Using the microID™ Programmer

1.0 INTRODUCTION

The following is a description of how to program Microchip's MCRF2XX family of RFID products. A contactless programmer (PG103001), user interface software (RFLAB™), and a host computer are needed to program the MCRF2XX devices. The device can also be programmed in a standard terminal mode (i.e., c:\windows\terminal.exe) rather than the RFLAB. See Figure 5-1 for the programming sequence.

The microID programmer requires an external power supply (+9 VDC, >750 mA). The RFLAB software runs under Microsoft® (MS) Windows® 95 environment only. The programmer communicates with a host computer via an RS-232 serial interface at 9600 baud, 8 data bits, 1 stop bit, and no parity.

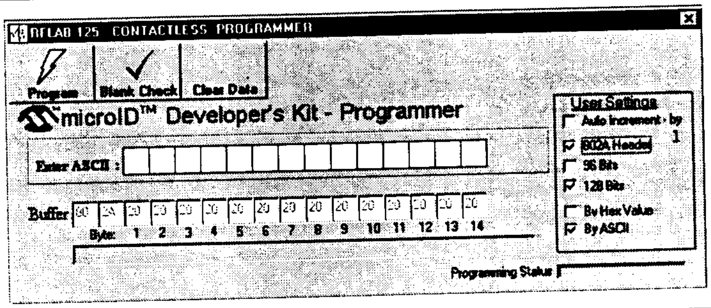
Since the MCRF2XX is a One-Time-Programmable (OTP) device, only a blank (unlocked) device can be programmed by the programmer. Therefore, the programmer first checks the status of the memory in the device before initiating programming. A blank device contains an array of all '1's.

The device can be programmed with 16 bytes (128 bits) or 12 bytes (96 bits) of data length. Once the MCRF2XX enters its programming mode, it sets a lock bit at the same time. If the programming is interrupted for any reason during the programming period, the programming will be stopped, and the device may be left partially programmed. The device will still be locked even though it has not been programmed completely. In this case, the programmer will return a fail code to the host computer.

Any device that has been programmed, either fully or partially, will remain in a locked status; therefore, it cannot to be reprogrammed. If programming has been successfully completed, the programmer will return a verification code to the host computer.

In order to program the MCRF2XX device, it is necessary to provide a proper programming signal level to the device. The device requires specific peak-to-peak voltages for programming. Since the voltage induced in the tag coil varies depending on the coil parameters, the output signal level of the programmer must be calibrated to provide a proper programming signal level at the tag coil. A detailed calibration procedure is described in Section 3.0.

FIGURE 1-1: RFLAB SOFTWARE RUNNING UNDER MS WINDOWS 95



RFLAB is a trademark of Microchip Technology Inc.

2.0 PROGRAMMING SIGNAL WAVEFORM

Figure 2-1 shows the waveform of the programming signal. Once the programmer sends a power-up and gap signal to the device, the device transmits back a verification bitstream in FSK. The verification signal represents the contents of the memory in the device. The blank device has all '1's in its memory. A bit '1' in FSK is represented by a low signal level for five cycles and a high signal level for an additional five cycles (Figure 2-1).

The device will respond with a nonmodulated (no data) signal if the device has not recognized the power-up signal. In this case, the power-up signal level should be calibrated to provide a proper signal level to the device. The calibration procedure is explained in Section 3.0.

After the device is verified as blank, the programmer sends a programming signal to the device. The programming data is represented by an amplitude modulation signal. Therefore, bit '1' and '0' are represented by a low-power (level) signal and a high power (level) signal, respectively, as shown in Figure 2-1. Each data bit is represented by 128 cycles of the carrier signal. An MCRF200 configured for 128 bits uses all bits in the transfer; an MCRF200 configured for 96 bits ignores bits 33 through 64, although they are present in the programming sequence. Therefore, for a 125 kHz carrier signal, it takes 1.024 ms for one data bit (128 cycles x 8 μs/cycles) and 131.072 ms for 128 data bits (128 cycles/bit x 8 μs/cycle x 128 bits).

A guard-band of Δt = 10 cycles (80 μs) should be kept at each end of a high-power (0) bit as shown in Figure 2-1. This is to prevent accidental programming or disturbing of adjacent bits in the array.

The memory array is locked at the start of the programming cycle. Therefore, when the device leaves the programming field, it locks the memory permanently, regardless of the programming status. The device should not be interrupted during the programming cycle.

The device transmits the programmed (data contents) circuits back to the programmer for verification. If the verification bitstream is correct, the programmer sends a verified signal ('v') to the host computer; otherwise, it sends an error message ('n', see Figure 5-1).

The programming signal level must be within a limit of the programming voltage window for successful programming. The calibration of the signal level is explained in Section 3.0.

2.1 Power-up, Gap, and Verification Signals

The programming signal starts with a power-up signal for 80 ~ 180 μS, followed by a gap signal (0 volt) for 50 ~ 100 μS. The purpose of these signals is to check whether the device is blank and establish a programming mode in the device. Once the device recognizes the power-up signal, it transmits back the contents of its memory. If the device transmits back with the blank bitstream (FSK with all '1's), it is ready to be programmed. If the device is not blank, the programmer informs the host computer that it is nonprogrammable.

If the power-up signal level is out of the programming voltage range, the device will transmit back a non-modulated signal (no data). The nonmodulated signal has no variation in the amplitude (constant voltage signal). A variable resistor, R5 in the microID programmer, should be adjusted to provide a proper power-up signal level. A typical signal level is about 22 ± 3 V_{PP} across the tag coil. This calibration procedure is described in Section 3.0.

2.2 Programming Sequence

Once the device has been verified blank for programming, the programmer sends a programming sequence to the device. The programming data entered in the RFLAB software is sent to the device via the programmer. The programming signal waveforms are shown in Figure 2-1. One bit of data is represented by 128 cycles of the carrier signal. It takes 131.072 ms to complete one programming cycle for the total of 128 data bits. An MCRF200 configured for 128 bits uses all bits in the transfer; an MCRF200 configured for 96 bits ignores bits 33 through 64, although they are present in the programming sequence. After the programming sequence, the device transmits back a verification bitstream. The programmer reports to the host computer the status of the programming.

The data is programmed only if the programming signal level is within the limit in the programming voltage requirement of the device. It takes several programming/verify cycles to completely program each bit of the MCRF200. The microID programmer uses ten (10) blind program/verify cycles before checking the final verify sequence for correct programming. Faster programmers can be designed by checking each program/verify cycle; after approximately 3 ~ 5 cycles, the device will verify correctly. Once a correct verify sequence is received, one additional program cycle should be run to ensure proper programming margin.

nd Verification

s with a power-up signal
a gap signal (0 volt) for
hese signals is to check
nd establish a program-
e the device recognizes
s back the contents of its
its back with the blank
it is ready to be pro-
blank, the programmer
t is nonprogrammable.

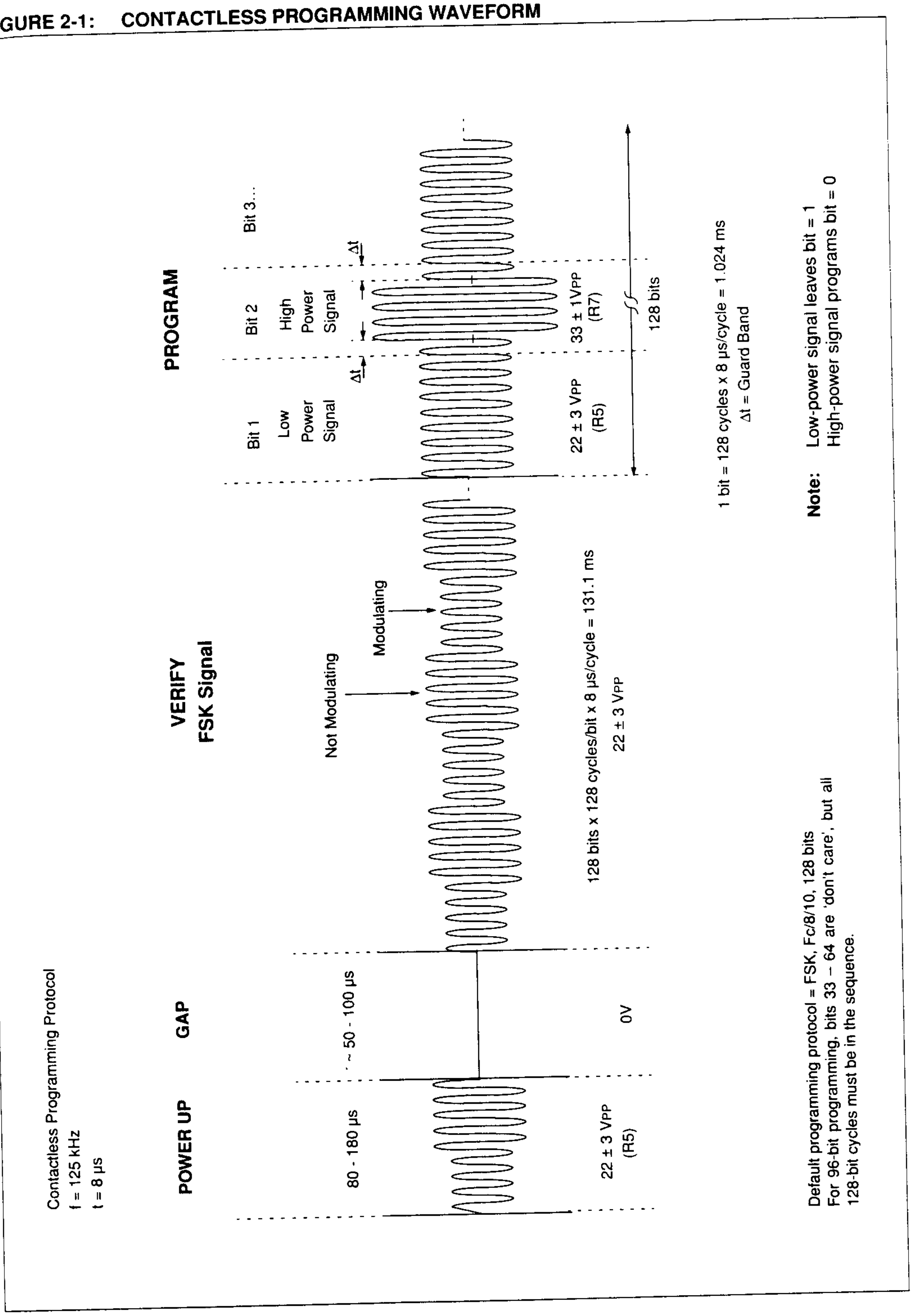
out of the programming
will transmit back a
ta). The nonmodulated
e amplitude (constant
istor, R5 in the microID
ed to provide a proper
al signal level is about
coil. This calibration
on 3.0.

uence

fied blank for program-
programming sequence
ng data entered in the
the device via the
signal waveforms are
data is represented by
It takes 131.072 ms to
cle for the total of 128
ed for 128 bits uses all
) configured for 96 bits
ugh they are present in
fter the programming
s back a verification
reports to the host
amming.

e programming signal
programming voltage
It takes several
pletely program each
programmer uses ten
before checking the
programming. Faster
by checking each
imately 3 ~ 5 cycles,
Once a correct verify
ional program cycle
ogramming margin.

FIGURE 2-1: CONTACTLESS PROGRAMMING WAVEFORM



3.0 CALIBRATION OF PROGRAMMING VOLTAGE

If you are using your own tag coil (with resonant capacitor) with the MCRF200 or MCRF250, you may need to calibrate the programmer for your circuit. Follow this procedure, if you are unable to program your tag.

a) Open the programmer, and turn R5 and R6 full counter-clockwise. Remove the four screws at the back of the programmer.

b) Set up the programmer and calibration tag as shown in Figure 3-1.

Set Up:

- Connect the +9 VDC power supply to the programmer.
- Connect the RS-232 cable from the external serial port in the programmer box to a COM port in the host.
- Open up the RFLAB software on the host computer.
- Place the calibration tag in the center of the tag area on the programmer. A calibration tag is any tag using MCRF200 or MCRF250 silicon and your own coil and capacitor.

c) Run the programming software (RFLAB).

Power-up Signal Level:

d) Click the **Blank Check** button in the RFLAB software.

If the device is blank, a green bar appears in the window with a message indicating that it is blank. If the device is not blank or the power-up signal is out of range, a red bar appears in the window with an error message indicating that it is not blank. The variable resistor (R5) in the programmer should be adjusted to provide a proper "low-power" voltage level to the tag coil. A typical signal level is about 22 ± 3 VPP at the tag coil, but it can vary outside of this range.

R5: Turn clockwise in 1/16-inch increments

Repeat step (d) while adjusting R5. Once the device has been verified as a blank, turn it clockwise one more increment. Then move to the next step.

Programming Signal Level:

e) Click on the buttons in RFLAB for the appropriate data type and protocol for your tag.

f) Enter the programming data in the text box.

g) Click the **Program** button. This will send the programming data to the device. A typical signal level for programming is 33± 1 VPP at the tag coil, but can vary outside of this range.

h) After the device has been programmed, it transmits back the programmed data for verification.

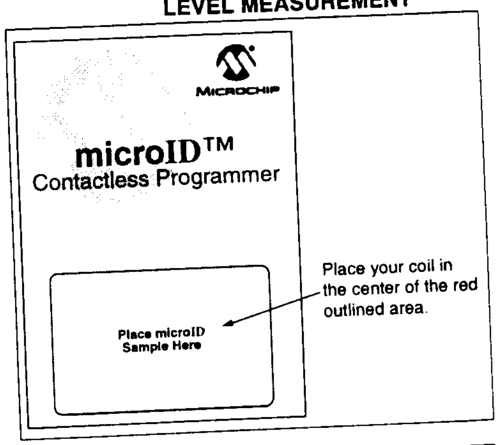
i) If the data has been programmed correctly, a green bar will appear for a few seconds with a message indicating *Programming successful.*

If the programming has been unsuccessful due to insufficient programming signal levels, a message indicating *Programming unsuccessful* will appear in the RFLAB. See Figure 1-1. In this case, R7 ("High Power") must be adjusted to provide a proper programming signal level to the tag coil. Turn R7 clockwise in 1/16-inch increments, repeating steps (f) through (h) until programming is successful. Then turn R7 clockwise one more increment.

Note:	The MCRF200 or MCRF250 lock may be locked even if the programming cycle was unsuccessful; therefore, a new MCRF200 sample may be required for each pass through steps (f) through (h).

j) After programming is completed successfully, keep these R5 and R7 settings for future programming of your tags. Once this calibration has been done, remove the calibration tag from the programmer and reinstall the four screws.

FIGURE 3-1: MCRF2XX microID PROGRAMMER AND CALIBRATION TAG COIL ARRANGEMENT FOR PROGRAMMING SIGNAL LEVEL MEASUREMENT



4.0 PROGRAMMING PROCEDURE

a) Set up the programmer and open up the RFLAB software on the host computer.

Set Up:

- Connect the +9 VDC power supply to the programmer.
- Connect from the external serial port in the programmer box to a COM port in the host computer using the RS-232 cable.

b) Place the RFID device at the center of the programmer.

c) Click **Blank Check** button if you want to check whether the device is blank. This button can also be used to verify that the device is assembled properly.

Note:	The device can't be programmed unless it is blank.

d) Enter the programming data in the RFLAB and select appropriate data type.

e) If several devices are going to be programmed sequentially by any number, click the **Auto Increment** button and specify the increment number.

f) Click the **Program** button. This will send the data to the device.

g) If the data has been programmed correctly, there will be a green bar with a message indicating *Programming successful*.

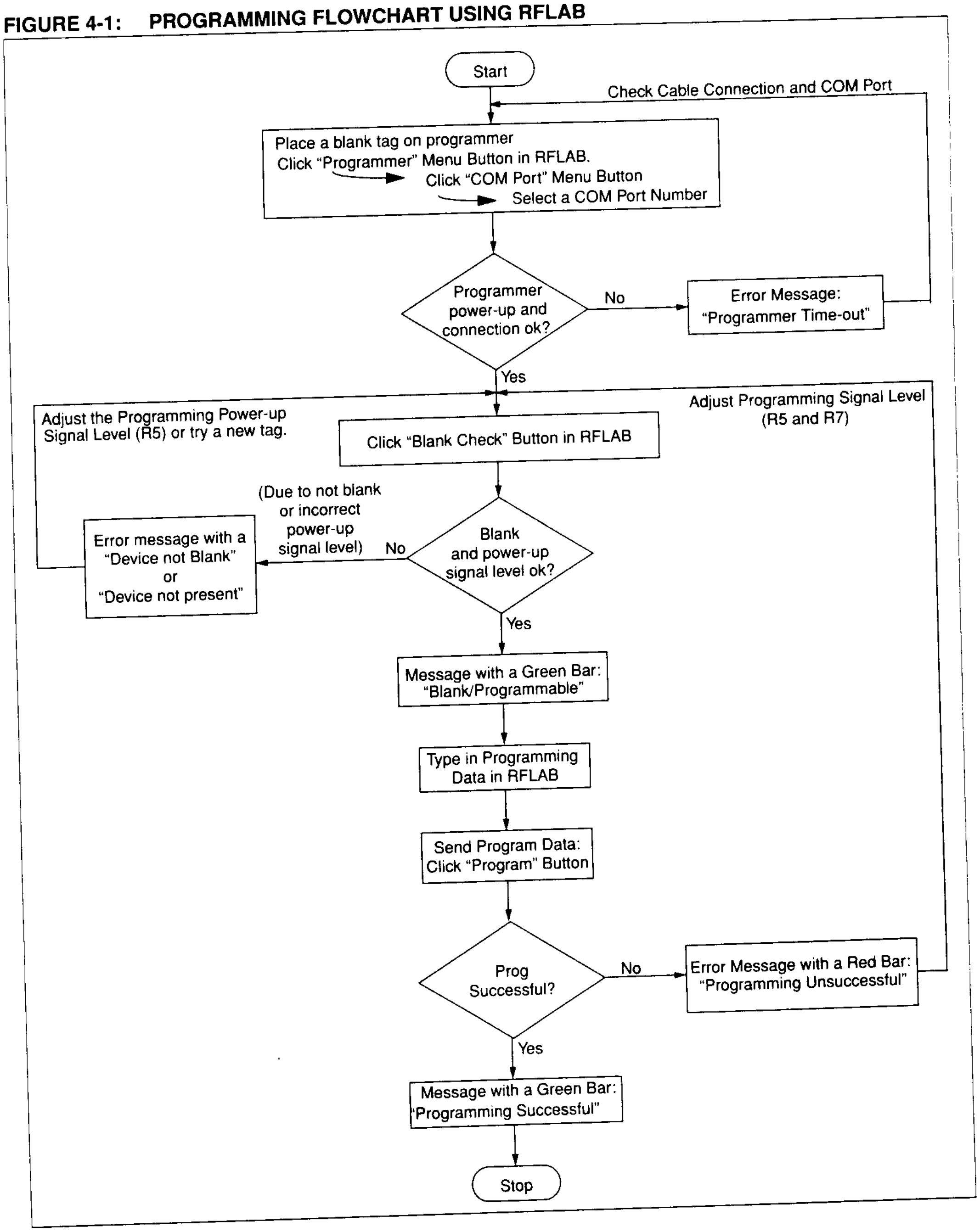
If the programming has been unsuccessful due to out-of-range in the programming signal level, a message and red bar will show up indicating *Programming unsuccessful*. In this case, the programming signal voltage may need to be calibrated for your tag. See the calibration procedure for the programming signal level in the previous section.

h) Repeat step (a) through (g) for other tags.

4.1 Error Conditions

If the host computer does not send programming data to the programmer for more than 3 seconds, the programmer will timeout and reset. If the programmer does not respond to the host computer, there will be an error message indicating *Programmer time out*. If invalid programming data is sent to the programmer during the loading of the program buffer, the programmer will return a message indicating *Invalid*.

FIGURE 4-1: PROGRAMMING FLOWCHART USING RFLAB



5.0 PROGRAMMING IN A STANDARD TERMINAL MODE

In special cases, the device can also be programmed in a standard terminal mode by executing the `terminal.exe` program (c:\windows\terminal.exe) or by any customer production software. The programmer setup, signal waveforms, and calibration procedure are the same as programming with the RFLAB.

The following is a description of how to interface a host computer to Microchip's contactless programmer without the use of RFLAB software. The programmer will check for a blank, unlocked MCRF2XX tag before initiating programming. Once programming has been completed, the programmer will return a pass or fail code. The programmer communicates at 9600 baud, 8 data bits, 1 stop bit, and no parity.

Figure 5-1 shows the programming flow and communication handshakes between host and programmer.

5.1 Programmer Wake-up

Sending an ASCII 'W' (57h) to the programmer on the RS-232 interface will tell the programmer to wake up and be prepared to receive commands. The programmer will reply with ASCII 'R' (52h) when it is ready.

5.2 Blank Check

Sending an ASCII 'T' (54h) will signal the programmer to read the part about being contactlessly programmed and check to see if it is blank (all 1's) and unlocked. If the part is blank and unlocked, the programmer will reply with an ASCII 'Y' (59h) to signify programming should continue. If the part is not blank or not unlocked, the programmer will reply with an ASCII 'N' (4Eh) to indicate an error. It is always necessary to perform a blank check before programming MCRF2XX devices.

5.2.1 SENDING DATA TO THE PROGRAMMER

If the programmer responds with an ASCII 'Y', indicating that the part is blank, the PC can begin passing the 16 bytes of required data to the programmer data buffer. AnMCRF200 configured for 128 bits uses all 16 bytes of data in the transfer; when programming a 96-bit device, however, bits 33 through 64 are 'don't care' and are ignored by the MCRF200. The data should be passed in ASCII equivalent hex bytes and the programmer will acknowledge the receipt of each byte by echoing back what it has received. For example, to program 05 hex data into the first byte, the PC would send ASCII '0' (30h), the programmer would echo '0' back. Next, the programmer would send ASCII '5' (35h), and the programmer will echo back '5'. All of the data must be sent in UPPERCASE ASCII equivalent only. See Figure 5-1 for a typical programming sequence.

5.3 Program and Verify the Device

After 16 bytes of data have been received by the programmer, it is ready to begin programming the data buffer into the MCRF2XX. Sending an ASCII 'V' (56h) will tell the programmer to program the 16 bytes it has received and verify that the device has programmed properly. When the device programs properly, the programmer replies with ASCII 'y' (79h). If the programming was not successful, the programmer replies with ASCII 'n' (6Eh). A successful programming operation should take about 3 to 4 seconds per device.

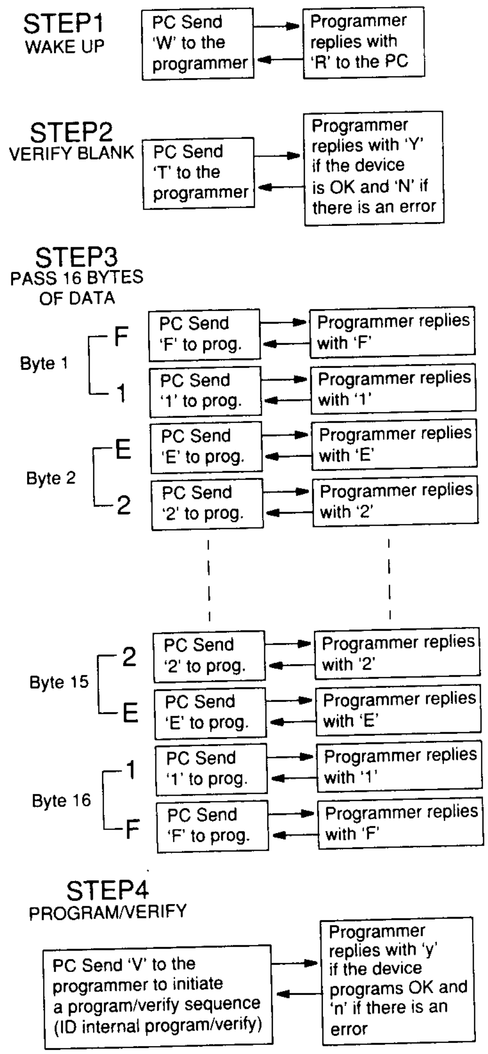
5.4 Error Conditions

If the PC does not send a byte to the programmer for more than 3 seconds, the programmer will timeout and reset. The entire programming sequence will need to be repeated, beginning with the programmer wake-up byte ASCII 'W'.

If invalid bytes are sent to the programmer during the loading of the program buffer, the programmer will return an ASCII 'I' (49h). In this case, the entire programming sequence must be repeated, beginning with the programmer wake-up byte ASCII 'W'.

FIGURE 5-1: TYPICAL SEQUENCE

The following is the programming sequence necessary to wake up the programmer, check if a MCRF2XX part is blank, unlocked and ready to be programmed, send F1E2D3C4B5A6978888796A5B4C3D2E1F ASCII data to the programmer, and instruct the programmer to program and verify the device.

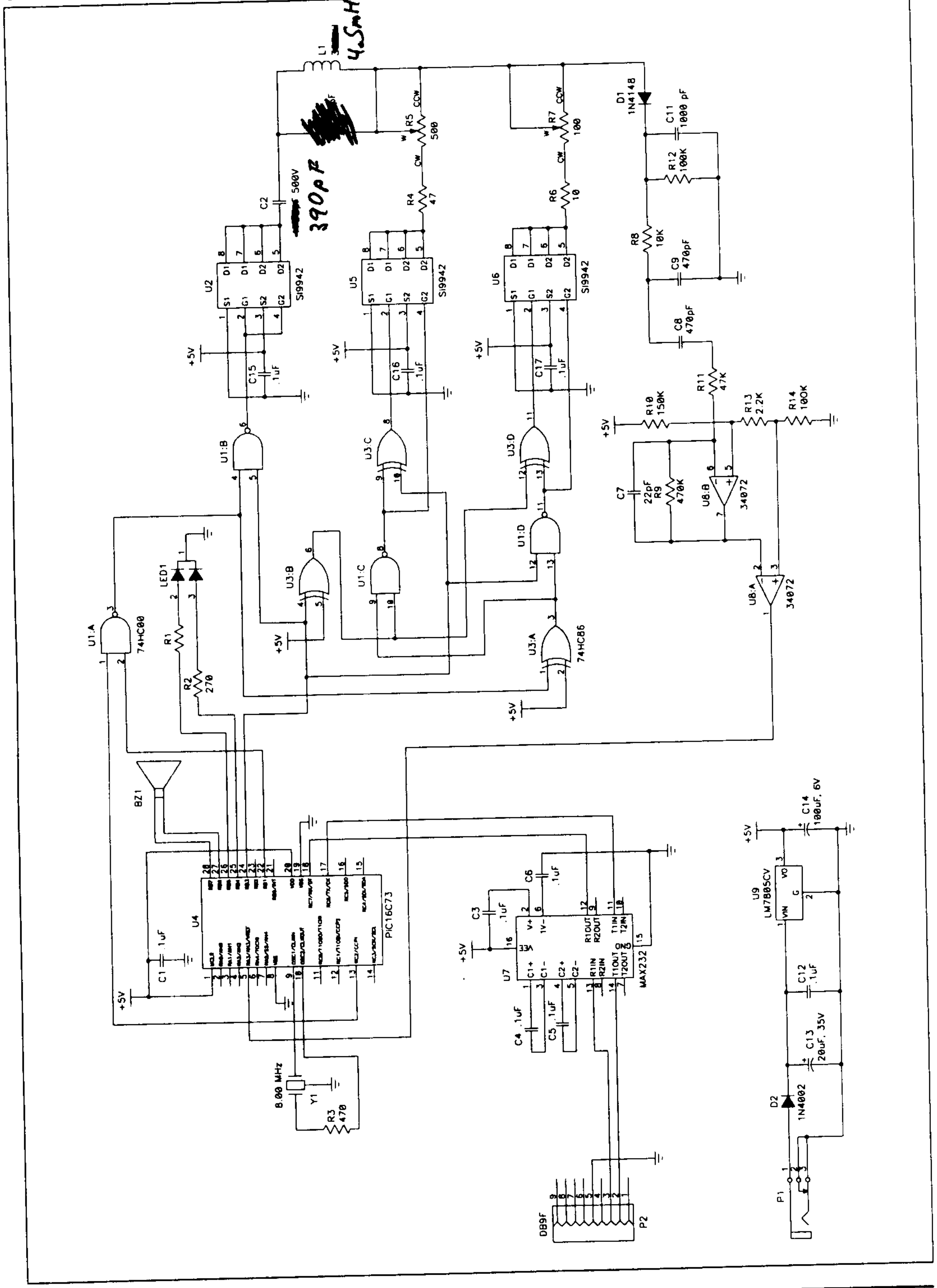


Note: See the signal waveforms and calibration procedure in Sections 2.0 and 3.0.

TABLE 5-1 ASCII CHARACTER SET

		Most Significant Characters							
	Hex	0	1	2	3	4	5	6	7
Least Significant Characters	0	NUL	DLE	Space	0	@	P	`	p
	1	SOH	DC1	!	1	A	Q	a	q
	2	STX	DC2	"	2	B	R	b	r
	3	ETX	DC3	#	3	C	S	c	s
	4	EOT	DC4	$	4	D	T	d	t
	5	ENQ	NAK	%	5	E	U	e	u
	6	ACK	SYN	&	6	F	V	f	v
	7	Bell	ETB	'	7	G	W	g	w
	8	BS	CAN	(	8	H	X	h	x
	9	HT	EM	)	9	I	Y	i	y
	A	LF	SUB	*	:	J	Z	j	z
	B	VT	ESC	+	;	K	[	k	{
	C	FF	FS	,	<	L	\	l	|
	D	CR	GS	-	=	M	]	m	}
	E	SO	RS	.	>	N	^	n	~
	F	SI	US	/	?	O	_	o	DEL

6.0 microID™ PROGRAMMER SCHEMATIC



7.0 microID™ PROGRAMMER BILL OF MATERIALS

Item #	Qty	Part #	Reference Designator	Part Description	Manufacturer	Vendor	Vendor Part #
1	1	ICA-328-S-ST	U4	SOCKET, 28 PIN,.300, MACHINED COLLET	SAMTEC		
2	1	-SPARE-	SP1, LED1, R1, R2	-SPARE- LOCATION DO NOT INSTALL			
3	1	PCC220CNCT-ND	C7	CAP SMT, 22 pF NPO 0805	PANASONIC		
4	2	0805N471J101NT	C8, C9	CAP SMT, 470 pF 5% 100V 0805	MALLORY		
5	1	CD15FC561JO3	C18	CAP, 560 pF, MICA, DIPPED, 300V, AX (0.234LS)	CORNELL DUBILIER	MOUSER	5982-15-300V560
6	1	ECU-V1H102JCX	C11	CAP SMT, 1000 pF 50V NPO CER, 0805	PANASONIC		
7	1	CD19FD472JO3	C2	CAP, 4700 pF, MICA, DIPPED, 500V, AX (0.344LS)	CORNELL DUBILIER	MOUSER	5982-19-500V4700
8	9	250R18Z104MV4 E-6	C1, C3-C6, C12, C15-C17	CAP SMT, 0.1 µF 20% 50V 0805	JOHANSON	NEWARK	50F3674
9	1	ECS-H1ED106R	C13	CAP SMT, 10 µF, TANT ELEC, 25V, 7343	PANASONIC	DIGIKEY	PCT5106CT-ND
10	1	ECE-V0JA101SP	C14	CAP SMT, 100 µF, TANT ELEC, 6.3V, (VS-D)	PANASONIC	DIGIKEY	PCE3058CT-ND
11	1	LL4148	D1	DIODE SMT, 5uA, 100V, 500 mW, FAST SWITCHING, DL-35	DIODES INC	DIGIKEY	LL4148DITR-ND
12	1	DL4002	D2	DIODE SMT, RECTIFIER, 1N4002, 1A, 100V, DL-41	DIODES INC.	DIGIKEY	DL4002DITR-ND
13	1	3345P-1-101	R7	RES, POT, 100 OHM 1/2 RD WW ST SL	BOURNS	DIGIKEY	3345P-101-ND
14	1	3345P-1-501	R5	RES, POT, 500 OHM 1/2 RD WW ST SL	BOURNS	DIGIKEY	3345P-501-ND
15	1	ERJ-6GEYJ100	R6	RES SMT, 10 OHM 1/10W 5% TYPE 0805	PANASONIC		P10ACT-ND
16	1	ERJ-6GEYJ470V	R4	RES SMT, 47 OHM 1/10W 5% TYPE 0805	PANASONIC	DIGIKEY	P470ATR-ND
17	1	ERJ-6GEYJ471V	R3	RES SMT, 470 OHM 1/10W 5% TYPE 0805	PANASONIC		P470ATR-ND
18	1	ERJ-6GEYJ222V	R13	RES SMT, 2.2K OHM 1/10W 5% TYPE 0805	PANASONIC		P2.2KATR-ND
19	1	ERJ-6GEYJ103V	R8	RES SMT, 10K 1/8W 5% TYPE 0805	PANASONIC	DIGIKEY	P10KATR-ND

Item #	Qty	Part #	Reference Designator	Part Description	Manufacturer	Vendor	Vendor Part #
20	1	ERJ-6GEYJ473V	R11	RES SMT, 47K OHM 1/10W 5% TYPE 0805	PANASONIC	DIGIKEY	P473ATR-ND
21	2	ERJ-6GEYJ104V	R12, R14	RES SMT, 100K OHM 1/10W 5% TYPE 0805	PANASONIC	DIGIKEY	P100KATR-ND
22	1	ERJ-6GEYJ154V	R10	RES SMT, 150K OHM 1/8W 5% 0805	PANASONIC	DIGIKEY	P150KATR-ND
23	1	ERJ-6GEYJ474V	R9	RES SMT, 470K OHM 1/8W 5% 0805	PANASONIC	DIGIKEY	P470KATR-ND
24	1	MM74HC00M	U1	IC, SMT, 74HC00 QUAD 2 IN NAND (SO-14)	FAIRCHILD SEMICONDUCTOR	DIGIKEY	MM74HC00M-ND
25	3	NDS9942	U2, U5, U6	IC, SMT, 9942 MOS-FET N-CH & P-CH 20V (SO-8)	FAIRCHILD SEMICONDUCTOR	DIGIKEY	NDS9942TR-ND
26	1	MM74HC86MX	U3	IC, SMT, 74HC86, QUAD XOR GATE (SO-14)	FAIRCHILD SEMICONDUCTOR	DIGIKEY	
27	1	PIC16C73A /P	U4	IC, PIC16C73A /P, PLASTIC DIP, 28P, 0.300	MICROCHIP		
28	1	MAX232ACSE	U7	IC, MAX232ACSE DUAL RS-232 TRANSMITTER/ RCVR, (SO-16)	MAXIM	DIGIKEY	MAX232ACSE-ND
29	1	MC34072D	U8	IC, DUAL OP AMP, (SO-8)	MOTOROLA		
30	1	L7805CV	U9	IC, REG, +5V, 1.5A, 10%, TO-220	SGS THOMSON	MOUSER	511-L7805CV
31	1	EFO-EC8004A4	Y1	OSC, 8.00 MHz CER RESONATOR W/ CAP 3 PIN	PANASONIC	DIGIKEY	PX800-ND
32	1	MCT0003-000	L1	INDUCTOR, 162 µH	CORNEL DUBILIER		
33	1	DE9S-FRS	P2	CONN, D-SUB 9P RECPT RT ANGLE	SPC TECHNOL-OGY		
34	1	DJ005B	P1	JACK, POWER, 2.5mm DC PC MOUNT	LZR ELECTRONICS		

Instruction to the User

This equipment has been tested and found to comply with the limits for a class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

This equipment has been certified to comply with the limits for a class B computing device, pursuant to FCC Rules. In order to maintain compliance with FCC regulations, shielded cables must be used with this equipment. Operation with non-approved equipment or unshielded cables is likely to result in interference to radio and TV reception. The user is cautioned that changes and modifications made to the equipment without the approval of manufacturer could avoid the user's authority to operate this equipment.