

HCS370 Programming Specifications

1.0 PROGRAMMING THE HCS370

HCS370 devices are programmed using a serial method that differs from previous KEELOQ® encoders. This Serial mode will still allow the HCS370 to be programmed while in the users system (ICSP™). This allows for great design flexibility. This programming specification applies to HCS370 devices in all packages.

1.1 Hardware Requirements

The HCS370 requires one programmable power supply for VDD (4.5V to 5.5V) and a VPP of 9V to 14V. Both supplies should have a minimum resolution of 0.25V.

FIGURE 1-1: HCS370 DIAGRAM

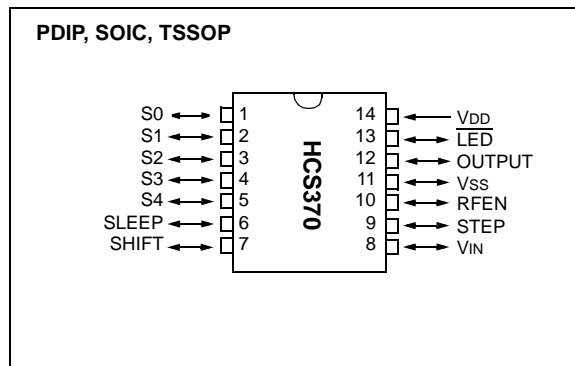


TABLE 1-1: PROGRAMMING PIN DESCRIPTION

Pin Name	Function	HCS370	Pin Type	Pin Description
S0	DATA	1	I/O	Data Input Output
S1	CLOCK	2	I	Clock Input
LED	VPP	13	P(*)	Program mode select
VSS	VSS	11	P	Ground
VDD	VDD	14	P	Power supply

Legend: I = Input, O = Output, P = Power

Note: In the HCS370, the programming high voltage is internally generated. To activate the Programming mode, high voltage needs to be applied to LED input. This is used as a level source, meaning that LED does not draw any significant current.

1.2 Program Mode Entry

Program mode is entered by holding pins S1 and S0 low while raising VPP pin from VIL to VIH. All other pins are don't care. Once in Program mode, the entire encoder memory can be accessed and programmed in a serial fashion. S1 and S0 are Schmitt Trigger inputs in this mode. It is important that VDD comes up first and VPP comes up less than 50 μ s later.

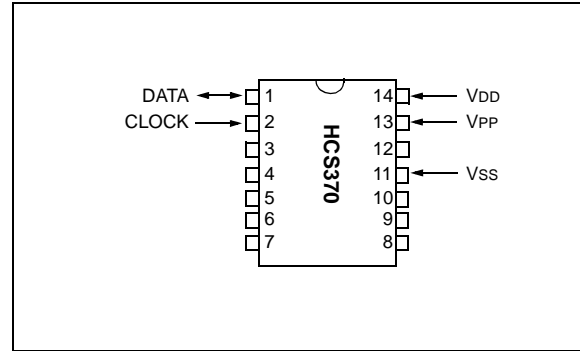
Programming of the device is obtained by writing to two separate memory areas: the encoder memory area and the configuration memory area.

The encoder memory consists of 64 bytes of EEPROM memory and contains the encoder specific data and user configuration options that determine the operation of the device as a KEELOQ encoder. Transmission baud rates, modulation formats and low voltage thresholds are examples of such encoder options.

The configuration memory contains important configuration bits affecting the basic functionality of the device, most of which are factory set and marked as Reserved. The only exception being the Encoder Protect Enable bit, that controls access to the encoder memory area.

In Program mode, all device memory is addressed by means of a single 10-bit Program Counter (PC) that upon entry is set to 3FFh.

FIGURE 1-2: PIN DIAGRAM



1.3 Encoder Memory Map

The encoder memory space is 8-bits wide and extends from address 00h to 7Fh (256 bytes) of which only the first block of 64 bytes is physically implemented from 00h to 3Fh.

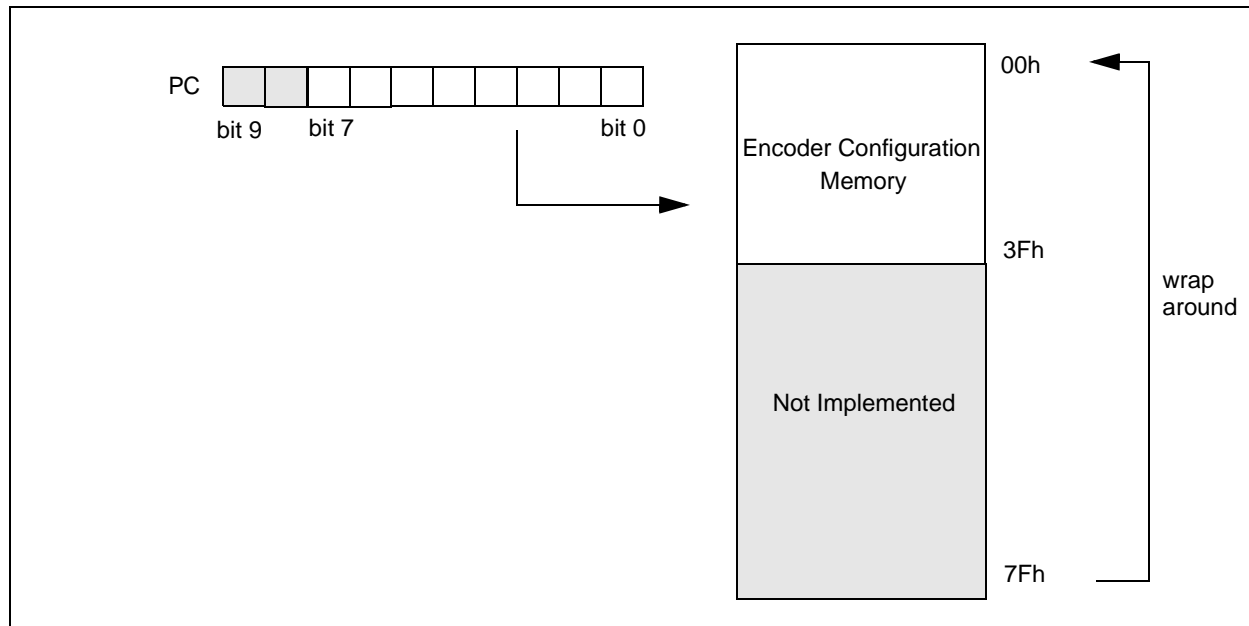
In Program mode this memory is addressed by means of the 8 LSB of the Program Counter (PC), the upper 2 bits of it being ignored.

The non-implemented portion of the addressing space reads always as 0.

Immediately after Program mode entry PC=3FFh, therefore, the Encoder memory address pointed to corresponds to location 0FFh, which is non-implemented.

Note: An Increment Address command is required to move the Program Counter to location 00h.

FIGURE 1-3: ENCODER MEMORY MAPPING



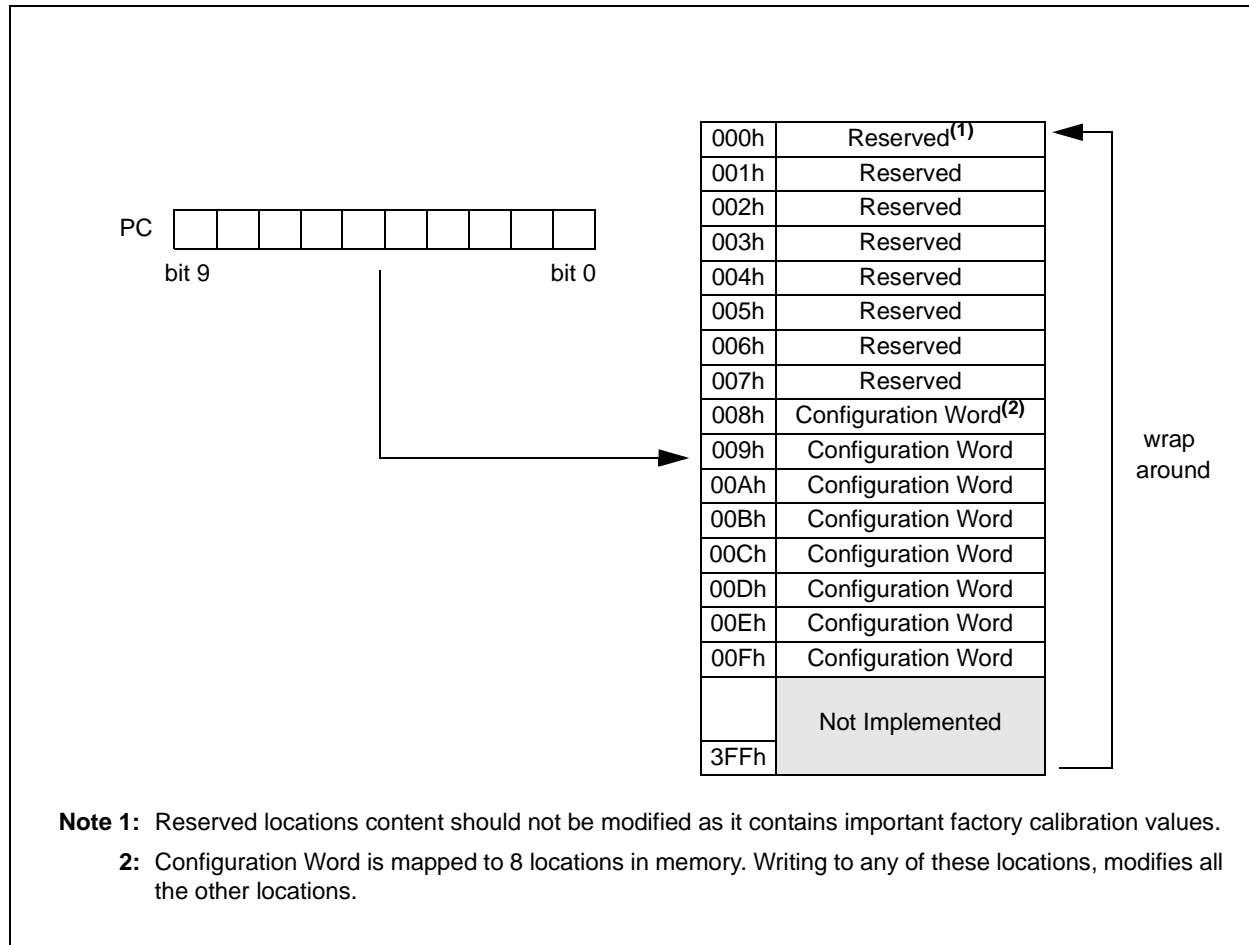
1.4 Configuration Memory Mapping

The device configuration memory space is 12 bits wide and extends from 000h to 3FFh (1 Kbytes). Only the first part, containing 16 words from 000h to 00Fh, is physically implemented and available to the user. The remaining non-implemented portion of the addressing space reads always as 0.

Immediately after Program mode entry PC=3FFh, therefore, the configuration memory address pointed to corresponds to a non-implemented location.

Note: A Load Configuration Memory command followed by an Increment Address command is required to move the Program Counter to location 000h.

FIGURE 1-4: CONFIGURATION MEMORY MAPPING

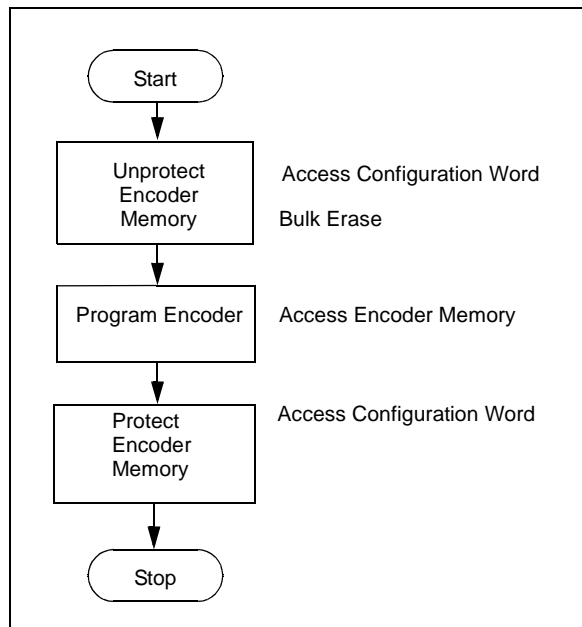


1.5 Programming Method

Programming of the device consists of three basic steps:

- The Configuration Word must be accessed to unprotect the encoder memory, turning the Encoder Protect bit from 0 to 1.
- As a side effect, turning the Encoder Protect bit from 0 back to 1, the encoder memory will be bulk erased (all locations set to 0FFh).
- The encoder memory can now be written with the new encoder data.
- Finally, the Configuration Word is accessed again to protect the encoder memory array.

FIGURE 1-1: PROGRAMMING METHOD



It must be noted that if at the beginning the Encoder Protect bit is not RESET (1), later it will not be possible to modify the contents of the encoder memory.

Failing to set the Encoder Protect bit (0), after writing the new encoder data, on the contrary, will expose the encoder memory contents, including the crypto keys assigned to the device with an extremely severe impact on the security of the customer application.

1.6 The programming sequence

Both when accessing the configuration memory and the encoder memory, a simple basic sequence of four commands is used:

1. Load Data command, transfers the data to the write latch ready for the following command
2. Begin Programming Cycle command, initiates the self timed memory write sequence
3. Read command, verifies the outcome of the previous command
4. Increment Address command, so the Program Counter points to the next memory location

This sequence is repeated for the required number of consecutive locations.

In the following sections we will analyze in detail the implementation of such sequence for access to the two memory areas.

1.6.1 CONFIGURATION MEMORY ACCESS

Use the “Load Data For Configuration Memory” command followed by an Increment Address command, immediately after the Program Mode Entry sequence (see Section 1.4). This will set the PC to 000h, eight subsequent “Increment Address” commands will advance into the Configuration Word memory area (008h to 00Fh).

Continue using the “Load Data For Configuration Memory” command in step 1 of the programming sequence.

Use the Read Data From Configuration Memory command in step 3 of the programming sequence.

1.6.2 ENCODER MEMORY ACCESS

Use the “Increment Address” command immediately after the Program mode entry sequence. This will set the Program Counter to 00h where the actual Encoder programming sequence can start. (see Section 1.3)

Use the “Load Data For Encoder Memory” command in step 1 of the programming sequence.

Use the “Read Data From Encoder Memory” command in step 3 of the programming sequence.

1.6.3 RESERVED LOCATIONS

It is important that the programmer does not modify the content of any of the reserved locations as this might affect some of the factory programmed calibration parameters of the device such as the brown-out threshold levels or the internal RC oscillator frequency.

1.7 Serial Programming Operation

The S1 pin is used as a clock input pin and the S0 pin is used for entering command bits and data input/output during serial operation. To input a command, the clock pin (S1) is cycled 6 times. Each command bit is latched on the falling edge of the clock with the Least Significant bit (LSb) of the command being input first. The data on pin S0 is required to have a minimum setup and hold time of 100 ns, with respect to the falling edge of the clock. Commands that have data associated with them (read and load) are specified to have a minimum delay of 1 μ s, between the command and the data. After this delay, the clock pin is cycled 16 times,

with the first cycle being a START bit, followed by 12 data bits, two zeros and the last cycle being a STOP bit. START and STOP bits are also set to zero.

Data is input and output LSb first. Data is set on the rising edge of the clock and latched on the falling edge of the clock. Therefore, during a read operation, the LSb will be transmitted onto pin S0 on the rising edge of the second cycle, and during a load operation, the LSb will be latched on the falling edge of the second cycle. To allow for decoding of commands and reversal of data pin configuration, a time separation, of at least 1 μ s, is required between a command and a data word (or another command).

FIGURE 1-5: SERIAL COMMAND EXAMPLE

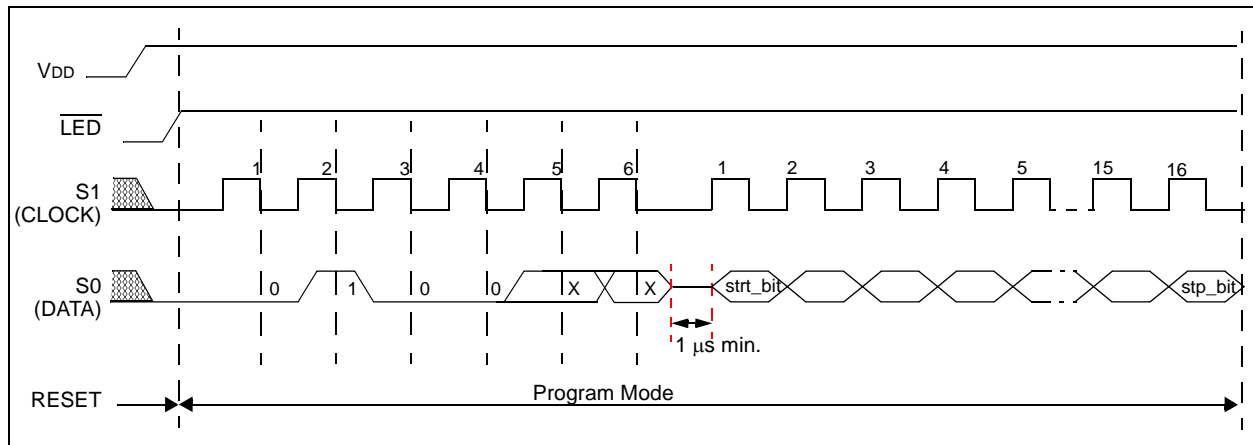


TABLE 1-2: SERIAL PROGRAMMING COMMAND SET

Command	(MSb ... LSb)	Data (order of input/output) START bit, data LSb ... data MSb, padding, STOP bit
Load Data For Configuration Memory	0 0 0 0 0 0	0 d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 0 0 0
Load Data For Encoder Memory	0 0 0 0 1 1	0 d0 d1 d2 d3 d4 d5 d6 d7 0 0 0 0 0 0 0
Read Data From Configuration Memory	0 0 0 1 0 0	0 d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 0 0 0
Read Data From Encoder Memory	0 0 0 1 0 1	0 d0 d1 d2 d3 d4 d5 d6 d7 0 0 0 0 0 0 0
Increment Address	0 0 0 1 1 0	
Begin Erase / Programming Cycle	0 0 1 0 0 0	
Begin Programming Only Cycle	0 0 1 0 1 0	
Bulk Erase Encoder Memory	0 0 1 0 1 1	

Note: All other 6-bit combinations are Reserved.

1.7.1 LOAD DATA FOR CONFIGURATION MEMORY

After receiving this command, the program counter (PC) will be set to 3FFh. An "Increment Address" command will then advance the PC to 000h pointing to the Configuration Word memory. By then applying 16 cycles to the clock pin, the chip will load 12 bits in, as the "data word".

1.7.2 LOAD DATA FOR ENCODER MEMORY

After receiving this command, the chip will load in 12 bits as a "data word" when 16 cycles are applied. However, the encoder memory is only 8-bits wide, and thus, only the first 8 bits of data after the START bit will be programmed into the data memory. It is still necessary to cycle the clock the full 16 cycles in order to allow the internal circuitry to RESET properly.

The PC is used to address the encoder memory. Only the lower 8 bits of the PC are decoded by the encoder memory. Locations 0h-3Fh are physically implemented, while locations 40h-FFh are not implemented and read as zeroes (low). If the PC is greater than FFh, it will wrap around and address a location from 0h-FFh.

1.7.3 READ DATA FROM CONFIGURATION MEMORY

After receiving this command, the chip will transmit data bits out of the configuration memory currently accessed, starting with the second rising edge of the clock input. The S0 pin will go into Output mode on the second rising clock edge, and it will revert back to Input mode (hi-impedance) after the 16th rising edge.

Configuration Word memory will read irrespective of the Encoder Protect enable bit status.

1.7.4 READ DATA FROM ENCODER MEMORY

After receiving this command, the chip will transmit data bits out of the data memory starting with the second rising edge of the clock input. The S0 pin will go into Output mode on the second rising clock edge, and it will revert back to Input mode (hi-impedance) after the 16th rising edge. As previously stated, the data memory is 8-bits wide, and therefore, only the first 8 bits that are output are actual data. A timing diagram for this read command is shown in Figure 5-2.

If the Encoder Protect mechanism is enabled, all the Encoder configuration data will read as zeros. Writing to the encoder memory will also be disabled.

1.7.5 INCREMENT ADDRESS

The PC is incremented when this command is received. A timing diagram of this command is shown in Figure 5-3.

1.7.6 BEGIN ERASE/PROGRAMMING CYCLE

A Load command must be given before every Begin Programming command. Word erasure and programming of Configuration Word and encoder memory will begin, after this command is received and decoded. An internal timing mechanism executes an erase before write. The user must allow the combined time for erase and programming, as specified in the electrical specs, for programming to complete. No "End Programming" command is required.

1.7.7 BEGIN PROGRAMMING ONLY CYCLE

This command is similar to the Erase/Programming Cycle command, except that a word erase is not done. Programming Configuration Word and encoder memory will begin after this command is received and decoded. The user must allow the time for programming, as specified in the electrical specs, for programming to complete. No "End Programming" command is required.

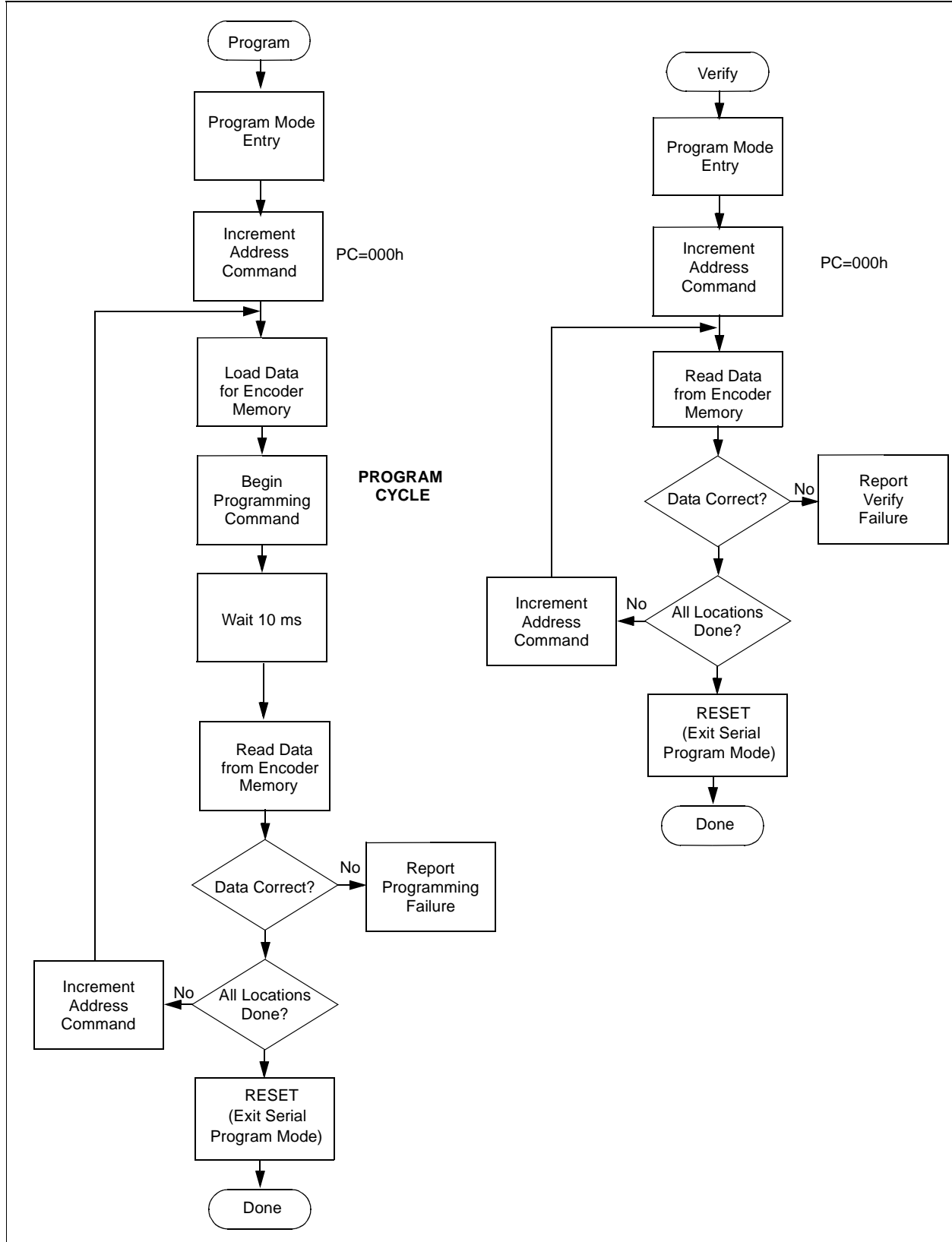
It is recommended that the user do a bulk erase before starting a series of programming only cycles.

1.7.8 BULK ERASE ENCODER MEMORY

After this command is performed, the "Next Program" command will erase the entire encoder memory. The erase time is specified to be 10 ms.

All bulk erase operations must take place at 4.5 to 5.5V VDD range.

FIGURE 1-6: PROGRAM/VERIFY FLOW CHART - ENCODER MEMORY



2.0 CONFIGURATION WORD

Special care must be taken when modifying the Configuration Word contents of the HCS370. In fact, in addition to the Encoder Protection enable bits, it contains several "factory calibration" parameters that are vital to the proper working of the device (marked Reserved).

Care must be taken not to lose these bits contents when manipulating the Configuration Word.

Writing the Configuration Word requires further consideration of the following constraints:

- writing will be performed **only** if the Encoder Protect bit differs from the current Configuration Word contents
- changing from 0 to 1, the Encoder Protect bit will produce a **bulk erase** of the encoder memory contents

Refer to Figure 2-1 (PROGRAM FLOW CHART - CONFIGURATION WORD) to learn how to access and program the Configuration Word of an HCS370 device.

TABLE 2-1: CONFIGURATION WORD BIT MAP

Res	Res	Res	Res	Res	\overline{EP}	Res	Res	Res	Res	Res	Res	
bit 11					bit 6		bit 0					

Res: **Reserved**, preserve value of these bits when writing to the Configuration Word

bit 6: \overline{EP} : Encoder Protection enable bit

Setting the bit from 0 to 1 removes read protection and bulk erases the Encoder EEPROM

1 = Not protected

0 = Protected - Can not read or write

2.1 Encoder Protection

The HCS370 features an Encoder Protection mechanism to control access to the encoder memory array. When Encoder Protection is enabled, the encoder memory locations read all '0's. Further programming is disabled for the entire array.

Once Encoder Protection has been enabled, disabling it will produce a bulk erase of the entire encoder memory array.

Configuration memory is not affected by the Encoder Protect bits and is always available to read.

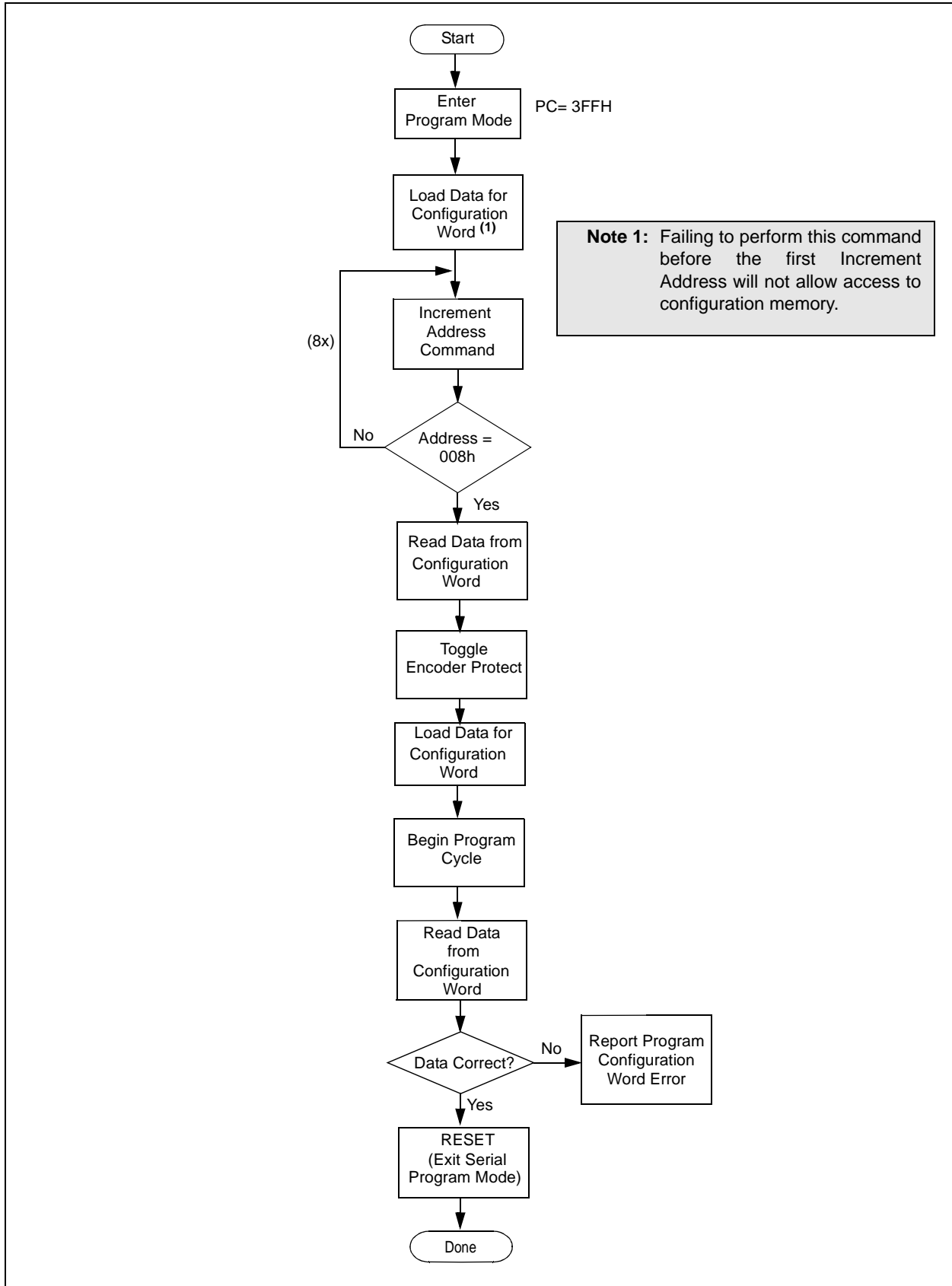
Note: Failing to set the Encoder Protect bit (0), after writing new encoder data will expose the encoder memory contents, including the crypto keys assigned to the device, with an extremely severe impact on the security of the customer application.

TABLE 2-2: CODE PROTECTION

Memory Segment	R/W Protected ($\overline{EP} = 0$)	R/W Unprotected ($\overline{EP} = 1$)
Configuration Memory [000h:00Fh]	Read Enabled, Write Enabled*	Read Enabled, Write Enabled*
Encoder memory [000h:03Fh]	Read Disabled (all 0's), Write Disabled	Read Enabled, Write Enabled

Note*: The Configuration Word is not updated unless EP is changed from existing value.

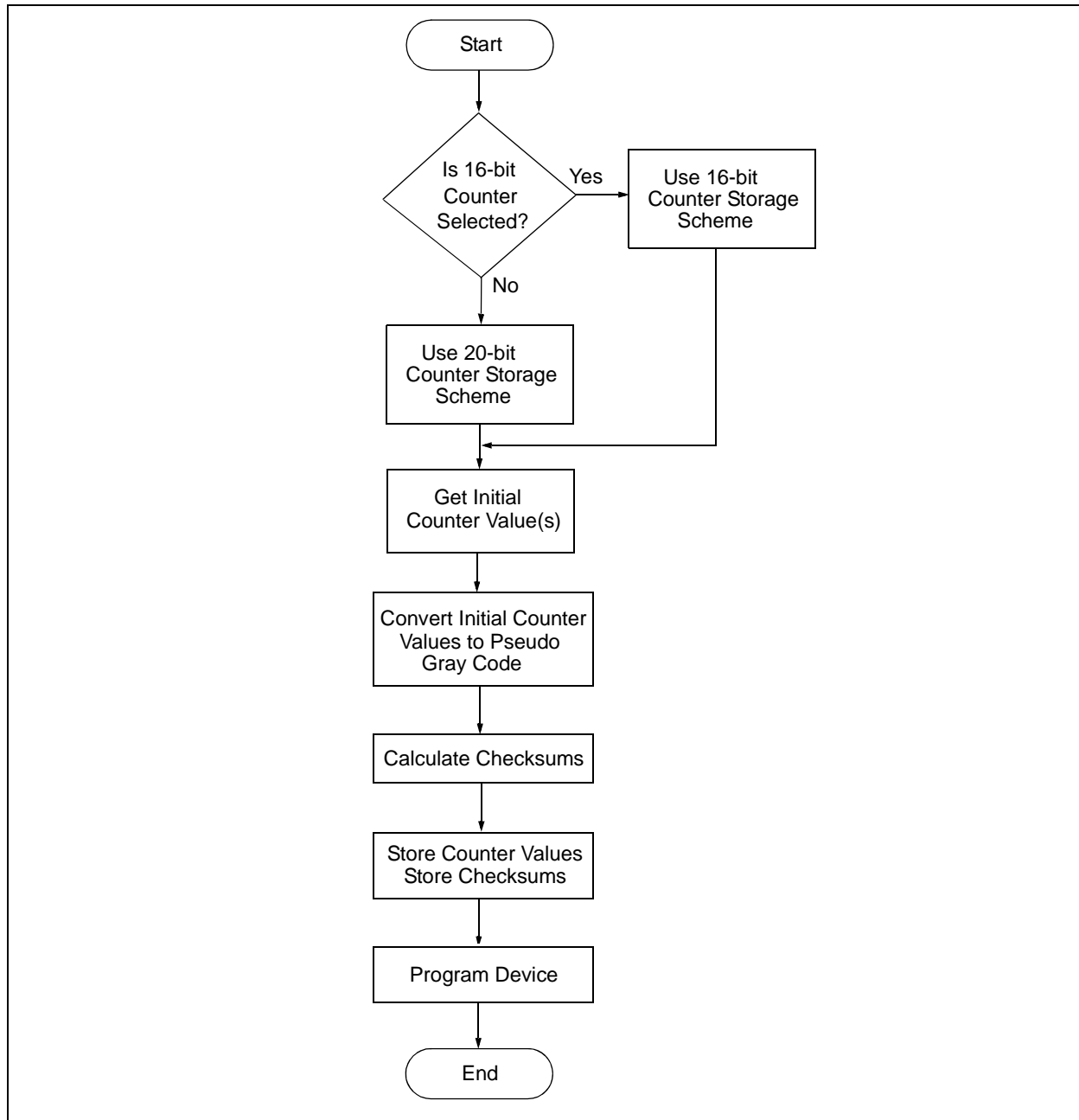
FIGURE 2-1: PROGRAM FLOW CHART - CONFIGURATION WORD



2.2 HCS370 Counter Implementation

The HCS370 uses a different method of storing the counter in EEPROM when compared to previous KEELOQ encoders. This section explains how to implement this different counter storage method and provides some examples for testing purposes.

FIGURE 2-1: COUNTER IMPLEMENTATION FLOWCHART



20-Bit Counter:

Reg	Bits							
A	23	22	21	20	19	18	17	16
	x	x	0	0	D	D	D	D

MSb

Reg	Bits							
B	15	14	13	12	11	10	9	8
	D	D	D	D	D	D	D	D

Reg	Bits							
C	7	6	5	4	3	2	1	0
	D	D	D	D	D	D	D	D

LSb

- D represents a counter bit
- x represents a don't care
- 0 represents a logic zero

16-Bit Counter:

Reg	Bits							
A	23	22	21	20	19	18	17	16
	x	x	0	0	0	0	F	F

MSb

Reg	Bits							
B	15	14	13	12	11	10	9	8
	D	D	D	D	D	D	D	D

Reg	Bits							
C	7	6	5	4	3	2	1	0
	D	D	D	D	D	D	D	D

LSb

- D represents a counter bit
- x represents a don't care
- 0 represents a logic zero
- FF represents overflow conditions.
 - FF = 11 -> None
 - FF = 01 -> Once
 - FF = 00 -> Twice

Convert Initial Counter Values to Pseudo Gray Code (C Implementation):

Note: See previous page for references to A, B, and C.

```

If ( B & 1)
    C ^= 0xFF;
If ( A & 1)
    B ^= 0xFF;
    
```

Calculate Checksums (C implementation):

```

Checksum1 = B^C;
Checksum2 = A^C;
Checksum3 = A^B;
    
```

Store Counter Values And Checksums:

Note: Checksum 3 comes before Checksum 2

For HCS370 Transmitter #1

	Bits							
Address	7	6	5	4	3	2	1	0
00	A							
01	B							
02	C							
03	Checksum 1							
04	Checksum 3							
05	Checksum 2							

For HCS370 Transmitter #2

	Bits							
Address	7	6	5	4	3	2	1	0
08	A							
09	B							
0A	C							
0B	Checksum 1							
0C	Checksum 3							
0D	Checksum 2							

Example 1:

Counter Scheme: 16-bit

Counter Value: 0x0000

Overflow Condition: None

Reg	Bits							
A	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	1	1
MSb								
Reg	Bits							
B	15	14	13	12	11	10	9	8
	0	0	0	0	0	0	0	0
Reg	Bits							
C	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
LSb								

Get Initial Counter Values:

A = 0x03

B = 0x00

C = 0x00

Convert Initial Counter Values To Pseudo Gray Code (See page 12):

A = **0x03**

B = **0xFF**

C = **0x00**

Calculate Checksums:

Checksum1= B ⊕ C = 0xFF ⊕ 0x00 = **0xFF**

Checksum2= A ⊕ C = 0x03 ⊕ 0x00 = **0x03**

Checksum3= A ⊕ B = 0x03 ⊕ 0xFF = **0xFC**

Store Counter Values And Checksums:

For HCS370 Transmitter #1:

	Bits							
Address	7	6	5	4	3	2	1	0
00	A = 0x03							
01	B = 0xFF							
02	C = 0x00							
03	Checksum 1 = 0xFF							
04	Checksum 3 = 0xFC							
05	Checksum 2 = 0x03							

HCS370

Example 2:

Counter Scheme: 16-bit

Counter Value: 0x1234

Overflow Condition: Once

Reg	Bits							
A	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	1
MSb								
Reg	Bits							
B	15	14	13	12	11	10	9	8
	0	0	0	1	0	0	1	0
Reg	Bits							
C	7	6	5	4	3	2	1	0
	0	0	1	1	1	0	0	0
LSb								

Get Initial Counter Values:

A = 0x01

B = 0x12

C = 0x34

Convert Initial Counter Values To Pseudo Gray Code (See page 12):

A = **0x01**

B = **0xED**

C = **0x34**

Calculate Checksums:

Checksum1= B ⊕ C = 0xED ⊕ 0x34 = **0xD9**

Checksum2= A ⊕ C = 0x01 ⊕ 0x34 = **0x35**

Checksum3= A ⊕ B = 0x01 ⊕ 0xED = **0xEC**

Store Counter Values And Checksums:

For HCS370 Transmitter #1:

	Bits							
Address	7	6	5	4	3	2	1	0
00	A = 0x01							
01	B = 0xED							
02	C = 0x34							
03	Checksum 1 = 0xD9							
04	Checksum 3 = 0xEC							
05	Checksum 2 = 0x35							

Example 3:

Counter Scheme: 16-bit

Counter Value: 0x789A

Overflow Condition: Twice

Reg	Bits							
A	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0
MSb								
Reg	Bits							
B	15	14	13	12	11	10	9	8
	0	1	1	1	1	0	0	0
Reg	Bits							
C	7	6	5	4	3	2	1	0
	1	0	0	1	1	0	1	0
LSb								

Get Initial Counter Values:

A = 0x00

B = 0x78

C = 0x9A

Convert Initial Counter Values To Pseudo Gray Code (See page 12):

A = **0x00**

B = **0x78**

C = **0x9A**

Calculate Checksums:

Checksum1 = B ⊕ C = 0x78 ⊕ 0x9A = **0xE2**

Checksum2 = A ⊕ C = 0x00 ⊕ 0x9A = **0x9A**

Checksum3 = A ⊕ B = 0x00 ⊕ 0x78 = **0x78**

Store Counter Values And Checksums:

For HCS370 Transmitter #1:

	Bits							
Address	7	6	5	4	3	2	1	0
00	A = 0x00							
01	B = 0x78							
02	C = 0x9A							
03	Checksum 1 = 0xE2							
04	Checksum 3 = 0x78							
05	Checksum 2 = 0x9A							

HCS370

Example 4:

Counter Scheme: 20-Bit

Counter Value: 0xFABCD

Reg	Bits							
A	23	22	21	20	19	18	17	16
	0	0	0	0	1	1	1	1
MSb								
Reg	Bits							
B	15	14	13	12	11	10	9	8
	1	0	1	0	1	0	1	1
Reg	Bits							
C	7	6	5	4	3	2	1	0
	1	1	0	0	1	1	0	1
LSb								

Get Initial Counter Values:

A = 0x0F

B = 0xAB

C = 0xCD

Convert Initial Counter Values To Pseudo Gray Code (See page 12):

A = 0x0F

B = 0x54

C = 0x32

Calculate Checksums:

Checksum1 = B ⊕ C = 0x54 ⊕ 0x32 = **0x66**

Checksum2 = A ⊕ C = 0x0F ⊕ 0x32 = **0x3D**

Checksum3 = A ⊕ B = 0x0F ⊕ 0x54 = **0x5B**

Store Counter Values And Checksums:

For HCS370 Transmitter #1:

	Bits							
Address	7	6	5	4	3	2	1	0
00	A = 0x0F							
01	B = 0x54							
02	C = 0x32							
03	Checksum 1 = 0x66							
04	Checksum 3 = 0x5B							
05	Checksum 2 = 0x3D							

3.0 EEPROM ORGANIZATION

A summary of the HCS370 EEPROM organization is shown in the table below.

TABLE 3-1: HCS370 MEMORY MAP

Bytes	Bits							
	7	6	5	4	3	2	1	0
00	SYNC COUNTER TX#1 USB							
01	SYNC COUNTER TX#1 MSB							
02	SYNC COUNTER TX#1 LSB							
03	SYNC COUNTER TX#1 CHKSUM 1							
04	SYNC COUNTER TX#1 CHKSUM 3							
05	SYNC COUNTER TX#1 CHKSUM 2							
06	RESERVED							
07	RESERVED							
08	SYNC COUNTER TX#2 USB							
09	SYNC COUNTER TX#2 MSB							
0A	SYNC COUNTER TX#2 LSB							
0B	SYNC COUNTER TX#2 CHKSUM 1							
0C	SYNC COUNTER TX#2 CHKSUM 3							
0D	SYNC COUNTER TX#2 CHKSUM 2							
0E	RESERVED							
0F	RESERVED							
10	32-BIT SERIAL NUMBER TX #1 (MSB)							
11	32-BIT SERIAL NUMBER TX #1							
12	32-BIT SERIAL NUMBER TX #1							
13	32-BIT SERIAL NUMBER TX#1 (LSB)							
14	4-BIT SEED CODE TX#1				60-BIT SEED_1 VALUE (MS-NIBBLE)			
15	60-BIT SEED_1 VALUE							
16	60-BIT SEED_1 VALUE							
17	60-BIT SEED_1 VALUE							
18	60-BIT SEED_1 VALUE							
19	60-BIT SEED_1 VALUE							
1A	60-BIT SEED_1 VALUE							
1B	60-BIT SEED_1 VALUE (LSB)							
1C	STEN_1	QUEN_1	XSER_1	HSEL_1	MSEL_1		DISC_1H (MS-BITS)	
1D	DISC_1L (LSB)							
1E	64-BIT KEY_1 (MSB)							
1F	64-BIT KEY_1							
20	64-BIT KEY_1							
21	64-BIT KEY_1							
22	64-BIT KEY_1							
23	64-BIT KEY_1							
24	64-BIT KEY_1							
25	64-BIT KEY_1 (LSB)							
26	32-BIT SERIAL NUMBER TX#2 (MSB)							
27	32-BIT SERIAL NUMBER TX#2							
28	32-BIT SERIAL NUMBER TX#2							
29	32-BIT SERIAL NUMBER TX#2 (LSB)							
2A	4-BIT SEED CODE TX#2				60-BIT SEED_2 VALUE (MS-NIBBLE)			
2B	60-BIT SEED_2 VALUE							
2C	60-BIT SEED_2 VALUE							
2D	60-BIT SEED_2 VALUE							
2E	60-BIT SEED_2 VALUE							
2F	60-BIT SEED_2 VALUE							

HCS370

Bytes	Bits							
	7	6	5	4	3	2	1	0
30	60-BIT SEED_2 VALUE							
31	60-BIT SEED_2 VALUE (LSB)							
32	STEN_2	QUEN_2	XSER_2	HSEL_2	MSEL_2		DISC_2H (MS-BITS)	
33	DISC_2L (LSB)							
34	64-BIT KEY_2 (MSB)							
35	64-BIT KEY_2							
36	64-BIT KEY_2							
37	64-BIT KEY_2							
38	64-BIT KEY_2							
39	64-BIT KEY_2							
3A	64-BIT KEY_2							
3B	64-BIT KEY_2 (LSB)							
3C	GSEL_1		BSEL_1		SDTM_1		SDMD_1	SDLM_1
3D	LEDOS_2	LEDBL_2	TSEL		SLEEP	WAIT	MTX	
3E	GSEL_2		BSEL_2		SDTM_2		SDMD_2	SDLM_2
3F	LEDOS_1	LEDBL_1	PLLSEL	VLOWSEL	VLOWL	CNTSEL	WAKE	

4.0 ENCODER CONFIGURATION OPTIONS SUMMARY

Data stored in the EEPROM can be classified as Encoder configuration (E) or System Configuration (S). In the case of dual Encoder Operation, two copies of the Encoder configuration must be stored for Encoder 1 and Encoder 2.

TABLE 4-1: FIRST ENCODER CONFIGURATION OPTIONS

Symbol	Address Bits	Class	Description (Note 1)		
SYNC_1	00: 16 bits	E	Encoder Synchronization Counter (CNTSEL=1)		
SER_1	10: 32 bits	E	Encoder Serial Number		
SDBT_1	14: 7654----	E	Seed Button Code		
SEED_1	14: 60 bits	E	Encoder Seed Value		
STEN_1	1C: 7-----	E	START/STOP Pulse Enable	Disable = 0	Enable = 1
QUEN_1	1C: -6-----	E	Queue counter Enable	Disable = 0	Enable = 1
XSER_1	1C: --5-----	E	Extended Serial Number	28 bits = 0	32 bits = 1
HSEL_1	1C: ---4----	E	Header Select	4 TE = 0	10 TE = 1
MSEL_1	1C: ----32--	E	Transmission Modulation Format	Value	Format
				00b	PWM
				01b	Manchester
				10b	VPWM
				11b	PPM
DISC_1h	1C: -----10	E	Encoder Discrimination value (2 MSB)		
DISC_1l	1D: 8 bits	E	Encoder Discrimination value (8 LSB)		
KEY_1	1E: 64 bits	E	Encoder Key		
SDLM_1	3C: -----0	E	Limited Seed	Disable = 0	Enable = 1
SDMD_1	3C: -----1-	E	Seed Mode	User = 0	Production = 1
SDTM_1	3C: ----32--	E	Time Before Seed Code Word	Value	Time (s)
				00b	0.0
				01b	0.8
				10b	1.6
				11b	3.2
BSEL_1	3C: --54----	E	Transmission Baud Rate Select	Value	TE (μ s)
				00b	100
				01b	200
				10b	400
				11b	800
GSEL_1	3C: 76-----	E	Guard Time Select	Value	Time (ms)
				00b	0.0
				01b	6.4
				10b	51.2
				11b	102.4
LEDBL_1	3F: -6-----	E	Low Voltage LED Blink	Continuous = 0	Once = 1
LEDOS_1	3F: 7-----	E	LED On Time Select	50 ms = 0	100 ms = 1

Note 1: All Timing values vary $\pm 10\%$.

TABLE 4-2: SECOND ENCODER CONFIGURATION OPTIONS

Symbol	Address Bits	Class	Description (Note 1)		
SYNC_2	08: 20 bits	E	Encoder Synchronization Counter (CNTSEL=1)		
SER_2	26: 32 bits	E	Encoder Serial Number		
SDBT_2	2A: 7654----	E	Seed Button Code		
SEED_2	2A: 60 bits	E	Encoder Seed Value		
STEN_2	32: 7-----	E	START/STOP Pulse Enable	Disable = 0	Enable = 1
QUEN_2	32: -6-----	E	Queue counter Enable	Disable = 0	Enable = 1
XSER_2	32: --5-----	E	Extended Serial Number	28 bits = 0	32 bits = 1
HSEL_2	32: ---4----	E	Header Select	4 TE = 0	10 TE = 1
MSEL_2	32: ----32--	E	Transmission Modulation Format	Value	Format
				00b	PWM
				01b	Manchester
				10b	VPWM
				11b	PPM
DISC_2h	32: -----10	E	Encoder Discrimination value (2 MSB)		
DISC_2l	33: 8 bits	E	Encoder Discrimination value (8 LSB)		
KEY_2	34: 64 bits	E	Encoder Key		
SDLM_2	3E: -----0	E	Limited Seed	Disable = 0	Enable = 1
SDMD_2	3E: -----1-	E	Seed Mode	User = 0	Production = 1
SDTM_2	3E: ----32--	E	Time Before Seed Code Word	Value	Time (s)
				00b	0.0
				01b	0.8
				10b	1.6
				11b	3.2
BSEL_2	3E: --54----	E	Transmission Baud Rate Select	Value	TE (μs)
				00b	100
				01b	200
				10b	400
				11b	800
GSEL_2	3E: 76-----	E	Guard Time Select	Value	Time (ms)
				00b	0.0
				01b	6.4
				10b	51.2
				11b	102.4
LEDBL_2	3D: -6-----	E	Low Voltage LED Blink	Continuous = 0	Once = 1
LEDOS_2	3D: 7-----	E	LED On Time Select	50 ms = 0	100 ms = 1

Note 1: All Timing values vary $\pm 10\%$.

TABLE 4-3: SYSTEM CONFIGURATION OPTIONS

Symbol	Address Bits	Class	Description (Note 1)		
				Value	Value
MTX	3D: -----10	S	Minimum Code Words	Value	Value
				00b	1
				01b	2
				10b	4
				11b	8
WAIT	3D: ----2--	S	Wait for Step Up regulator	Disable = 0	Enable = 1
SLEEP	3D: ----3---	S	SLEEP Output Enable	Disable = 0	Enable = 1
TSEL	3D: --54----	S	Time-out Select	Value	Time (s)
				00b	Disabled
				01b	0.8
				10b	3.2
				11b	25.6
WAKE	3F: -----10	S	Wake-up	Value	Value
				00b	No Wake-up
				01b	75 ms 50%
				10b	50 ms 33.3%
				11b	100 ms 16.6%
CNTSEL	3F: ----2--	S	Counter Select	16 bits = 0	20 bits = 1
VLOWL	3F: ----3---	S	Low Voltage Latch Enable	Disable = 0	Enable = 1
VLOWSEL	3F: ---4----	S	Low Voltage Trip Point Select	2.2 V = 0	3.2V = 1
PLLSEL	3F: --5-----	S	PLL Interface Select	ASK = 0	FSK = 1

Note 1: All Timing values vary $\pm 10\%$.

5.0 PROGRAM MODE ELECTRICAL CHARACTERISTICS

TABLE 5-1: AC/DC TIMING REQUIREMENTS FOR PROGRAM MODE

AC/DC CHARACTERISTICS, POWER SUPPLY PINS						
Characteristics	Sym	Min	Typ	Max	Units	Conditions/Comments
General						
Supply voltage during programming	Vbatt	4.5	5.0	5.5	V	
Supply voltage during verify	Vbatt	VDD min.		VDD max.	V	
High voltage on $\overline{\text{MCLR}}$ Test mode entry	Vihh	$V_{\text{BATT}} + 4.5$		14.0	V	Always use minimum on VIH
Power-up Reset time before entering Test mode	tvdd		50		μs	
$\overline{\text{MCLR}}$ rise time (V_{SS} to V_{HH}) for Test mode entry	tvhr			1.0	ms	
(S1, S0) input high level	Vih1	$0.8V_{\text{BATT}}$			V	Schmitt Trigger input
(S1, S0) input low level	Vil1	$0.2V_{\text{BATT}}$			V	Schmitt Trigger input
S<3:0> setup time before $\overline{\text{MCLR}}$ (Test mode selection pattern setup time)	tset0	100			ns	
S<3:0> hold time after $\overline{\text{MCLR}}$ (Test mode selection pattern setup time)	thld0	5			ms	
Serial Program/Verify						
Data in setup time before clock	tset1	100			ns	
Data in hold time after clock	thld1	100			ns	
Data input not driven to next clock input (delay required between command/data or command/command)	tdly1	1.0			μs	
Delay between clock to clock of next command or data	tdly2	1.0			μs	
Clock to data out valid (during read data)	tdly3	80			ns	
Serial Clock Period		1			μs	
Erase Cycle Time				10	ms	
Program Cycle Time				10	ms	

FIGURE 5-1: LOAD DATA FOR CONFIG MEMORY COMMAND

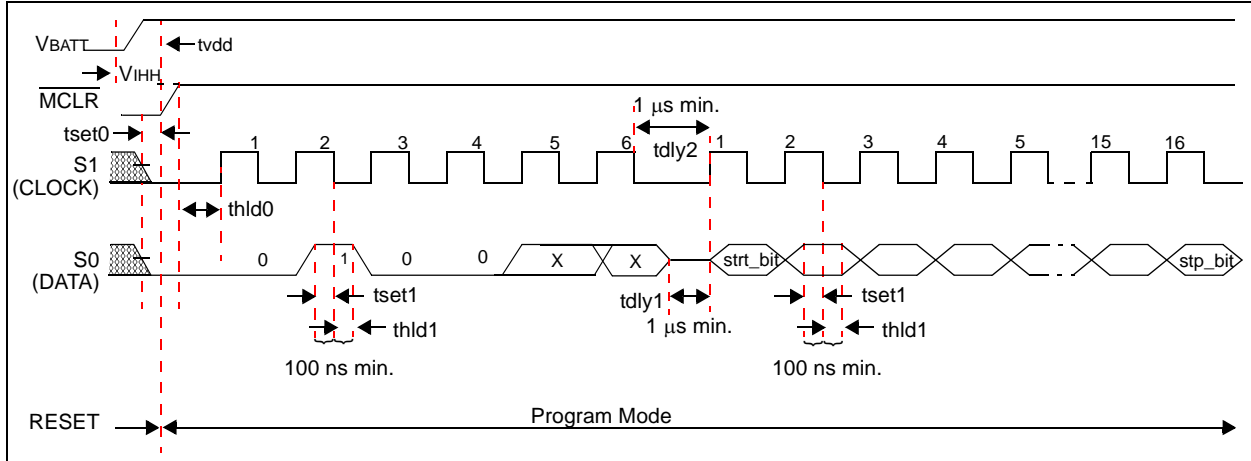


FIGURE 5-2: READ DATA FROM ENCODER MEMORY COMMAND

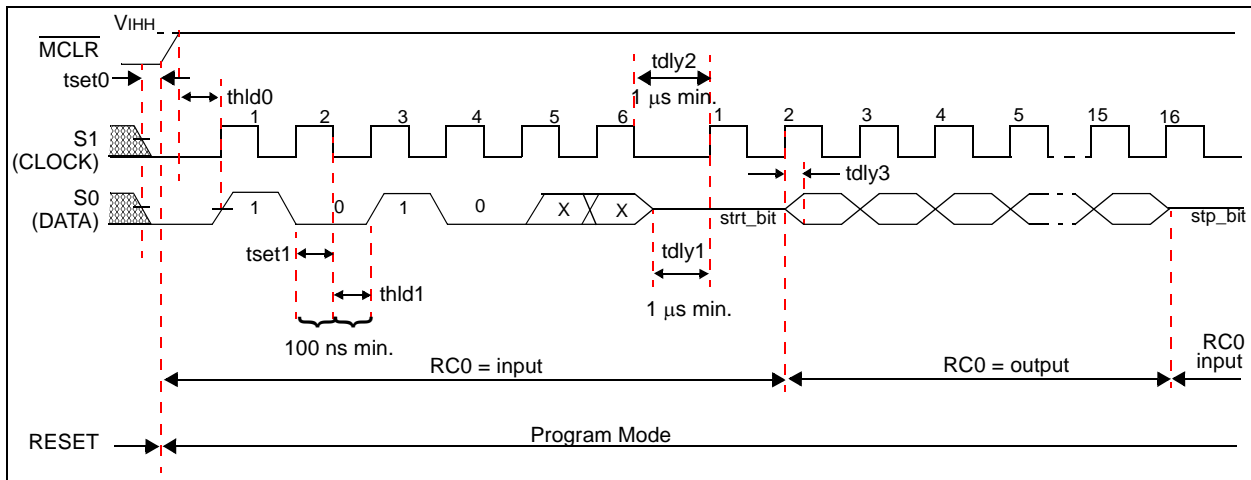
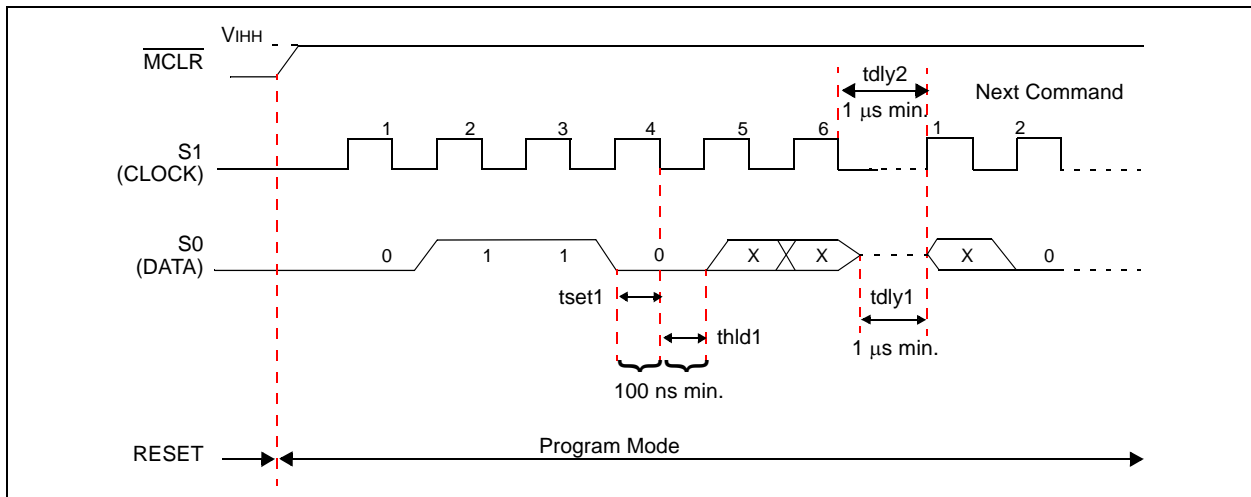


FIGURE 5-3: INCREMENT ADDRESS COMMAND



6.0 ADDITIONAL INFORMATION

Microchip's Secure Data Products are covered by some or all of the following: Code hopping encoder patents issued in European countries and U.S.A. Secure learning patents issued in European countries, U.S.A. and R.S.A.

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC³² logo, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICKit, PICtail, REAL ICE, rfLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2001-2011, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

ISBN: 978-1-61341-210-7

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2009 ==

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC[®] MCUs and dsPIC[®] DSCs, KEELOQ[®] code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.



MICROCHIP

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
<http://www.microchip.com/support>
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3180
Fax: 86-571-2819-3189

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-213-7830
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

05/02/11