---------|------|------|-----|-------|
| Green QFN36 (6x6 mm) | Min | 0.8 | 0.0 | 0.65 | 0.18 | 6 BSC | 5.75 BSC | 0.5 BSC | 4.47 | 4.47 | 0.3 | 1.735 |
| | typ. | | 0.02 | | 0.23 | | | | 4.57 | 4.57 | 0.4 | 1.835 |
| | Max | 0.9 | 0.05 | 0.69 | 0.3 | | | | 4.67 | 4.67 | 0.5 | 1.935 |

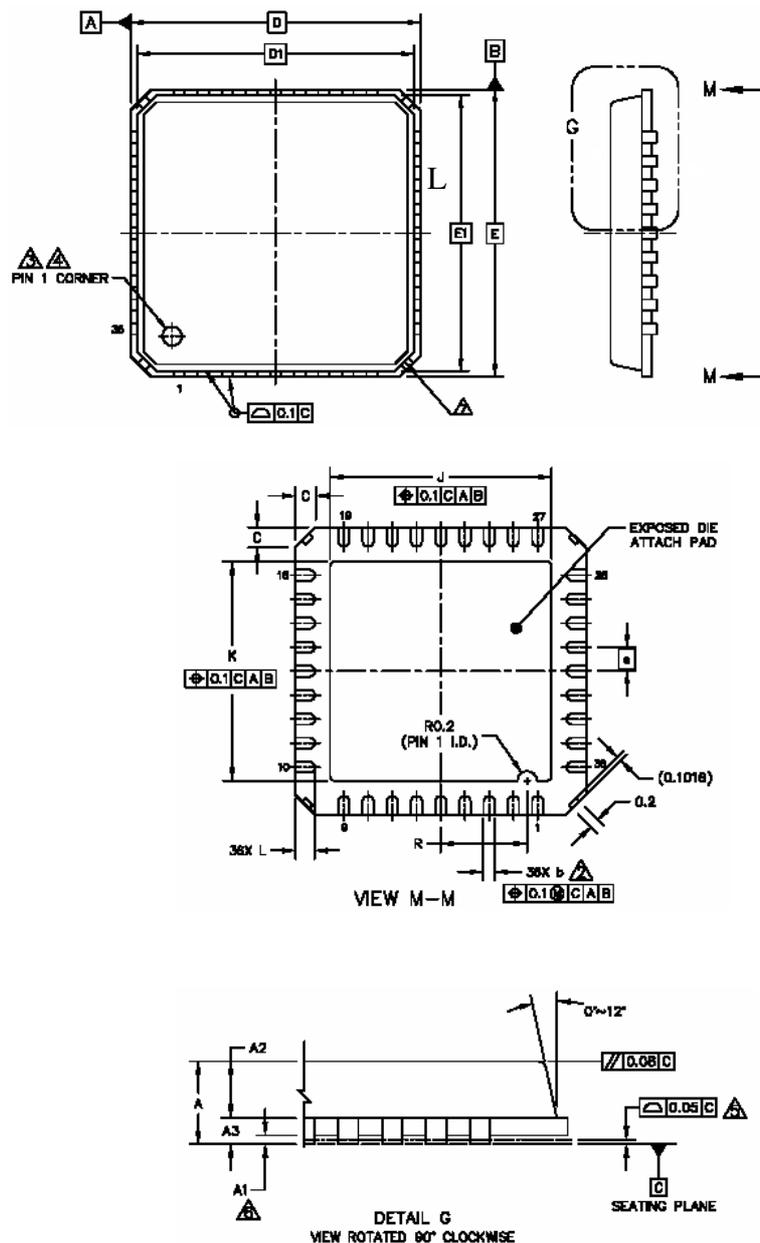


Figure 16-1 : nRF24Z1 package outline.



17 APPLICATION INFORMATION

17.1 Antenna I/O

The ANT1 & ANT2 output pins provide a balanced RF output to the antenna. The pins must have a DC path to VDD_PA, either via a RF choke or via the center point in a balanced antenna. Differential load impedance between the ANT1 and ANT2 pins, $100\Omega + j175\Omega$, is recommended for maximum output power. Antennas with lower load impedance (for example 50Ω) can be matched to nRF24Z1 by using a simple matching network.

17.2 Crystal Specification

Tolerance includes initially accuracy and tolerance over temperature and aging.

| Frequency | C_L | ESR | C_{0max} | Tolerance |
|-----------|------------|--------------|------------|--------------|
| 16MHz | 8pF – 16pF | 100 Ω | 7.0pF | ± 30 ppm |

Table 17-1 Crystal specification of nRF24Z1

To achieve a crystal oscillator solution with low power consumption and fast start-up time, it is recommended to specify the crystal with a low value of crystal load capacitance.

The crystal load capacitance, C_L , is given by:

$$C_L = \frac{C_1' C_2'}{C_1' + C_2'}, \quad \text{where } C_1' = C_1 + C_{PCB1} + C_{I1} \text{ and } C_2' = C_2 + C_{PCB2} + C_{I2}$$

C_1 and C_2 are SMD capacitors as shown in the application schematics. C_{PCB1} and C_{PCB2} are the layout parasitic on the circuit board. C_{I1} and C_{I2} are the capacitance seen into the XC1 and XC2 pin respectively; the value is typical 1pF.

17.3 Bias reference resistor

A resistor between pin IREF (pin24) and ground sets up the bias reference for the nRF24Z1. A 22 k Ω (1%) resistor is to be fitted, and changing the value of this resistor will degrade the nRF24Z1 performance directly.

17.4 Internal digital supply de-coupling

Pin DVDD (pin15) is a regulated output of the internal digital power supply of nRF24Z1. The pin is purely for de-coupling purposes and only a 33 nF (X7R) capacitor is to be connected. The pin must not be connected to external VDD and can not be used as power supply for external devices.



17.5 PCB layout and de-coupling guidelines

A well-designed PCB is necessary especially to achieve good RF performance. Keep in mind that a poor layout may lead to loss of performance, or even functionality. A fully qualified RF-layout for the nRF24Z1 and its surrounding components, including antenna matching network, can be downloaded from www.nordicsemi.no.

A PCB with a minimum of two layers with ground planes is recommended for optimum performance. The nRF24Z1 DC supply voltage should be de-coupled as close as possible to the VDD pins, see ch.18. It is preferable to mount a large surface mount capacitor (e.g. 4.7 μ F tantalum) in parallel with the smaller value capacitors. The nRF24Z1 supply voltage should be filtered and routed separately from the supply voltages of other circuitry.

Long power supply lines on the PCB should be avoided. All device grounds, VDD connections and VDD bypass capacitors must be connected as close as possible to the nRF24Z1 IC. All the VSS pads in a nRF24Z1 layout should be connected directly to a ground plane and one via should be put as close as possible to each of them. Full swing digital data or control signals should not be routed close to the crystal or the power supply lines.



18 APPLICATION EXAMPLE

The following example shows a nRF24Z1 schematic and layout where the ATX is controlled over SPI by an external MCU and the ARX has a SPI EEPROM attached for stand alone operation.

Interfaces that are routed out from nRF24Z1:

- Audio: I2S
- ATX control: SPI slave
- ATX GPIO: DD[1:0]
- ATX external interrupt pin (IRQ)
- ARX GPIO: DI[3:0], DO[3:0]

18.1 nRF24Z1 schematics

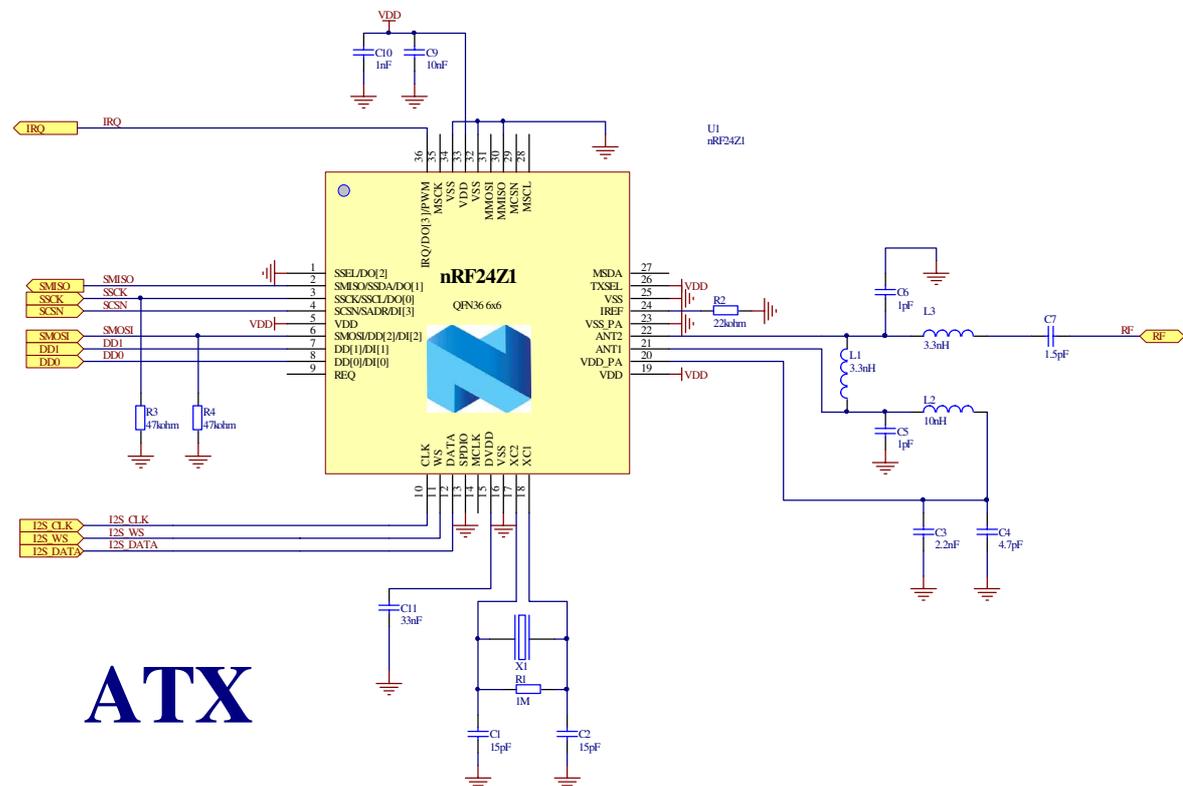
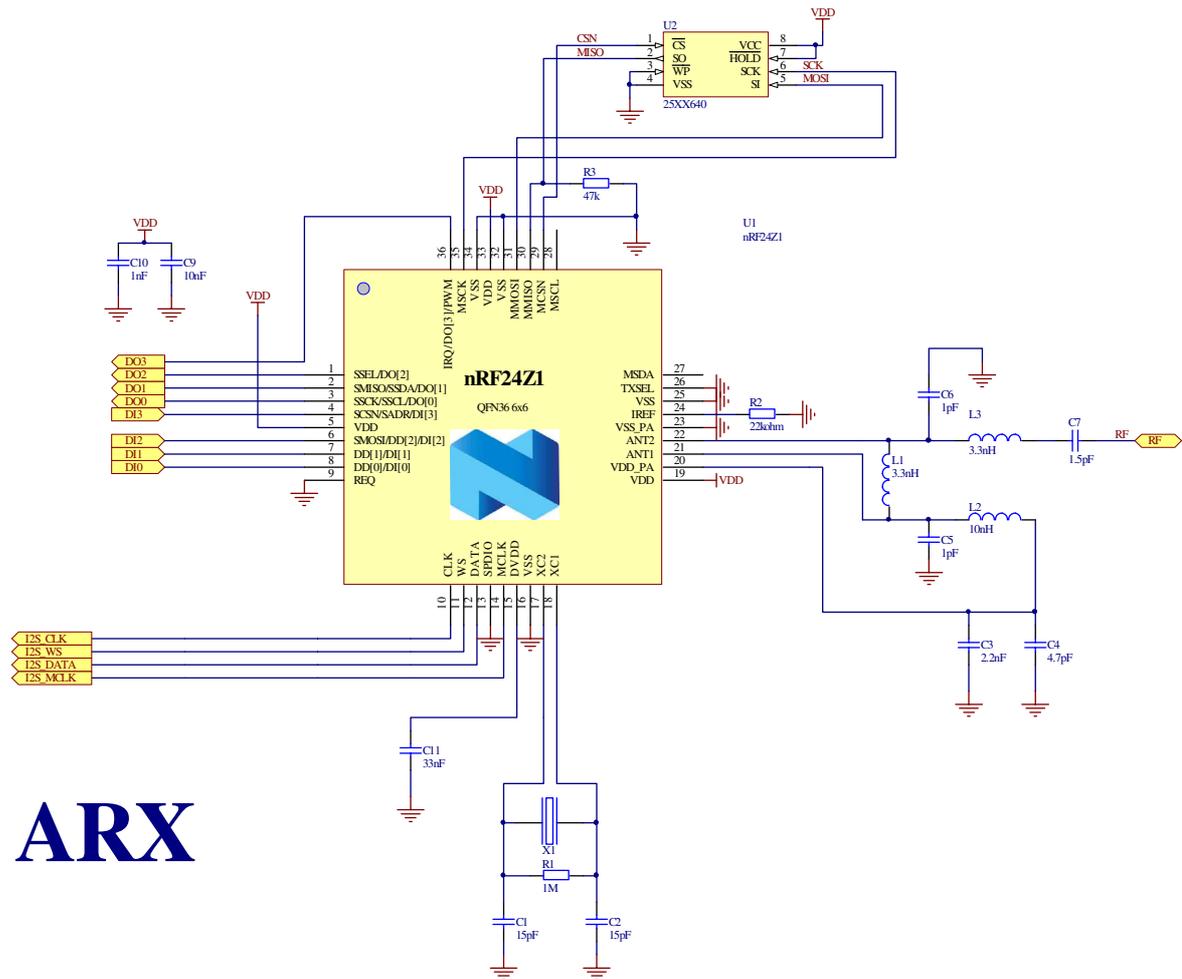


Figure 18-1 nRF24Z1 MCU controlled ATX schematic

Resistors R3 and R4 are not necessary for device functionality. R3 is added to guarantee that no nRF24Z1 registers are changed if external MCU is resetting. R4 is put in to terminate the nRF24Z1 SPI input to avoid any floating signals if the SPI bus is disabled (power down). Only one resistor on the bus is needed and if MCU MOSI output has internal pull up/down it can be omitted.



nRF24Z1 wireless audio streamer



ARX

Figure 18-2 nRF24Z1 ARX with SPI EEPROM schematic



18.2 nRF24Z1 layout

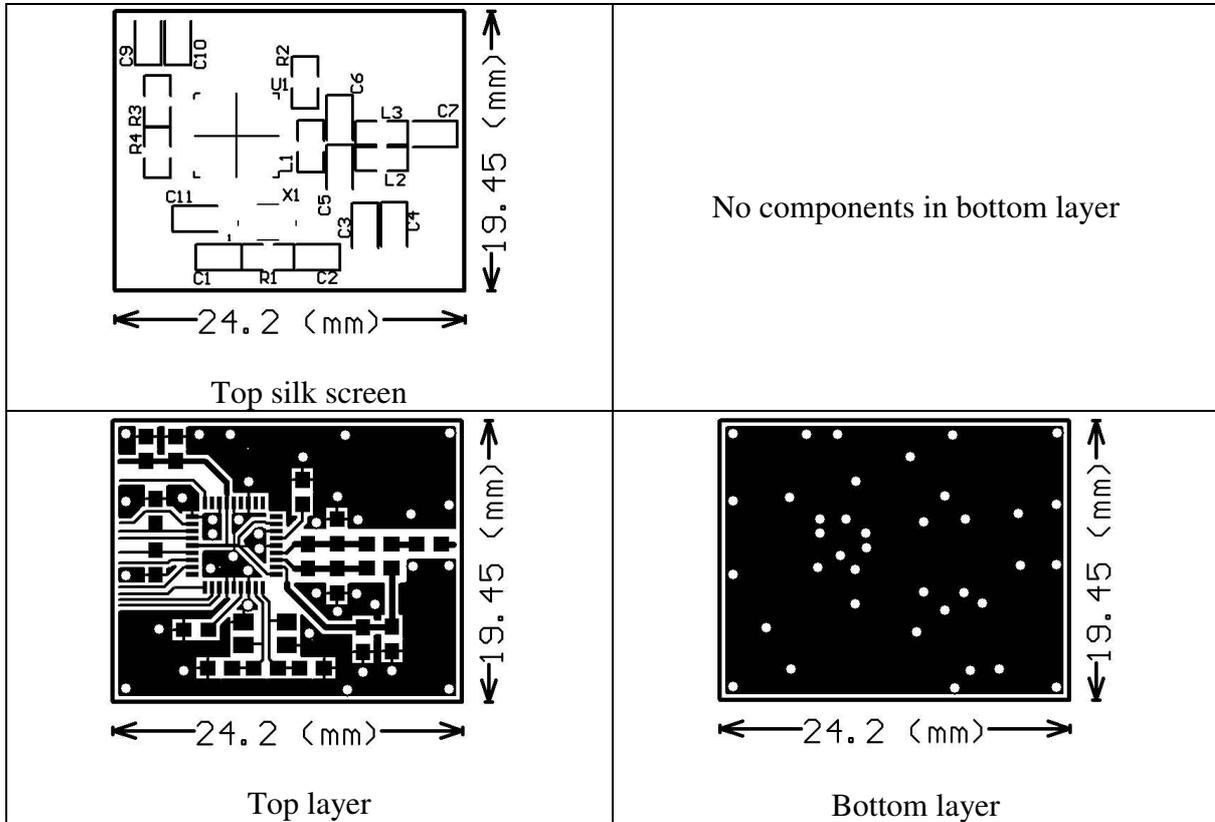


Figure 18-3 : nRF24Z1 ATX PCB layout

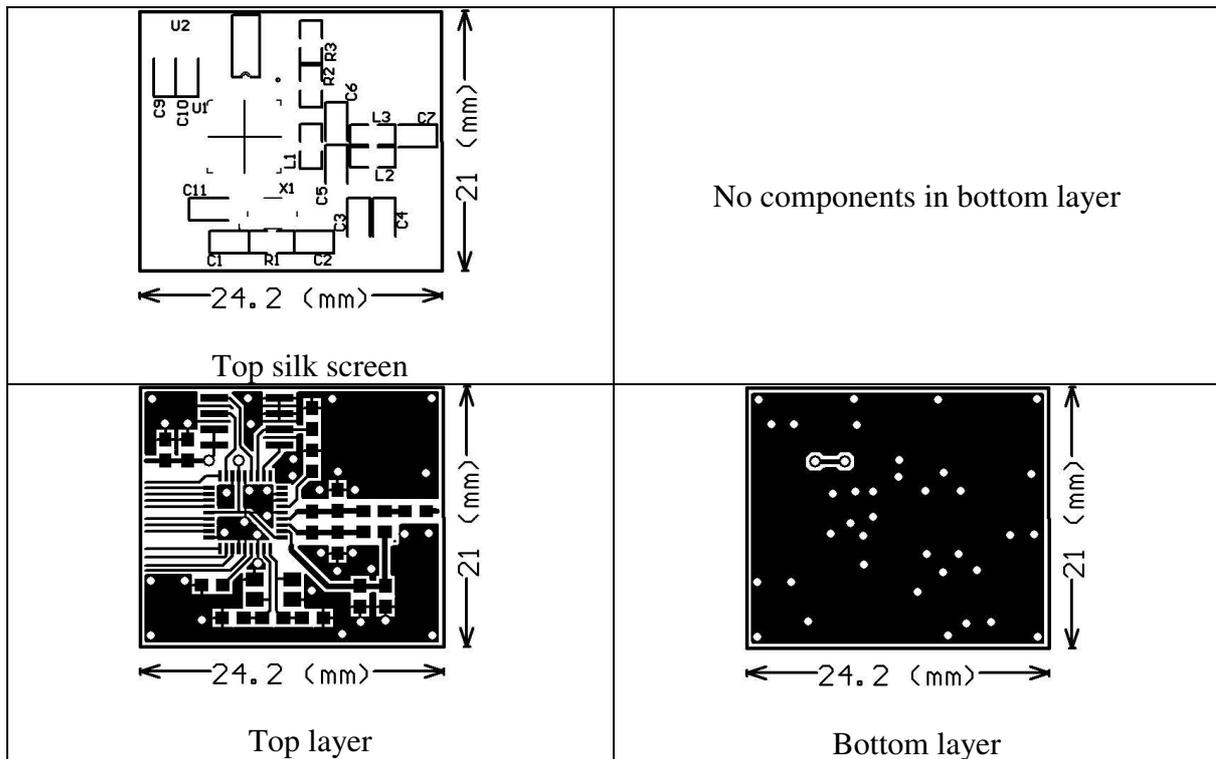


Figure 18-4 : nRF24Z1 ARX PCB layout



18.3 nRF24Z1 Bill of Material

ATX:

| Designator | Value | Description | Footprint |
|------------|---------|---|-------------|
| C1 | 15pF | Capacitor NP0 +/- 5% | CC1608-0603 |
| C2 | 15pF | Capacitor NP0 +/- 5% | CC1608-0603 |
| C3 | 2.2nF | Capacitor X7R, +/- 10% | CC1608-0603 |
| C4 | 4.7pF | Capacitor NP0 +/- 5% | CC1608-0603 |
| C5 | 1pF | Capacitor NP0 +/- 0.1 pF | CC1608-0603 |
| C6 | 1pF | Capacitor NP0 +/- 0.1 pF | CC1608-0603 |
| C7 | 1.5pF | Capacitor NP0 +/- 0.25pF | CC1608-0603 |
| C9 | 10nF | Capacitor X7R, +/- 10% | CC1608-0603 |
| C10 | 1nF | Capacitor X7R, +/- 10% | CC1608-0603 |
| C11 | 33nF | Capacitor X7R, +/- 10% | CC1608-0603 |
| L1 | 3.3nH | TOKO LL1608-FS chip inductor series* | CR1608-0603 |
| L2 | 10nH | Inductor Chip, +/- 5% | CR1608-0603 |
| L3 | 3.3nH | TOKO LL1608-FS chip inductor series* | CR1608-0603 |
| R1 | 1M | Resistor 5% | CR1608-0603 |
| R2 | 22kohm | Resistor 1 % | CR1608-0603 |
| R3 | 47kohm | Resistor 5% | CR1608-0603 |
| R4 | 47kohm | Resistor 5% | CR1608-0603 |
| U1 | nRF24Z1 | 2.4 GHz audio streamer | QFN36L/6x6 |
| X1 | 16 MHz | Crystal Cl=9pF, ESR < 100 ohm, stab.+ drift = +/-30ppm, | BT-XTAL |

Table 18-1 nRF24Z1 ATX BOM

* Inductance vs. frequency may differ significantly in inductors with the same value but different part numbers and/or vendors!

Inductor value is usually characterized at 100-250 MHz. But actual value at 2.4 GHz may vary significantly even though the given inductance at 250 MHz is the same.

Inductors from other TOKO series and other vendors may well be used, but antenna match performance MUST be verified as inductor value may need to be changed.

**ARX:**

| Designator | Value | Description | Footprint |
|------------|---------|---|-------------|
| C1 | 15pF | Capacitor NP0 +/-5% | CC1608-0603 |
| C2 | 15pF | Capacitor NP0 +/-5% | CC1608-0603 |
| C3 | 2.2nF | Capacitor X7R +/-10% | CC1608-0603 |
| C4 | 4.7pF | Capacitor NP0 +/-5% | CC1608-0603 |
| C5 | 1pF | Capacitor NP0 +/-0.1 pF | CC1608-0603 |
| C6 | 1pF | Capacitor NP0 +/-0.1 pF | CC1608-0603 |
| C7 | 1.5pF | Capacitor NP0 +/-0.25 pF | CC1608-0603 |
| C9 | 10nF | Capacitor X7R +/-10% | CC1608-0603 |
| C10 | 1nF | Capacitor X7R +/-10% | CC1608-0603 |
| C11 | 33nF | Capacitor X7R +/-10% | CC1608-0603 |
| L1 | 3.3nH | TOKO LL1608-FS chip inductor series* | CR1608-0603 |
| L2 | 10nH | Chip Inductor, +/-5% | CR1608-0603 |
| L3 | 3.3nH | TOKO LL1608-FS chip inductor series* | CR1608-0603 |
| R1 | 1M | Resistor 5% | CR1608-0603 |
| R2 | 22kohm | Resistor 1% | CR1608-0603 |
| R3 | 47k | Resistor 5% | CR1608-0603 |
| U1 | nRF24Z1 | 2.4 GHz audio streamer | QFN36L/6x6 |
| U2 | 25XX640 | EEPROM | SO-G8 |
| X1 | 16 MHz | Crystal Cl=9pF, ESR < 100 ohm, stab.+ drift = +/-30ppm, | BT-XTAL |

Table 18-2 nRF24Z1 ARX BOM

* Inductance vs. frequency may differ significantly in inductors with the same value but different part numbers and/or vendors!

Inductor value is usually characterized at 100-250 MHz. But actual value at 2.4 GHz may vary significantly even though the given inductance at 250 MHz is the same.

Inductors from other TOKO series and other vendors may well be used, but antenna match performance MUST be verified as inductor value may need to be changed.



19 REFERENCES

For latest version of documents, please visit <http://www.nordicsemi.no>



20 DEFINITIONS

| Data sheet status | |
|---|---|
| Objective product specification | This datasheet contains target specifications for product development. |
| Preliminary product specification | This datasheet contains preliminary data; supplementary data may be published from Nordic Semiconductor ASA later. |
| Product specification | This datasheet contains final product specifications. Nordic Semiconductor ASA reserves the right to make changes at any time without notice in order to improve design and supply the best possible product. |
| Limiting values | |
| Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Specifications sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability. | |
| Application information | |
| Where application information is given, it is advisory and does not form part of the specification. | |

Table 20-1. Definitions.

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function or design. Nordic Semiconductor does not assume any liability arising out of the application or use of any product or circuits described herein.

LIFE SUPPORT APPLICATIONS

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

Preliminary product specification revision date: 2005-07-25

All rights reserved ®. Reproduction in whole or in part is prohibited without the prior written permission of the copyright holder.



21 YOUR NOTES



Nordic Semiconductor ASA – World Wide Distributors

For Your nearest dealer, please see <http://www.nordicsemi.no>



Main Office:

Vestre Rosten 81, N-7075 Tiller, Norway
Phone: +47 72 89 89 00, Fax: +47 72 89 89 89

E-mail: nRF@nordicsemi.no

Visit the Nordic Semiconductor ASA website at <http://www.nordicsemi.no>

