

## KEELOQ<sup>®</sup> Code Hopping Encoder

### FEATURES

#### Security

- Programmable 28-bit serial number
- Programmable 64-bit encryption key
- Each transmission is unique
- 66-bit transmission code length
- 32-bit hopping code
- 34-bit fixed code (28-bit serial number, 4-bit function code, 2-bit status)
- Encryption keys are read protected

#### Operating

- 3.5V - 13.0V operation
- Shift key and three inputs
- 16 functions available
- Selectable baud rate
- Automatic code word completion
- Battery low signal transmitted to receiver
- Battery low indication on LED
- Non-volatile synchronization data

#### Other

- Easy-to-use programming interface
- On-chip EEPROM
- On-chip oscillator and timing components
- Button inputs have internal pull-down resistors
- Current limiting on LED output
- Low external component cost

#### Typical Applications

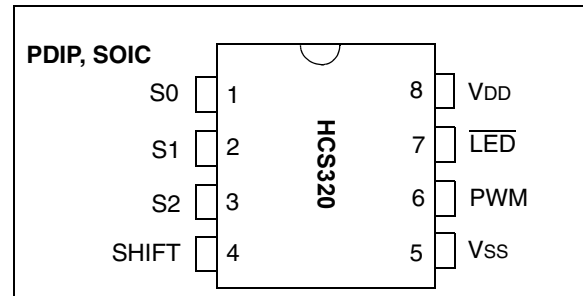
The HCS320 is ideal for Remote Keyless Entry (RKE) applications. These applications include:

- Automotive RKE systems
- Automotive alarm systems
- Automotive immobilizers
- Gate and garage door openers
- Identity tokens
- Burglar alarm systems

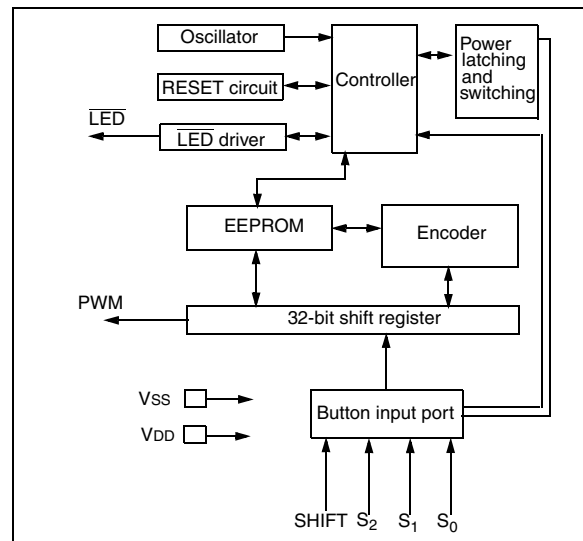
### DESCRIPTION

The HCS320 from Microchip Technology Inc. is a code hopping encoder designed for secure Remote Keyless Entry (RKE) systems. The HCS320 utilizes the code hopping technology which incorporates high security, a small package outline, and low cost, to make this device a perfect solution for unidirectional remote keyless entry systems and access control systems.

### PACKAGE TYPES



### HCS320 BLOCK DIAGRAM



The HCS320 combines a 32-bit hopping code generated by a nonlinear encryption algorithm, with a 28-bit serial number and six status bits to create a 66-bit transmission stream. The length of the transmission eliminates the threat of code scanning and the code hopping mechanism makes each transmission unique, thus rendering code capture and resend (code grabbing) schemes useless.

The crypt key, serial number and configuration data are stored in an EEPROM array which is not accessible via any external connection. The EEPROM data is programmable but read-protected. The data can be verified only after an automatic erase and programming operation. This protects against attempts to gain access to keys or manipulate synchronization values. The HCS320 provides an easy-to-use serial interface for programming the necessary keys, system parameters and configuration data.

## 1.0 SYSTEM OVERVIEW

### Key Terms

The following is a list of key terms used throughout this data sheet. For additional information on KEELOQ® and Code Hopping, refer to Technical Brief 3 (TB003).

- **RKE** - Remote Keyless Entry
- **Button Status** - Indicates what button input(s) activated the transmission. Encompasses the 4 button status bits S3, S2, S1 and S0 (Figure 4-2).
- **Code Hopping** - A method by which a code, viewed externally to the system, appears to change unpredictably each time it is transmitted.
- **Code word** - A block of data that is repeatedly transmitted upon button activation (Figure 4-1).
- **Transmission** - A data stream consisting of repeating code words (Figure 9-2).
- **Crypt key** - A unique and secret 64-bit number used to encrypt and decrypt data. In a symmetrical block cipher such as the KEELOQ algorithm, the encryption and decryption keys are equal and will therefore be referred to generally as the crypt key.
- **Encoder** - A device that generates and encodes data.
- **Encryption Algorithm** - A recipe whereby data is scrambled using a crypt key. The data can only be interpreted by the respective decryption algorithm using the same crypt key.
- **Decoder** - A device that decodes data received from an encoder.
- **Decryption algorithm** - A recipe whereby data scrambled by an encryption algorithm can be unscrambled using the same crypt key.

- **Learn** – Learning involves the receiver calculating the transmitter's appropriate crypt key, decrypting the received hopping code and storing the serial number, synchronization counter value and crypt key in EEPROM. The KEELOQ product family facilitates several learning strategies to be implemented on the decoder. The following are examples of what can be done.

- **Simple Learning**

The receiver uses a fixed crypt key, common to all components of all systems by the same manufacturer, to decrypt the received code word's encrypted portion.

- **Normal Learning**

The receiver uses information transmitted during normal operation to derive the crypt key and decrypt the received code word's encrypted portion.

- **Secure Learn**

The transmitter is activated through a special button combination to transmit a stored 60-bit seed value used to generate the transmitter's crypt key. The receiver uses this seed value to derive the same crypt key and decrypt the received code word's encrypted portion.

- **Manufacturer's code** – A unique and secret 64-bit number used to generate unique encoder crypt keys. Each encoder is programmed with a crypt key that is a function of the manufacturer's code. Each decoder is programmed with the manufacturer code itself.

The HCS320 code hopping encoder is designed specifically for keyless entry systems; primarily vehicles and home garage door openers. The encoder portion of a keyless entry system is integrated into a transmitter, carried by the user and operated to gain access to a vehicle or restricted area. The HCS320 is meant to be a cost-effective yet secure solution to such systems, requiring very few external components (Figure 2-1).

Most low-end keyless entry transmitters are given a fixed identification code that is transmitted every time a button is pushed. The number of unique identification codes in a low-end system is usually a relatively small number. These shortcomings provide an opportunity for a sophisticated thief to create a device that 'grabs' a transmission and retransmits it later, or a device that quickly 'scans' all possible identification codes until the correct one is found.

The HCS320 on the other hand, employs the KEELOQ code hopping technology coupled with a transmission length of 66 bits to virtually eliminate the use of code 'grabbing' or code 'scanning'. The high security level of the HCS320 is based on the patented KEELOQ technology. A block cipher based on a block length of 32 bits and a key length of 64 bits is used. The algorithm obscures the information in such a way that even if the transmission information (before coding) differs by only one bit from that of the previous transmission, the next

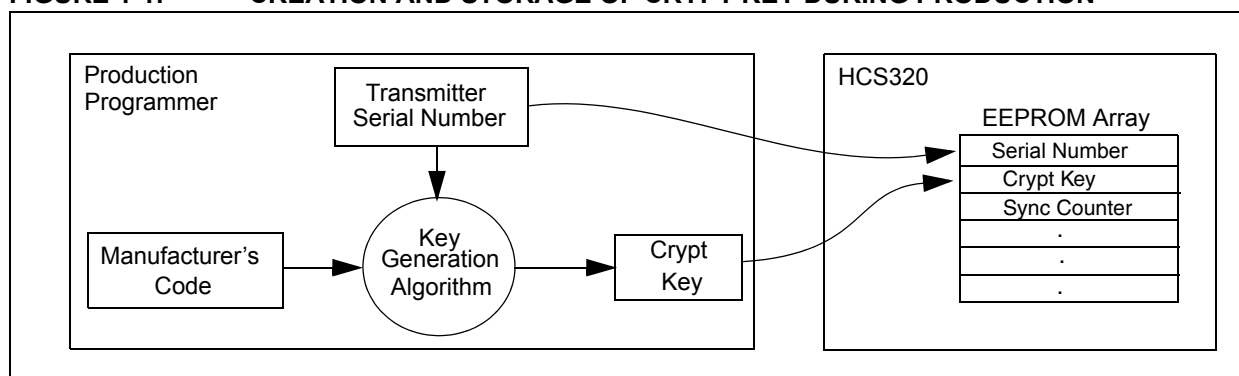
coded transmission will be completely different. Statistically, if only one bit in the 32-bit string of information changes, greater than 50 percent of the coded transmission bits will change.

As indicated in the block diagram on page one, the HCS320 has a small EEPROM array which must be loaded with several parameters before use; most often programmed by the manufacturer at the time of production. The most important of these are:

- A 28-bit serial number, typically unique for every encoder
- A crypt key
- An initial 16-bit synchronization value
- A 16-bit configuration value

The crypt key generation typically inputs the transmitter serial number and 64-bit manufacturer's code into the key generation algorithm (Figure 1-1). The manufacturer's code is chosen by the system manufacturer and must be carefully controlled as it is a pivotal part of the overall system security.

**FIGURE 1-1: CREATION AND STORAGE OF CRYPT KEY DURING PRODUCTION**



The 16-bit synchronization counter is the basis behind the transmitted code word changing for each transmission; it increments each time a button is pressed. Due to the code hopping algorithm's complexity, each increment of the synchronization value results in greater than 50% of the bits changing in the transmitted code word.

Figure 1-2 shows how the key values in EEPROM are used in the encoder. Once the encoder detects a button press, it reads the button inputs and updates the synchronization counter. The synchronization counter and crypt key are input to the encryption algorithm and the output is 32 bits of encrypted information. This data will change with every button press, its value appearing externally to 'randomly hop around', hence it is referred to as the hopping portion of the code word. The 32-bit hopping code is combined with the button information and serial number to form the code word transmitted to the receiver. The code word format is explained in greater detail in Section 4.0.

A receiver may use any type of controller as a decoder, but it is typically a microcontroller with compatible firmware that allows the decoder to operate in conjunction with an HCS320 based transmitter. Section 7.0 provides detail on integrating the HCS320 into a system.

A transmitter must first be 'learned' by the receiver before its use is allowed in the system. Learning includes calculating the transmitter's appropriate crypt key, decrypting the received hopping code and storing the serial number, synchronization counter value and crypt key in EEPROM.

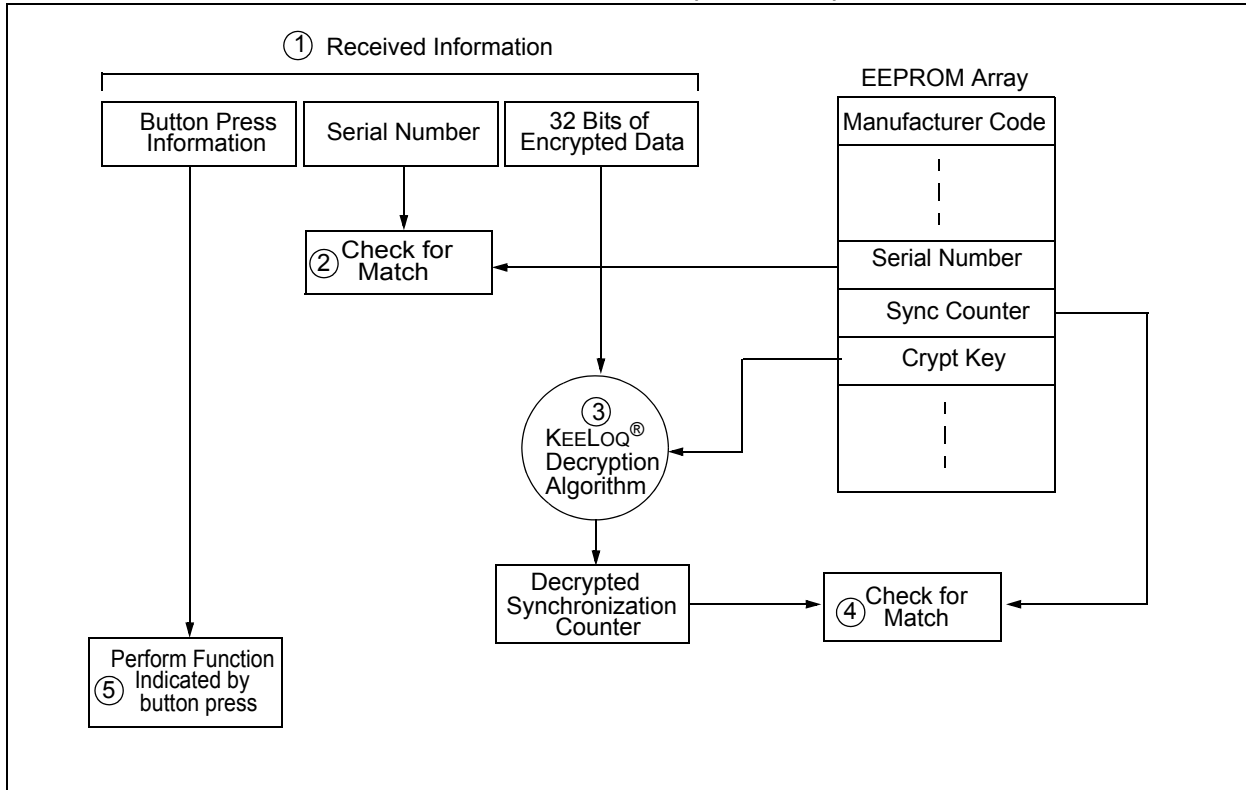
In normal operation, each received message of valid format is evaluated. The serial number is used to determine if it is from a learned transmitter. If from a learned transmitter, the message is decrypted and the synchronization counter is verified. Finally, the button status is checked to see what operation is requested. Figure 1-3 shows the relationship between some of the values stored by the receiver and the values received from the transmitter.

# HCS320

**FIGURE 1-2: BUILDING THE TRANSMITTED CODE WORD (ENCODER)**



**FIGURE 1-3: BASIC OPERATION OF RECEIVER (DECODER)**

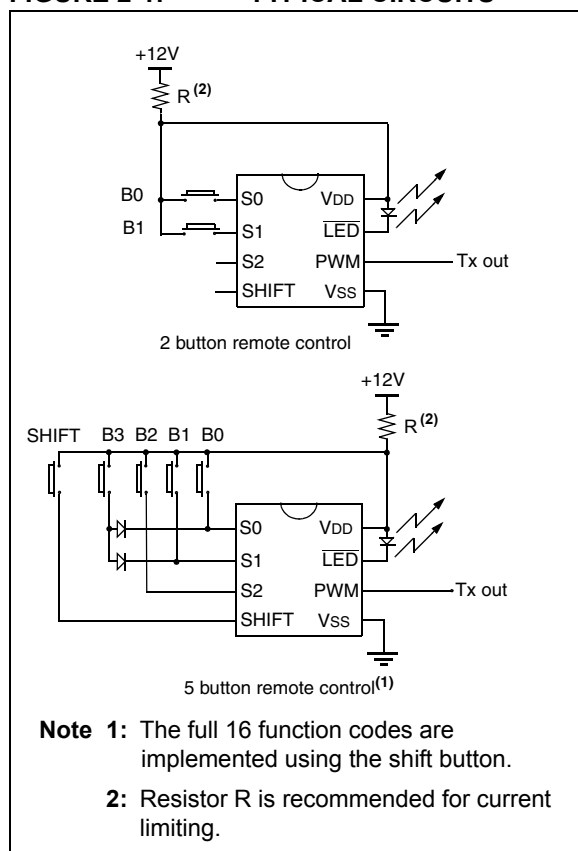


**NOTE:** Circled numbers indicate the order of execution.

## 2.0 DEVICE OPERATION

As shown in the typical application circuits (Figure 2-1), the HCS320 is a simple device to use. It requires only the addition of buttons and RF circuitry for use as the transmitter in your security application. A description of each pin is described in Table 2-1.

**FIGURE 2-1: TYPICAL CIRCUITS**



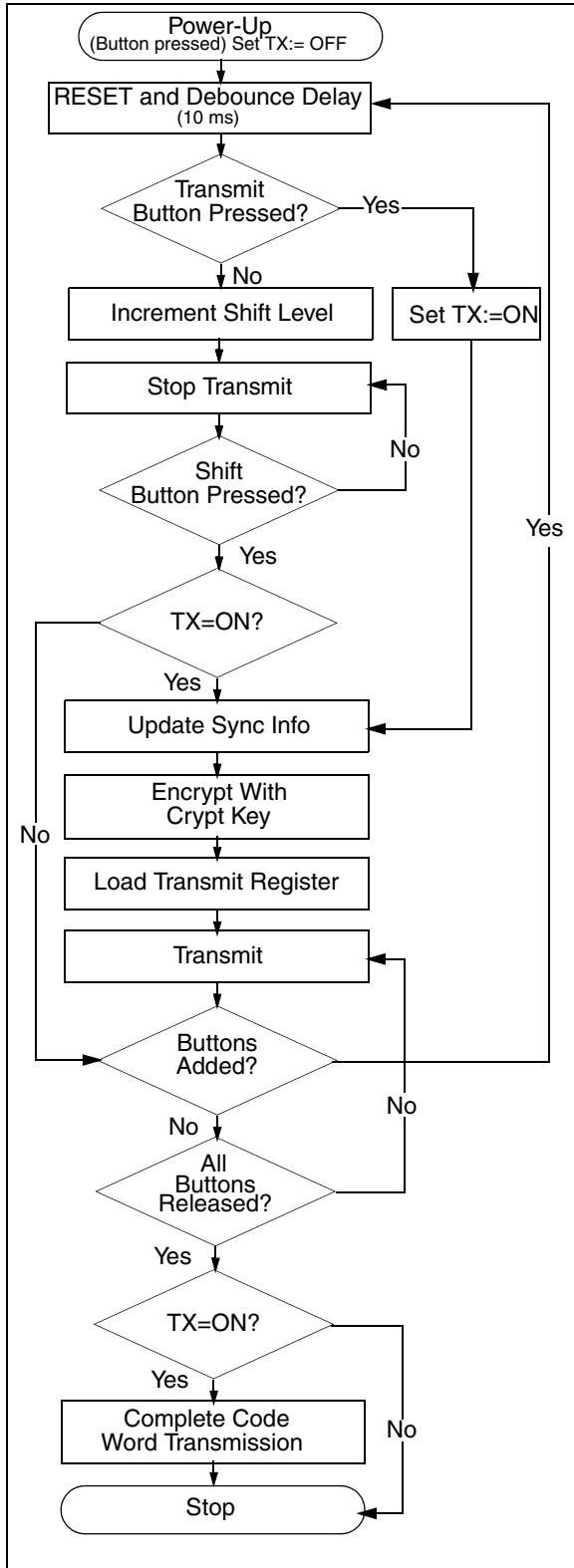
**TABLE 2-1: PIN DESCRIPTIONS**

| Name  | Pin Number | Description   |
|-------|------------|---|
| S0    | 1          | Switch input 0  |
| S1    | 2          | Switch input 1  |
| S2    | 3          | Switch input 2/Clock pin when in Programming mode                       |
| SHIFT | 4          | Switch input for Shift  |
| VSS   | 5          | Ground reference  |
| PWM   | 6          | Pulse Width Modulation (PWM) output pin / Data pin for Programming mode |
| LED   | 7          | Cathode connection for LED  |
| VDD   | 8          | Positive supply voltage   |

The HCS320 will wake-up upon detecting a button press and delay approximately 10 ms for button debounce (Figure 2-2). The synchronization counter, discrimination value and button information will be encrypted to form the hopping code. The hopping code portion will change every transmission, even if the same button is pushed again. A code word that has been transmitted will not repeat for more than 64K transmissions. This provides more than 18 years of use before a code is repeated; based on 10 operations per day. Overflow information sent from the encoder can be used to extend the number of unique transmissions to more than 192K.

If in the transmit process it is detected that a new button(s) has been pressed, a RESET will immediately occur and the current code word will not be completed. Please note that buttons removed will not have any effect on the code word unless no buttons remain pressed; in which case the code word will be completed and the power-down will occur.

**FIGURE 2-2: ENCODER OPERATION**



## 3.0 EEPROM MEMORY ORGANIZATION

The HCS320 contains 192 bits (12 x 16-bit words) of EEPROM memory (Table 3-1). This EEPROM array is used to store the encryption key information, synchronization value, etc. Further descriptions of the memory array is given in the following sections.

**TABLE 3-1: EEPROM MEMORY MAP**

| WORD ADDRESS | MNEMONIC     | DESCRIPTION                          |
|--------------|--------------|--------------------------------------|
| 0            | KEY_0        | 64-bit encryption key (word 0) LSB's |
| 1            | KEY_1        | 64-bit encryption key (word 1)       |
| 2            | KEY_2        | 64-bit encryption key (word 2)       |
| 3            | KEY_3        | 64-bit encryption key (word 3) MSb's |
| 4            | SYNC         | 16-bit synchronization value         |
| 5            | RESERVED     | Set to 0000H                         |
| 6            | SER_0        | Device Serial Number (word 0) LSB's  |
| 7            | SER_1(Notes) | Device Serial Number (word 1) MSb's  |
| 8            | —            | Not used                             |
| 9            | —            | Not used                             |
| 10           | RESERVED     | Set to 0000H                         |
| 11           | CONFIG       | Configuration Word                   |

**Note:** The MSB of the serial number contains a bit used to select the Auto-shutoff timer.

### 3.1 KEY\_0 - KEY\_3 (64-Bit Crypt Key)

The 64-bit crypt key is used to create the encrypted message transmitted to the receiver. This key is calculated and programmed during production using a key generation algorithm. The key generation algorithm may be different from the KEELoQ algorithm. Inputs to the key generation algorithm are typically the transmitter's serial number and the 64-bit manufacturer's code. While the key generation algorithm supplied from Microchip is the typical method used, a user may elect to create their own method of key generation. This may be done providing that the decoder is programmed with the same means of creating the key for decryption purposes.

## 3.2 SYNC (Synchronization Counter)

This is the 16-bit synchronization value that is used to create the hopping code for transmission. This value will be changed after every transmission.

## 3.3 Reserved

Must be initialized to 0000H.

## 3.4 SER\_0, SER\_1 (Encoder Serial Number)

SER\_0 and SER\_1 are the lower and upper words of the device serial number, respectively. Although there are 32 bits allocated for the serial number, only the lower order 28 bits are transmitted. The serial number is meant to be unique for every transmitter. The Most Significant bit of the serial number (Bit 31) is used to turn the Auto-shutoff timer on or off.

### 3.4.1 AUTO-SHUTOFF TIMER ENABLE

The Most Significant bit of the serial number (Bit 31) is used to turn the Auto-shutoff timer on or off. This timer prevents the transmitter from draining the battery should a button get stuck in the on position for a long period of time. The time period is approximately 25 seconds, after which the device will go to the Time-out mode. When in the Time-out mode, the device will stop transmitting, although since some circuits within the device are still active, the current draw within the Shutoff mode will be more than Standby mode. If the Most Significant bit in the serial number is a one, then the Auto-shutoff timer is enabled, and a zero in the Most Significant bit will disable the timer. The length of the timer is not selectable.

## 3.5 CONFIG (Configuration Word)

The Configuration Word is a 16-bit word stored in EEPROM array that is used by the device to store information used during the encryption process, as well as the status of option configurations. The following sections further explain these bits.

**TABLE 3-2: CONFIGURATION WORD**

| Bit Number | Bit Description                          |
|------------|--|
| 0          | Discrimination Bit 0                     |
| 1          | Discrimination Bit 1                     |
| 2          | Discrimination Bit 2                     |
| 3          | Discrimination Bit 3                     |
| 4          | Discrimination Bit 4                     |
| 5          | Discrimination Bit 5                     |
| 6          | Discrimination Bit 6                     |
| 7          | Discrimination Bit 7                     |
| 8          | Discrimination Bit 8                     |
| 9          | Discrimination Bit 9                     |
| 10         | Overflow Bit 0 (OVR0)                    |
| 11         | Overflow Bit 1 (OVR1)                    |
| 12         | Low Voltage Trip Point Select (VLOW SEL) |
| 13         | Baud rate Select Bit 0 (BSL0)            |
| 14         | Baud rate Select Bit 1 (BSL1)            |
| 15         | Reserved, set to 0                       |

### 3.5.1 DISCRIMINATION VALUE (DISC0 TO DISC9)

The discrimination value aids the post-decryption check on the decoder end. It may be any value, but in a typical system it will be programmed as the 10 Least Significant bits of the serial number. Values other than this must be separately stored by the receiver when a transmitter is learned. The discrimination bits are part of the information that form the encrypted portion of the transmission (Figure 4-2). After the receiver has decrypted a transmission, the discrimination bits are checked against the receiver's stored value to verify that the decryption process was valid. If the discrimination value was programmed as the 10 LSB's of the serial number then it may merely be compared to the respective bits of the received serial number; saving EEPROM space.

### 3.5.2 OVERFLOW BITS (OVR0, OVR1)

The overflow bits are used to extend the number of possible synchronization values. The synchronization counter is 16 bits in length, yielding 65,536 values before the cycle repeats. Under typical use of 10 operations a day, this will provide nearly 18 years of use before a repeated value will be used. Should the system designer conclude that is not adequate, then the overflow bits can be utilized to extend the number

# HCS320

of unique values. This can be done by programming OVR0 and OVR1 to 1s at the time of production. The encoder will automatically clear OVR0 the first time that the synchronization value wraps from 0xFFFF to 0x0000 and clear OVR1 the second time the counter wraps. Once cleared, OVR0 and OVR1 cannot be set again, thereby creating a permanent record of the counter overflow. This prevents fast cycling of 64K counter. If the decoder system is programmed to track the overflow bits, then the effective number of unique synchronization values can be extended to 196,608.

### 3.5.3 BAUD RATE SELECT BITS (BSL0, BSL1)

BSL0 and BSL1 select the speed of transmission and the code word blanking. Table 3-3 shows how the bits are used to select the different baud rates and Section 5.6 provides detailed explanation in code word blanking.

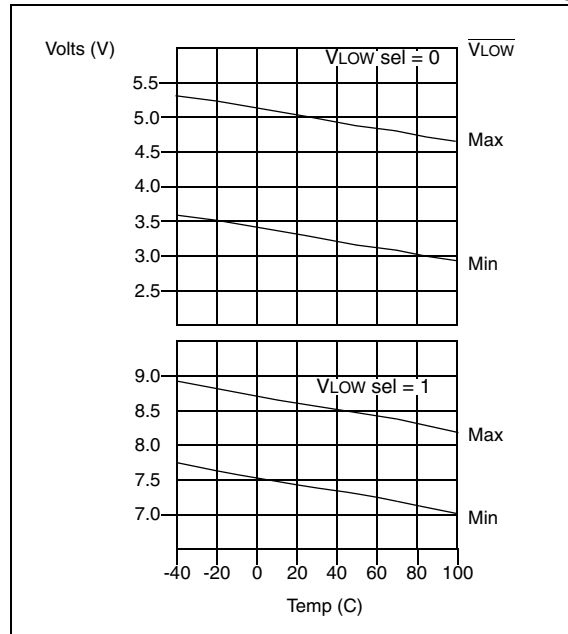
**TABLE 3-3: BAUD RATE SELECT**

| BSL1 | BSL0 | Basic Pulse Element | Code Words Transmitted |
|------|------|---------------------|------------------------|
| 0    | 0    | 400 $\mu$ s         | All                    |
| 0    | 1    | 200 $\mu$ s         | 1 out of 2             |
| 1    | 0    | 100 $\mu$ s         | 1 out of 2             |
| 1    | 1    | 100 $\mu$ s         | 1 out of 4             |

### 3.5.4 LOW VOLTAGE TRIP POINT SELECT

The low voltage trip point select bit is used to tell the HCS320 what VDD level is being used. This information will be used by the device to determine when to send the voltage low signal to the receiver. When this bit is set to a one, the VDD level is assumed to be operating from a 9.0 volt or 12.0 volt VDD level. If the bit is set low, then the VDD level is assumed to be 6.0 volts. Refer to Figure 3-1 for voltage trip point.

**FIGURE 3-1: VOLTAGE TRIP POINTS BY CHARACTERIZATION)**





## 4.0 TRANSMITTED WORD

### 4.1 Code Word Format

The HCS320 code word is made up of several parts (Figure 4-1). Each code word contains a 50% duty cycle preamble, a header, 32 bits of encrypted data and 34 bits of fixed data followed by a guard period before another code word can begin. Refer to Table 9-3 for code word timing.

### 4.2 Code Word Organization

The HCS320 transmits a 66-bit code word when a button is pressed. The 66-bit word is constructed from a Fixed Code portion and an Encrypted Code portion (Figure 4-2).

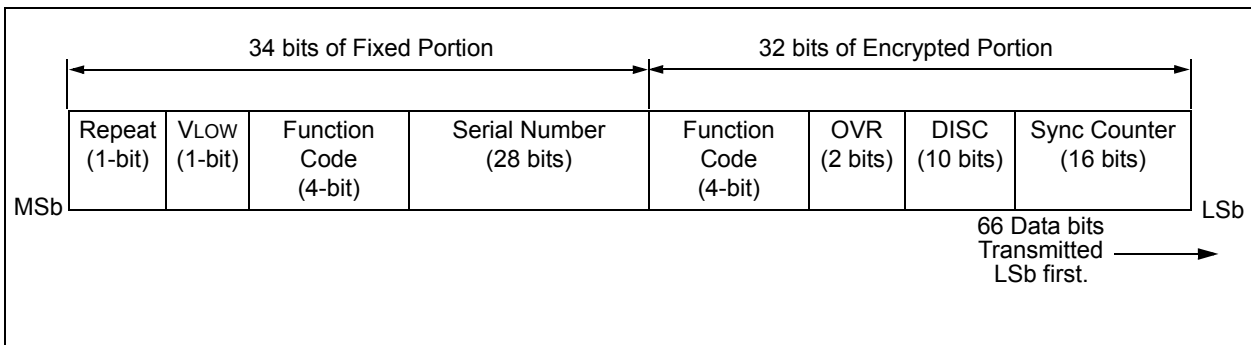
The 32 bits of **Encrypted Data** are generated from 4 button bits, 12 discrimination bits and the 16-bit sync value. The encrypted portion alone provides up to four billion changing code combinations.

The 34 bits of **Fixed Code Data** are made up of 2 status bits, 4 button bits and the 28-bit serial number. The fixed and encrypted sections combined increase the number of code combinations to  $7.38 \times 10^{19}$ .

**FIGURE 4-1: CODE WORD FORMAT**



**FIGURE 4-2: CODE WORD ORGANIZATION**



# HCS320

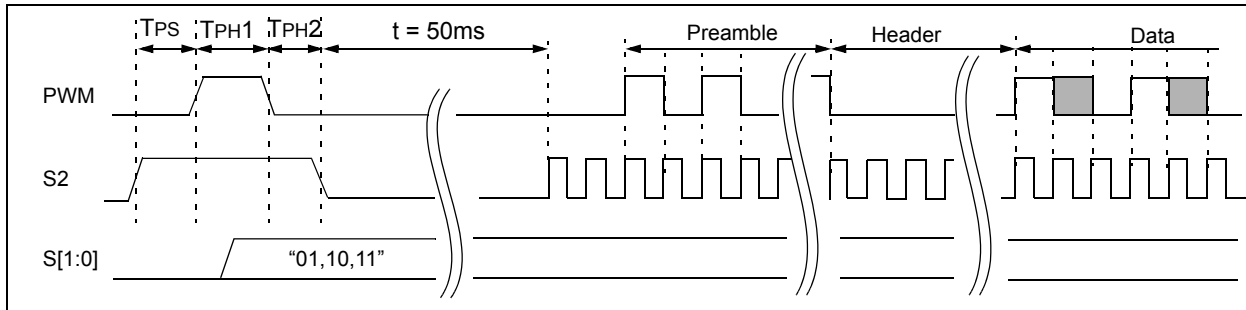
## 4.3 Synchronous Transmission Mode

Synchronous Transmission mode can be used to clock the code word out using an external clock.

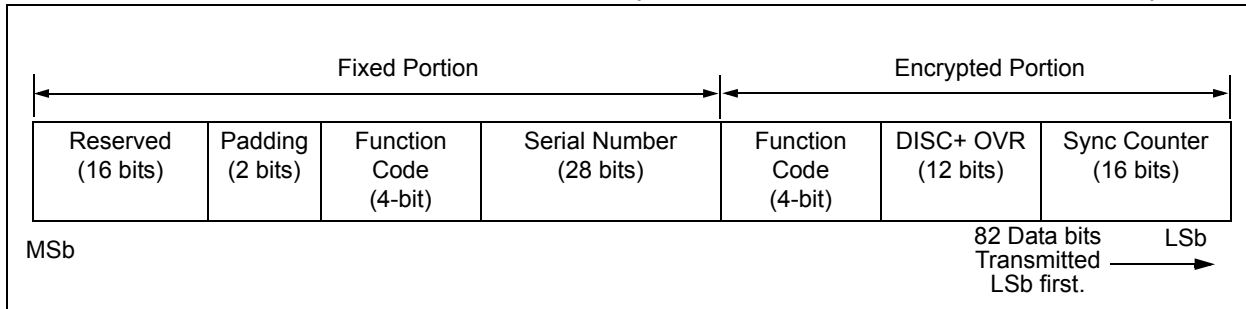
To enter Synchronous Transmission mode, the Programming mode start-up sequence must be executed as shown in Figure 4-3. If either S1 or S0 is set on the falling edge of S2, the device enters Synchronous Transmission mode. In this mode, it functions as a normal transmitter, with the exception that the timing of the PWM data string is controlled externally and 16 extra bits are transmitted at the end with the code word.

The button code will be the S0, S1 value at the falling edge of S2. The timing of the PWM data string is controlled by supplying a clock on S2 and should not exceed 20 kHz. The code word is the same as in PWM mode with 16 reserved bits at the end of the word. The reserved bits can be ignored. When in Synchronous Transmission mode S2 should not be toggled until all internal processing has been completed as shown in Figure 4-4.

**FIGURE 4-3: SYNCHRONOUS TRANSMISSION MODE**



**FIGURE 4-4: CODE WORD ORGANIZATION (SYNCHRONOUS TRANSMISSION MODE)**



## 5.0 SPECIAL FEATURES

### 5.1 Code Word Completion

Code word completion is an automatic feature that makes sure that the entire code word is transmitted, even if the transmit button is released before the transmission is complete. The HCS320 encoder powers itself up when a button is pushed and powers itself down after the command is finished, if the user has already released the button. If the button is held down beyond the time for one transmission, then multiple transmissions will result. If another button is activated during a transmission, the active transmission will be aborted and the function new code will be generated using the new button information.

### 5.2 Auto-Shutoff

The Auto-shutoff function automatically stops the device from transmitting if a button inadvertently gets pressed for a long period of time. This will prevent the device from draining the battery if a button gets pressed while the transmitter is in a pocket or purse. This function can be enabled or disabled and is selected by setting or clearing the Auto-shutoff bit (Section 3.4.1). Setting this bit high will enable the function (turn Auto-shutoff function on) and setting the bit low will disable the function. Time-out period is dependent on the shift level and is approximately  $42 \pm 10$  seconds.

### 5.3 VLOW: Voltage LOW Indicator

The VLOW bit is transmitted with every transmission (Figure 9-6) and will be transmitted as a one if the operating voltage has dropped below the low voltage trip point. The trip point is selectable between two values, based on the battery voltage being used. See Section 3.5.4 for a description of how the low voltage select option is set. This VLOW signal is transmitted so the receiver can alert the user that the transmitter battery is low.

**Note:** Depending on the internal resistance of the VDD source, VDD may normally be above the VLOW trip point except when the LED is turned on. In this case, the VLOW bit will be transmitted as a one when a transmission occurs while the LED is on. The VLOW bit will be transmitted as a zero when a transmission occurs while the LED is off.

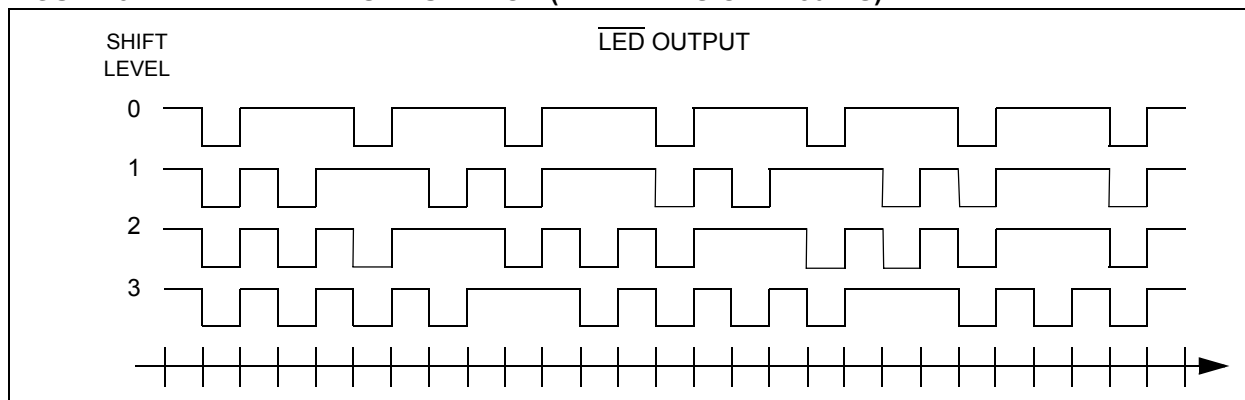
### 5.4 RPT: Repeat Indicator

This bit will be low for the first transmitted word. If a button is held down for more than one transmitted code word, this bit will be set to indicate a repeated code word and remain set until the button is released.

### 5.5 $\overline{\text{LED}}$ Output Operation

During normal transmission the  $\overline{\text{LED}}$  output (Figure 5-1) indicates the shift level (Section 5.7) by flashing the LED in a pattern corresponding to the shift level. If the supply voltage drops below the low voltage trip point (Section 3.5.4), the  $\overline{\text{LED}}$  output will be toggled at approximately 5 Hz during the transmission.

**FIGURE 5-1: LED FLASH FUNCTION (EACH DIVISION - 180 MS)**

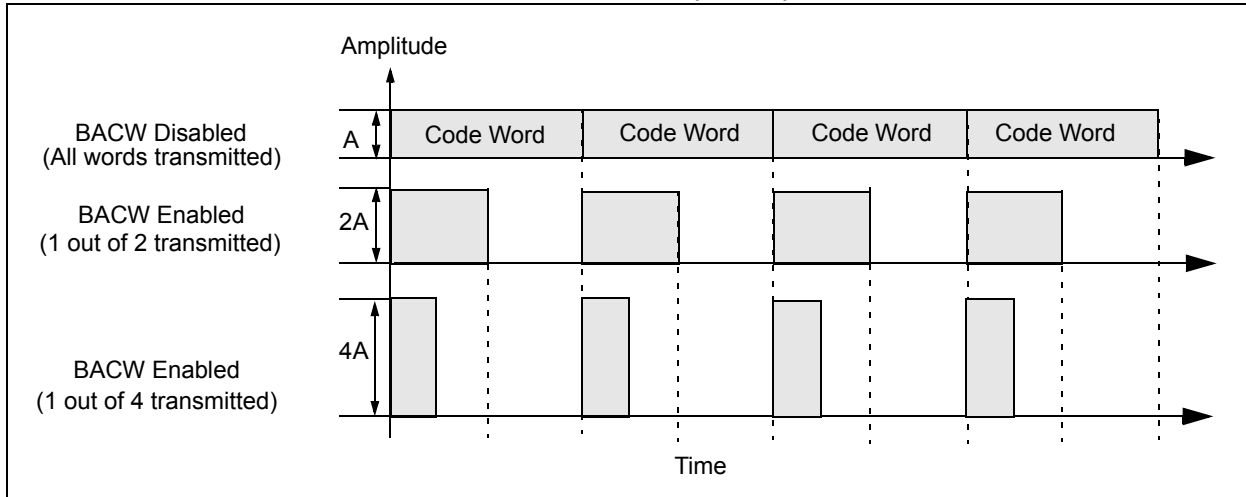


## 5.6 Blank Alternate Code Word

Federal Communications Commission (FCC) part 15 rules specify the limits on fundamental power and harmonics that can be transmitted. Power is calculated on the worst case average power transmitted in a 100 ms window. It is therefore advantageous to minimize the duty cycle of the transmitted word. This can be achieved by minimizing the duty cycle of the individual bits and by blanking out consecutive words. Blank Alternate Code Word (BACW) is used for reducing the average power of a transmission

(Figure 5-1). This is a selectable feature that is determined in conjunction with the baud rate selection bits BSL0 and BSL1. Using the BACW allows the user to transmit a higher amplitude transmission if the transmission length is shorter. The FCC puts constraints on the average power that can be transmitted by a device, and BACW effectively prevents continuous transmission by only allowing the transmission of every second or every fourth code word. This reduces the average power transmitted and hence, assists in FCC approval of a transmitter device.

**FIGURE 5-2: BLANK ALTERNATE CODE WORD (BACW)**



## 5.7 SHIFT Key Operation

The HCS320 has four switch inputs usually connected to buttons as shown in Figure 2-1: Typical Circuits.

Any button connected to input S0, S1 or S2 is called a TRANSMIT button as it causes a transmission when pressed.

The SHIFT button is connected to the SHIFT input. Pressing the SHIFT button increments a counter by one count and does not result in a transmission. The counter value is called the shift level. Successive presses of the SHIFT button can increase the shift level up to three before wrapping back to zero. The shift level is available for eight seconds when the SHIFT button is released, after which the shift level is reset to zero.

When a TRANSMIT button is pressed, the function code transmitted for that button depends on the shift level. The transmitted function code corresponding to shift level and S0, S1 and S2 switch activation is shown in Table 5-1 for all legal combinations of shift level and button input. Note that a shift level of zero means that the SHIFT button has not been pressed (or it has been pressed four times). The shift level is reset to zero after a transmission.

The volatile nature of the shift level register requires the HCS320 to be powered continuously for correct operation and not powered via the buttons.

**TABLE 5-1: PIN ACTIVATION TABLE**

| SHIFT LEVEL | S2 | S1 | S0 | FUNCTION CODE   |
|-------------|----|----|----|-----------------|
| 0           | 0  | 0  | 0  | No Transmission |
| 0           | 0  | 0  | 1  | 0h              |
| 0           | 0  | 1  | 0  | 1h              |
| 0           | 0  | 1  | 1  | 2h              |
| 0           | 1  | 0  | 0  | 3h              |
| 1           | 0  | 0  | 0  | No Transmission |
| 1           | 0  | 0  | 1  | 4h              |
| 1           | 0  | 1  | 0  | 5h              |
| 1           | 0  | 1  | 1  | 6h              |
| 1           | 1  | 0  | 0  | 7h              |
| 2           | 0  | 0  | 0  | No Transmission |
| 2           | 0  | 0  | 1  | 8h              |
| 2           | 0  | 1  | 0  | 9h              |
| 2           | 0  | 1  | 1  | Ah              |
| 2           | 1  | 0  | 0  | Bh              |
| 3           | 0  | 0  | 0  | No Transmission |
| 3           | 0  | 0  | 1  | Ch              |
| 3           | 0  | 1  | 0  | Dh              |
| 3           | 0  | 1  | 1  | Eh              |
| 3           | 1  | 0  | 0  | Fh              |

# HCS320

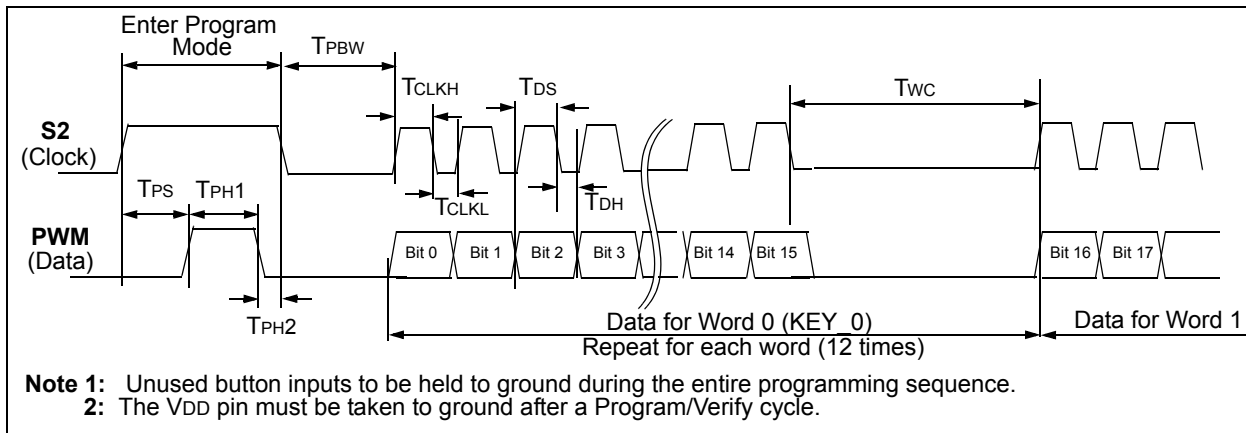
## 6.0 PROGRAMMING THE HCS320

When using the HCS320 in a system, the user will have to program some parameters into the device including the serial number and the secret key before it can be used. The programming cycle allows the user to input all 192 bits in a serial data stream, which are then stored internally in EEPROM. Programming will be initiated by forcing the PWM line high, after the S2 line has been held high for the appropriate length of time line (Table 6-1 and Figure 6-1). After the Program mode is entered, a delay must be provided to the device for the automatic bulk write cycle to complete. This will set all locations in the EEPROM to zeros. The device can then be programmed by clocking in 16 bits at a time, using S2 as the clock line and PWM as the

data in line. After each 16-bit word is loaded, a programming delay is required for the internal program cycle to complete. This delay can take up to  $T_{WC}$ . At the end of the programming cycle, the device can be verified (Figure 6-2) by reading back the EEPROM. Reading is done by clocking the S2 line and reading the data bits on PWM. For security reasons, it is not possible to execute a verify function without first programming the EEPROM. **A Verify operation can only be done once, immediately following the Program cycle.**

**Note:** To ensure that the device does not accidentally enter Programming mode, PWM should never be pulled high by the circuit connected to it. Special care should be taken when driving PNP RF transistors.

**FIGURE 6-1: PROGRAMMING WAVEFORMS**



**FIGURE 6-2: VERIFY WAVEFORMS**

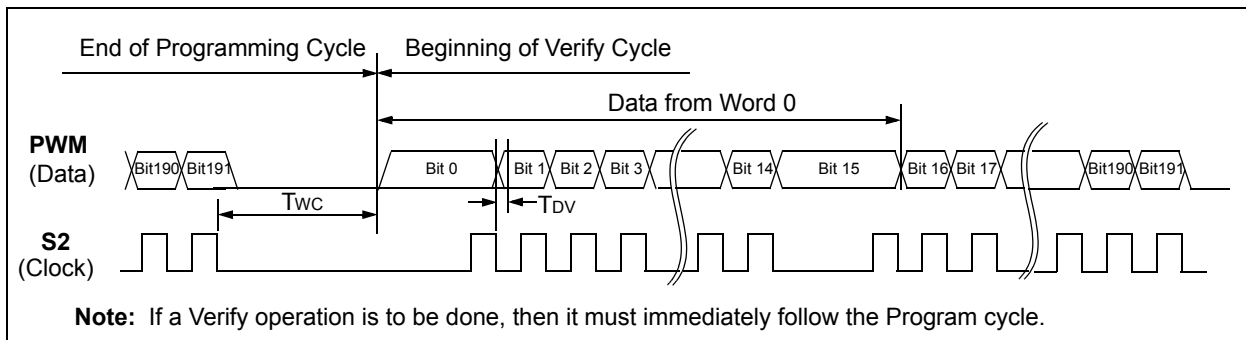


TABLE 6-1: PROGRAMMING/VERIFY TIMING REQUIREMENTS

| VDD = 5.0V ± 10%, 25 °C ± 5 °C |                   |      |      |                   |
|--------------------------------|-------------------|------|------|-------------------|
| Parameter                      | Symbol            | Min. | Max. | Units             |
| Program mode setup time        | T <sub>PS</sub>   | 3.5  | 4.5  | ms                |
| Hold time 1                    | T <sub>PH1</sub>  | 3.5  | —    | ms                |
| Hold time 2                    | T <sub>PH2</sub>  | 50   | —    | μs                |
| Bulk Write time                | T <sub>PBW</sub>  | 4.0  | —    | ms                |
| Program delay time             | T <sub>PROG</sub> | 4.0  | —    | ms                |
| Program cycle time             | T <sub>WC</sub>   | 50   | —    | ms                |
| Clock low time                 | T <sub>CLKL</sub> | 50   | —    | μs                |
| Clock high time                | T <sub>CLKH</sub> | 50   | —    | μs                |
| Data setup time                | T <sub>DS</sub>   | 0    | —    | μs <sup>(1)</sup> |
| Data hold time                 | T <sub>DH</sub>   | 30   | —    | μs <sup>(1)</sup> |
| Data out valid time            | T <sub>DV</sub>   | —    | 30   | μs <sup>(1)</sup> |

**Note 1:** Typical values - not tested in production.

## 7.0 INTEGRATING THE HCS320 INTO A SYSTEM

Use of the HCS320 in a system requires a compatible decoder. This decoder is typically a microcontroller with compatible firmware. Microchip will provide (via a license agreement) firmware routines that accept transmissions from the HCS320 and decrypt the hopping code portion of the data stream. These routines provide system designers the means to develop their own decoding system.

### 7.1 Learning a Transmitter to a Receiver

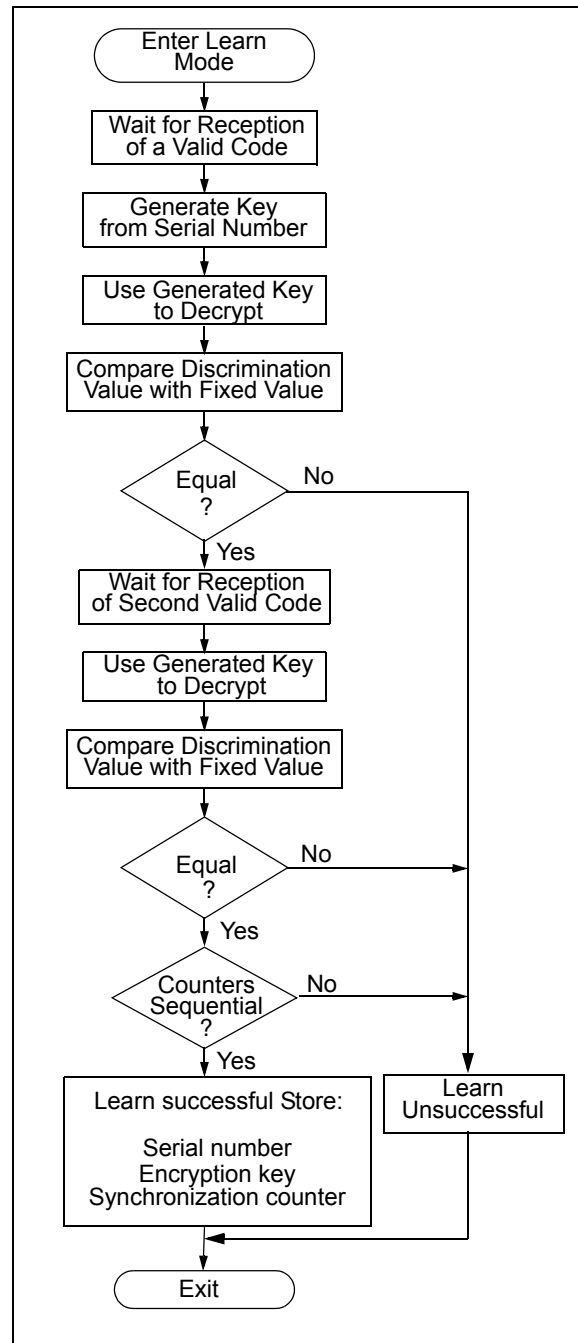
A transmitter must first be 'learned' by a decoder before its use is allowed in the system. Several learning strategies are possible, Figure 7-1 details a typical learn sequence. Core to each, the decoder must minimally store each learned transmitter's serial number and current synchronization counter value in EEPROM. Additionally, the decoder typically stores each transmitter's unique crypt key. The maximum number of learned transmitters will therefore be relative to the available EEPROM.

A transmitter's serial number is transmitted in the clear but the synchronization counter only exists in the code word's encrypted portion. The decoder obtains the counter value by decrypting using the same key used to encrypt the information. The KEELOQ algorithm is a symmetrical block cipher so the encryption and decryption keys are identical and referred to generally as the crypt key. The encoder receives its crypt key during manufacturing. The decoder is programmed with the ability to generate a crypt key as well as all but one required input to the key generation routine; typically the transmitter's serial number.

Figure 7-1 summarizes a typical learn sequence. The decoder receives and authenticates a first transmission; first button press. Authentication involves generating the appropriate crypt key, decrypting, validating the correct key usage via the discrimination bits and buffering the counter value. A second transmission is received and authenticated. A final check verifies the counter values were sequential; consecutive button presses. If the learn sequence is successfully complete, the decoder stores the learned transmitter's serial number, current synchronization counter value and appropriate crypt key. From now on the crypt key will be retrieved from EEPROM during normal operation instead of recalculating it for each transmission received.

Certain learning strategies have been patented and care must be taken not to infringe.

FIGURE 7-1: TYPICAL LEARN SEQUENCE

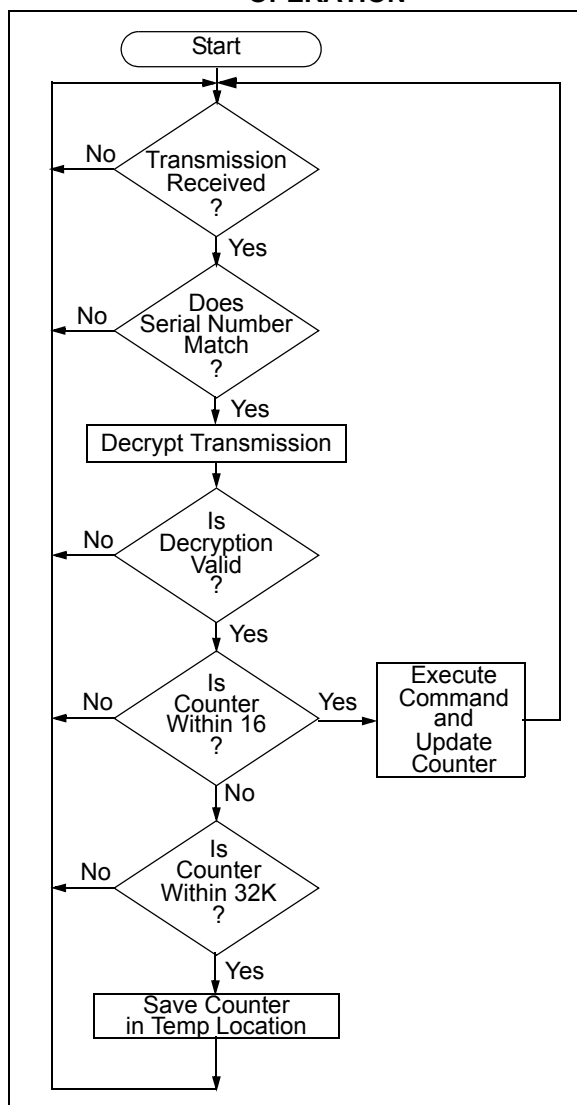




## 7.2 Decoder Operation

Figure 7-2 summarizes normal decoder operation. The decoder waits until a transmission is received. The received serial number is compared to the EEPROM table of learned transmitters to first determine if this transmitter's use is allowed in the system. If from a learned transmitter, the transmission is decrypted using the stored crypt key and authenticated via the discrimination bits for appropriate crypt key usage. If the decryption was valid the synchronization value is evaluated.

**FIGURE 7-2: TYPICAL DECODER OPERATION**



## 7.3 Synchronization with Decoder (Evaluating the Counter)

The KEELOQ technology patent scope includes a sophisticated synchronization technique that does not require the calculation and storage of future codes. The technique securely blocks invalid transmissions while providing transparent resynchronization to transmitters inadvertently activated away from the receiver.

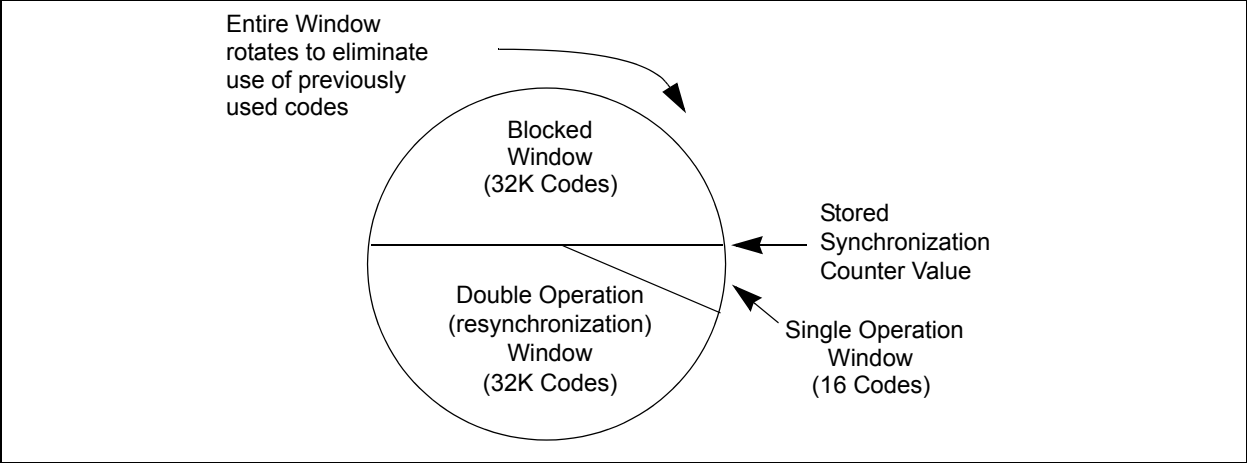
Figure 7-3 shows a 3-partition, rotating synchronization window. The size of each window is optional but the technique is fundamental. Each time a transmission is authenticated, the intended function is executed and the transmission's synchronization counter value is stored in EEPROM. From the currently stored counter value there is an initial "Single Operation" forward window of 16 codes. If the difference between a received synchronization counter and the last stored counter is within 16, the intended function will be executed on the single button press and the new synchronization counter will be stored. Storing the new synchronization counter value effectively rotates the entire synchronization window.

A "Double Operation" (resynchronization) window further exists from the Single Operation window up to 32K codes forward of the currently stored counter value. It is referred to as "Double Operation" because a transmission with synchronization counter value in this window will require an additional, sequential counter transmission prior to executing the intended function. Upon receiving the sequential transmission the decoder executes the intended function and stores the synchronization counter value. This resynchronization occurs transparently to the user as it is human nature to press the button a second time if the first was unsuccessful.

The third window is a "Blocked Window" ranging from the double operation window to the currently stored synchronization counter value. Any transmission with synchronization counter value within this window will be ignored. This window excludes previously used, perhaps code-grabbed transmissions from accessing the system.

**Note:** The synchronization method described in this section is only a typical implementation and because it is usually implemented in firmware, it can be altered to fit the needs of a particular system.

**FIGURE 7-3: SYNCHRONIZATION WINDOW**



## 8.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers and dsPIC® digital signal controllers are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB C Compiler for Various Device Families
  - HI-TECH C for Various Device Families
  - MPASM™ Assembler
  - MPLINK™ Object Linker/  
MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers
  - MPLAB ICD 3
  - PICKit™ 3 Debug Express
- Device Programmers
  - PICKit™ 2 Programmer
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits, and Starter Kits

## 8.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16/32-bit microcontroller market. The MPLAB IDE is a Windows® operating system-based application that contains:

- A single graphical interface to all debugging tools
  - Simulator
  - Programmer (sold separately)
  - In-Circuit Emulator (sold separately)
  - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either C or assembly)
- One-touch compile or assemble, and download to emulator and simulator tools (automatically updates all project information)
- Debug using:
  - Source files (C or assembly)
  - Mixed C and assembly
  - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

## 8.2 MPLAB C Compilers for Various Device Families

The MPLAB C Compiler code development systems are complete ANSI C compilers for Microchip's PIC18, PIC24 and PIC32 families of microcontrollers and the dsPIC30 and dsPIC33 families of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

## 8.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

The compilers include a macro assembler, linker, pre-processor, and one-step driver, and can run on multiple platforms.

## 8.4 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

## 8.5 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

## 8.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

## 8.7 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC<sup>®</sup> MCUs and dsPIC<sup>®</sup> DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

## 8.8 MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs PIC<sup>®</sup> Flash MCUs and dsPIC<sup>®</sup> Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

## 8.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC<sup>®</sup> Flash microcontrollers and dsPIC<sup>®</sup> DSCs with the powerful, yet easy-to-use graphical user interface of MPLAB Integrated Development Environment (IDE).

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

## 8.10 PICkit 3 In-Circuit Debugger/Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC<sup>®</sup> and dsPIC<sup>®</sup> Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (IDE). The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface and can be connected to the target via an Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

## 8.11 PICkit 2 Development Programmer/Debugger and PICkit 2 Debug Express

The PICkit™ 2 Development Programmer/Debugger is a low-cost development tool with an easy to use interface for programming and debugging Microchip's Flash families of microcontrollers. The full featured Windows® programming interface supports baseline (PIC10F, PIC12F5xx, PIC16F5xx), midrange (PIC12F6xx, PIC16F), PIC18F, PIC24, dsPIC30, dsPIC33, and PIC32 families of 8-bit, 16-bit, and 32-bit microcontrollers, and many Microchip Serial EEPROM products. With Microchip's powerful MPLAB Integrated Development Environment (IDE) the PICkit™ 2 enables in-circuit debugging on most PIC® microcontrollers. In-Circuit-Debugging runs, halts and single steps the program while the PIC microcontroller is embedded in the application. When halted at a breakpoint, the file registers can be examined and modified.

The PICkit 2 Debug Express include the PICkit 2, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

## 8.12 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages and a modular, detachable socket assembly to support various package types. The ICSP™ cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices and incorporates an MMC card for file storage and data applications.

## 8.13 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page ([www.microchip.com](http://www.microchip.com)) for the complete list of demonstration, development and evaluation kits.

## 9.0 ELECTRICAL CHARACTERISTICS

**TABLE 9-1: ABSOLUTE MAXIMUM RATINGS**

| Symbol | Item                | Rating            | Units     |
|--------|---------------------|-------------------|-----------|
| VDD    | Supply voltage      | -0.3 to 13.3      | V         |
| VIN    | Input voltage       | -0.3 to 13.3      | V         |
| VOUT   | Output voltage      | -0.3 to VDD + 0.3 | V         |
| IOUT   | Max output current  | 25                | mA        |
| TSTG   | Storage temperature | -55 to +125       | °C (Note) |
| TLSOL  | Lead soldering temp | 300               | °C (Note) |
| VESD   | ESD rating          | 4000              | V         |

**Note:** Stresses above those listed under “ABSOLUTE MAXIMUM RATINGS” may cause permanent damage to the device.

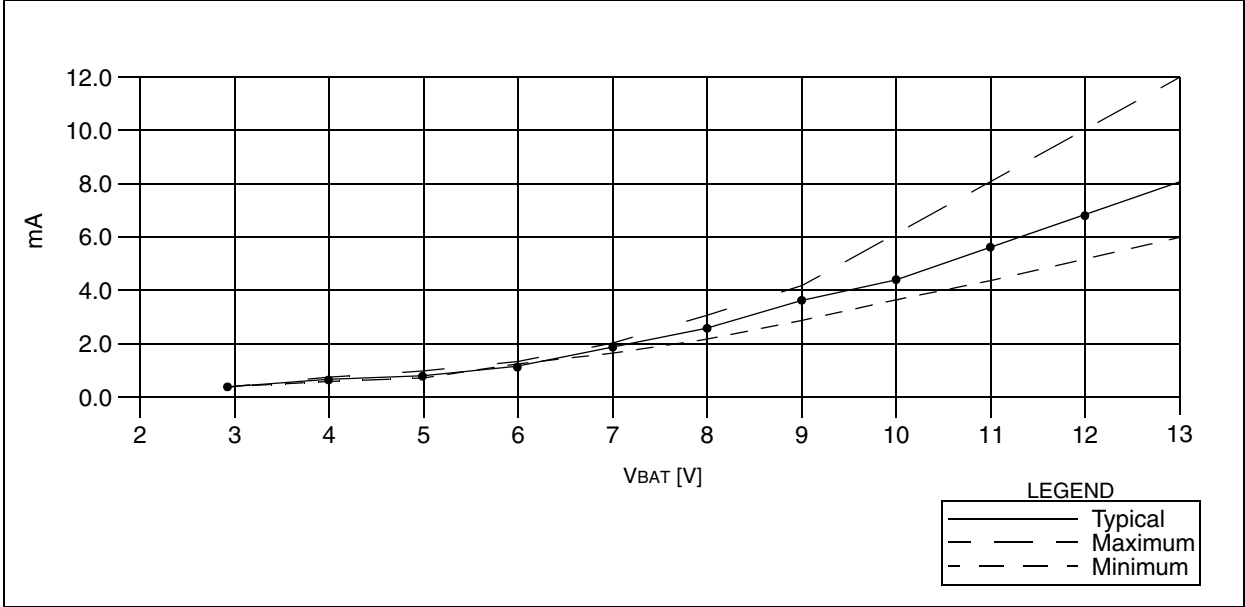
**TABLE 9-2: DC CHARACTERISTICS**

| Commercial(C):Tamb = 0°C to +70°C<br>Industrial(I):Tamb = -40°C to +85°C |       |             |                    |                    |      |   |
|--|-------|-------------|--------------------|--------------------|------|---|
| 3.5V < VDD < 13.0V   |       |             |                    |                    |      |   |
| Parameter  | Sym.  | Min         | Typ*               | Max                | Unit | Conditions  |
| Operating current (avg)  | ICC   |             | 0.6<br>2.0<br>10.0 | 1.0<br>3.0<br>15.0 | mA   | VDD = 3.5V<br>VDD = 6.6V<br>VDD = 13.0V<br>(Figure 9-1) |
| Standby current  | ICCS  |             | 1                  | 10                 | µA   |   |
| High level Input voltage   | VIH   | 0.4 VDD     |                    | VDD+<br>0.3        | V    |   |
| Low level input voltage  | VIL   | -0.3        |                    | 0.15 VDD           | V    |   |
| High level output voltage  | VOH   | 0.5 VDD     |                    |                    | V    | IOH = -2 mA   |
| Low level output voltage   | VOL   |             |                    | 0.11 VDD           | V    | IOL = 2 mA  |
| LED sink current   | ILED  | 5.0<br>11.0 | 6.5<br>14          | 9.0<br>20          | mA   | VDD = 6.6V<br>VDD = 13.0V                               |
| Pull-down Resistance;<br>S0,S1,S2, SHIFT                                 | RS0-3 | 40          | 60                 | 80                 | KΩ   | VIN = 4.0V  |
| Pull-down Resistance;<br>PWM   | RPWM  | 80          | 120                | 160                | KΩ   | VIN = 4.0V  |

**Note:** Typical values are at 25°C.

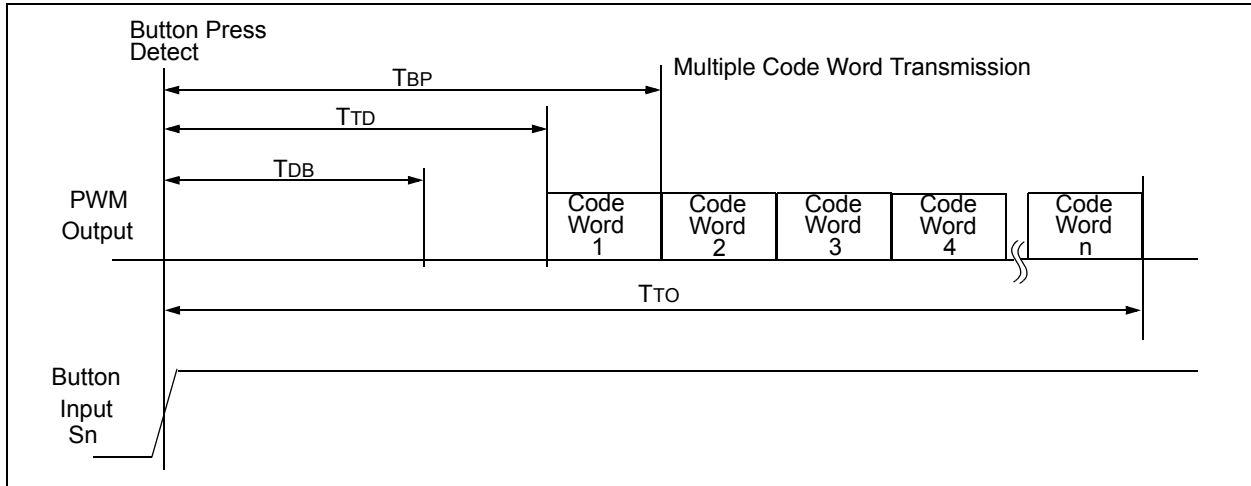
# HCS320

FIGURE 9-1: TYPICAL ICC CURVE OF HCS320





**FIGURE 9-2: POWER-UP AND TRANSMIT TIMING**



**FIGURE 9-3: POWER-UP AND TRANSMIT TIMING REQUIREMENTS**

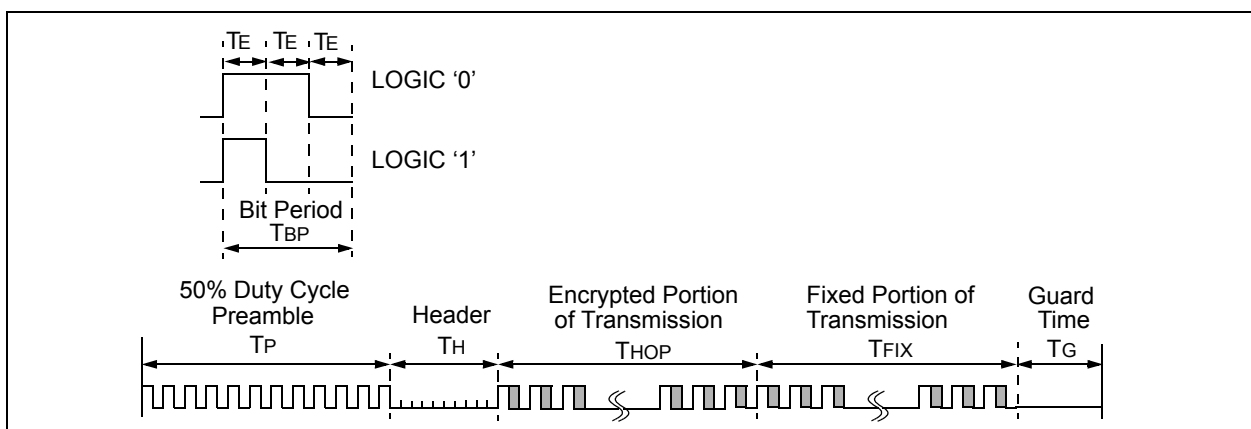
VDD = +3.5 to 13.0V  
 Commercial (C): Tamb = 0°C to +70°C  
 Industrial (I): Tamb = -40°C to +85°C

| Parameter                         | Symbol | Min                 | Max                 | Unit | Remarks  |
|-----------------------------------|--------|---------------------|---------------------|------|----------|
| Time to second button press       | TBP    | 10 + Code Word Time | 27 + Code Word Time | ms   | (Note 1) |
| Transmit delay from button detect | TTD    | 10                  | 27                  | ms   |          |
| Debounce delay                    | TDB    | 6                   | 15                  | ms   |          |
| Auto-shutoff time-out period      | TTO    | 22                  | 77                  | s    | (Note 2) |

**Note 1:** TBP is the time in which a second button can be pressed without completion of the first code word and the intention was to press the combination of buttons.

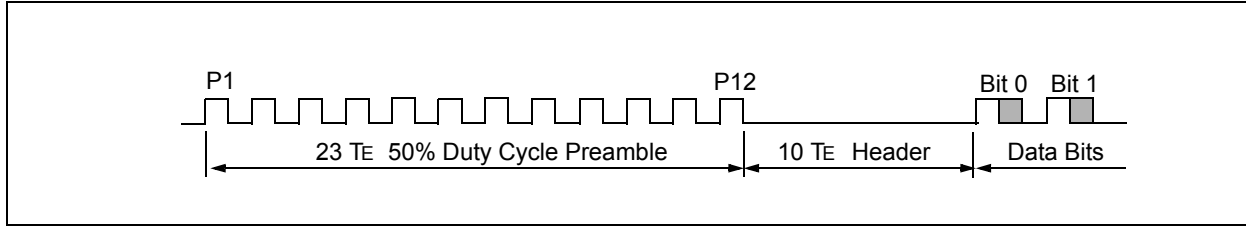
**2:** The Auto-shutoff time-out period is not tested.

**FIGURE 9-4: CODE WORD FORMAT**

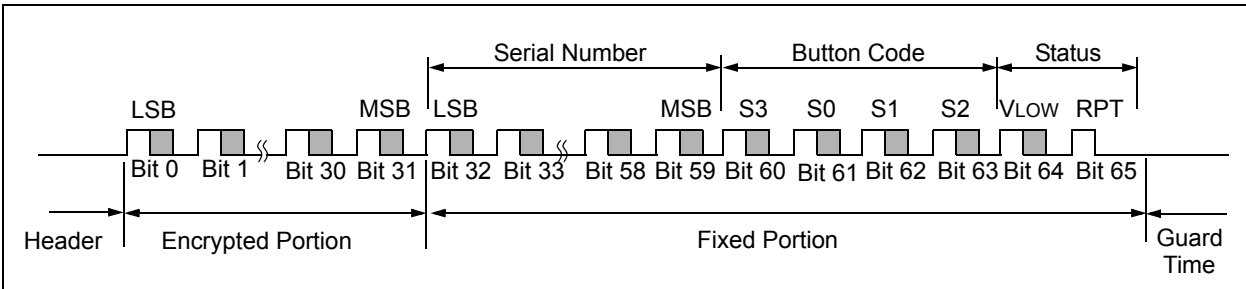


# HCS320

**FIGURE 9-5: CODE WORD FORMAT: PREAMBLE/HEADER PORTION**



**FIGURE 9-6: CODE WORD FORMAT: DATA PORTION**

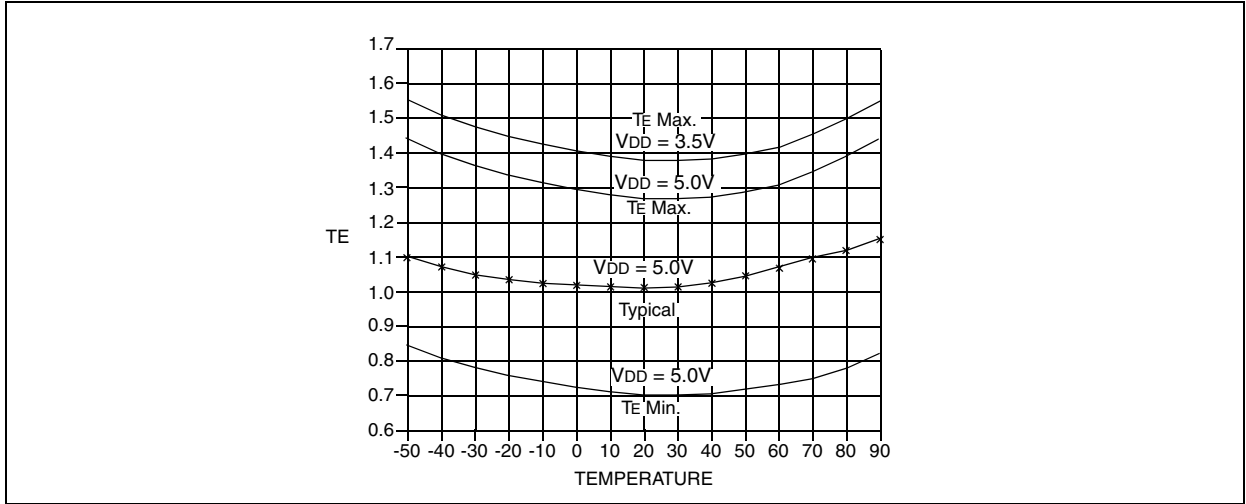


**TABLE 9-3: CODE WORD TRANSMISSION TIMING REQUIREMENTS**

| VDD = +3.5 to 13.0<br>Commercial(C): Tamb = 0°C to +70°C<br>Industrial(I): Tamb = -40°C to +85°C |                       |              | Code Words Transmitted |       |       |            |      |       |            |      |      | Units |
|--|-----------------------|--------------|------------------------|-------|-------|------------|------|-------|------------|------|------|-------|
| Symbol   | Characteristic        | Number of TE | All                    |       |       | 1 out of 2 |      |       | 1 out of 4 |      |      |       |
|  |                       |              | Min.                   | Typ.  | Max.  | Min.       | Typ. | Max.  | Min.       | Typ. | Max. |       |
| TE   | Basic pulse element   | 1            | 280                    | 400   | 620   | 140        | 200  | 310   | 70         | 100  | 155  | μs    |
| TBP  | PWM bit pulse width   | 3            | 840                    | 1200  | 1860  | 420        | 600  | 930   | 210        | 300  | 465  | μs    |
| TP   | Preamble duration     | 23           | 6.4                    | 9.2   | 14.3  | 3.2        | 4.6  | 7.1   | 1.6        | 2.3  | 3.6  | ms    |
| TH   | Header duration       | 10           | 2.8                    | 4.0   | 6.2   | 1.4        | 2.0  | 3.1   | 0.7        | 1.0  | 1.6  | ms    |
| THOP   | Hopping code duration | 96           | 26.9                   | 38.4  | 59.5  | 13.4       | 19.2 | 29.8  | 6.7        | 9.6  | 14.9 | ms    |
| TFIX   | Fixed code duration   | 102          | 28.6                   | 40.8  | 63.2  | 14.3       | 20.4 | 31.6  | 7.1        | 10.2 | 15.8 | ms    |
| TG   | Guard Time            | 199          | 55.6                   | 79.6  | 123.5 | 28.1       | 39.8 | 61.7  | 13.8       | 19.9 | 30.6 | ms    |
| —  | Total Transmit Time   | 430          | 120.3                  | 172.0 | 266.7 | 60.4       | 86.0 | 133.3 | 29.9       | 43.0 | 66.5 | ms    |
| —  | PWM data rate         | —            | 1190                   | 833   | 538   | 2381       | 1667 | 1075  | 4762       | 3333 | 2151 | bps   |

**Note:** The timing parameters are not tested but derived from the oscillator clock.

**FIGURE 9-7: HCS320 TE VS. TEMP (BY CHARACTERIZATION ONLY)**



## 10.0 PACKAGING INFORMATION

### 10.1 Package Marking Information

#### 8-Lead PDIP



#### Example



#### 8-Lead SOIC



#### Example



|                |        |  |
|----------------|--------|--|
| <b>Legend:</b> | XX...X | Customer specific information*             |
|                | Y      | Year code (last digit of calendar year)    |
|                | YY     | Year code (last 2 digits of calendar year) |
|                | WW     | Week code (week of January 1 is week '01') |
|                | NNN    | Alphanumeric traceability code             |

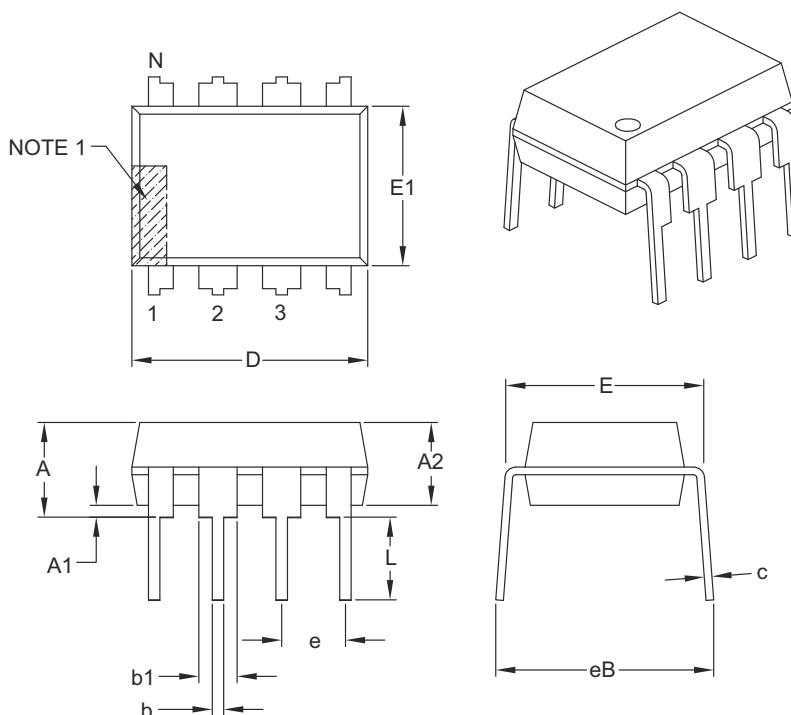
**Note:** In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line thus limiting the number of available characters for customer specific information.

\* Standard PIC MCU device marking consists of Microchip part number, year code, week code, and traceability code. For PIC MCU device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

## 10.2 Package Details

### 8-Lead Plastic Dual In-Line (P) – 300 mil Body [PDIP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



| Dimension Limits           | Units | INCHES   |      |      |
|----------------------------|-------|----------|------|------|
|                            |       | MIN      | NOM  | MAX  |
| Number of Pins             | N     | 8        |      |      |
| Pitch                      | e     | .100 BSC |      |      |
| Top to Seating Plane       | A     | –        | –    | .210 |
| Molded Package Thickness   | A2    | .115     | .130 | .195 |
| Base to Seating Plane      | A1    | .015     | –    | –    |
| Shoulder to Shoulder Width | E     | .290     | .310 | .325 |
| Molded Package Width       | E1    | .240     | .250 | .280 |
| Overall Length             | D     | .348     | .365 | .400 |
| Tip to Seating Plane       | L     | .115     | .130 | .150 |
| Lead Thickness             | c     | .008     | .010 | .015 |
| Upper Lead Width           | b1    | .040     | .060 | .070 |
| Lower Lead Width           | b     | .014     | .018 | .022 |
| Overall Row Spacing §      | eB    | –        | –    | .430 |

**Notes:**

1. Pin 1 visual index feature may vary, but must be located with the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

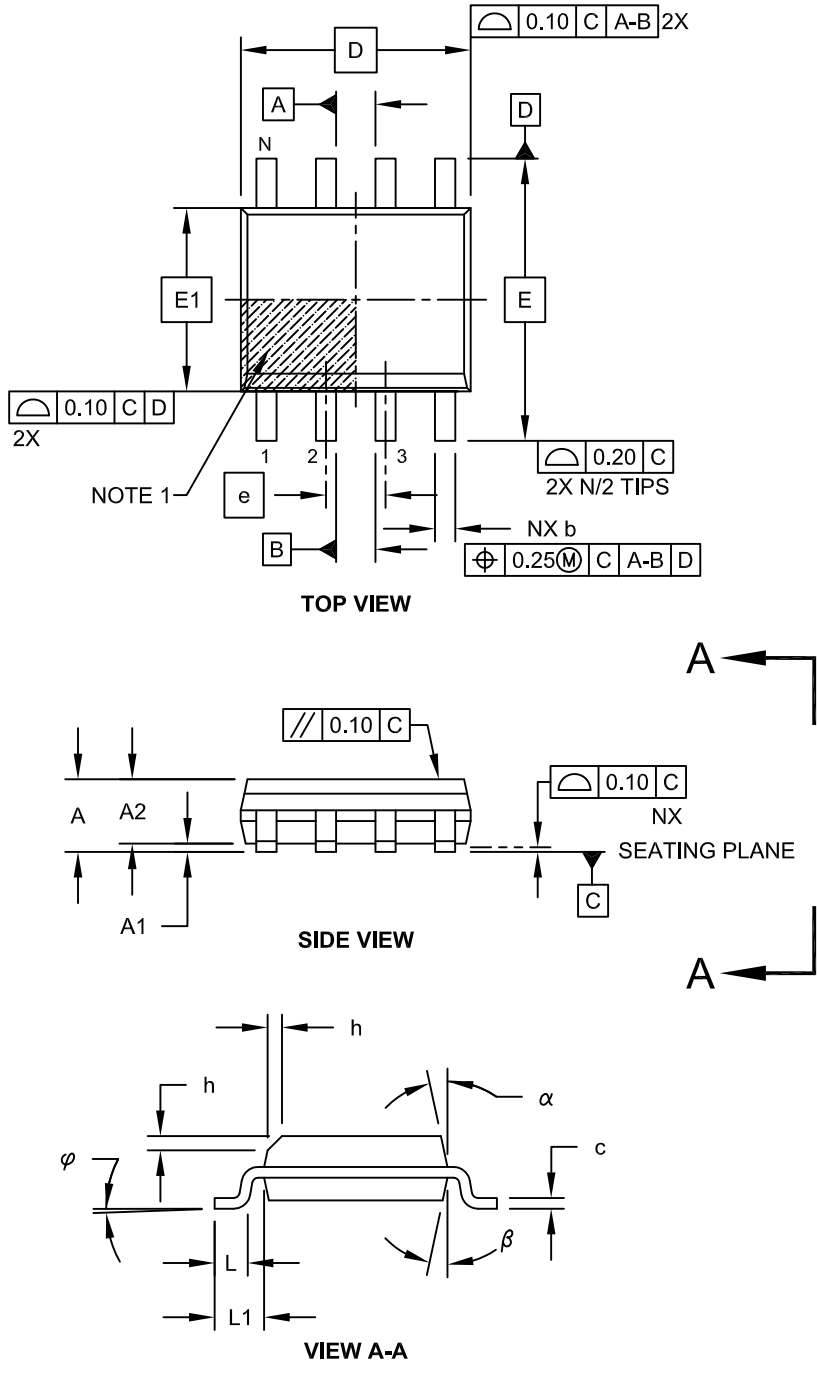
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-018B

# HCS320

## 8-Lead Plastic Small Outline (SN) - Narrow, 3.90 mm Body [SOIC]

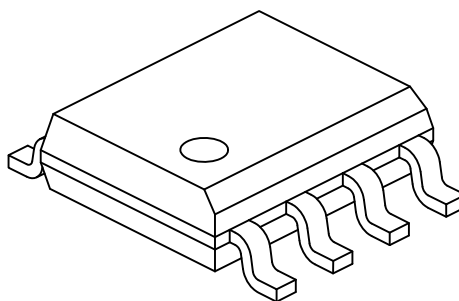
**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



Microchip Technology Drawing No. C04-057C Sheet 1 of 2

## 8-Lead Plastic Small Outline (SN) - Narrow, 3.90 mm Body [SOIC]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



| Dimension Limits         | Units | MILLIMETERS |     |      |
|--------------------------|-------|-------------|-----|------|
|                          |       | MIN         | NOM | MAX  |
| Number of Pins           | N     | 8           |     |      |
| Pitch                    | e     | 1.27 BSC    |     |      |
| Overall Height           | A     | -           | -   | 1.75 |
| Molded Package Thickness | A2    | 1.25        | -   | -    |
| Standoff §               | A1    | 0.10        | -   | 0.25 |
| Overall Width            | E     | 6.00 BSC    |     |      |
| Molded Package Width     | E1    | 3.90 BSC    |     |      |
| Overall Length           | D     | 4.90 BSC    |     |      |
| Chamfer (Optional)       | h     | 0.25        | -   | 0.50 |
| Foot Length              | L     | 0.40        | -   | 1.27 |
| Footprint                | L1    | 1.04 REF    |     |      |
| Foot Angle               | φ     | 0°          | -   | 8°   |
| Lead Thickness           | c     | 0.17        | -   | 0.25 |
| Lead Width               | b     | 0.31        | -   | 0.51 |
| Mold Draft Angle Top     | α     | 5°          | -   | 15°  |
| Mold Draft Angle Bottom  | β     | 5°          | -   | 15°  |

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15mm per side.
- Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

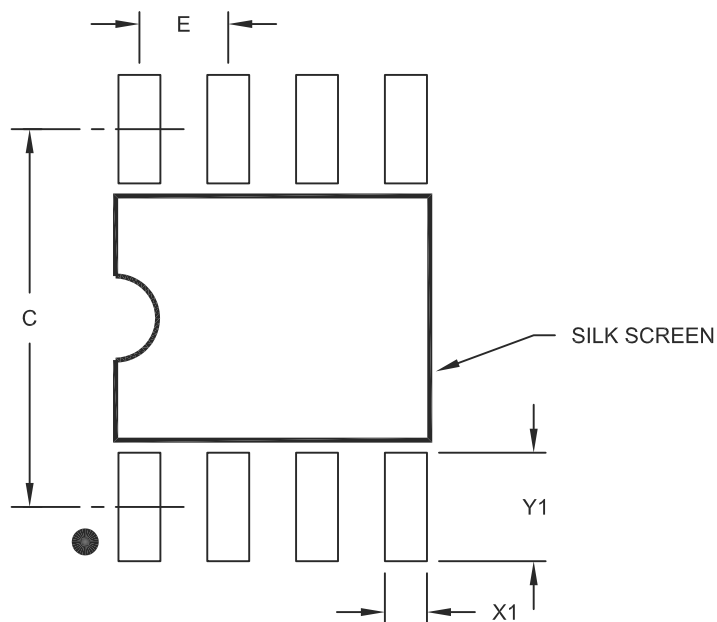
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing No. C04-057C Sheet 2 of 2

# HCS320

## 8-Lead Plastic Small Outline (SN) – Narrow, 3.90 mm Body [SOIC]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

| Dimension Limits        | Units | MILLIMETERS |      |      |
|-------------------------|-------|-------------|------|------|
|                         |       | MIN         | NOM  | MAX  |
| Contact Pitch           | E     | 1.27 BSC    |      |      |
| Contact Pad Spacing     | C     |             | 5.40 |      |
| Contact Pad Width (X8)  | X1    |             |      | 0.60 |
| Contact Pad Length (X8) | Y1    |             |      | 1.55 |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2057A



## APPENDIX A: 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.

## REVISION HISTORY

### Revision D (June 2011)

- Updated the following sections: Development Support, The Microchip Web Site, Reader Response and HCS320 Product Identification System
- Added new section **Appendix A**
- Minor formatting and text changes were incorporated throughout the document

## THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at [www.microchip.com](http://www.microchip.com). This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

## CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at [www.microchip.com](http://www.microchip.com). Under "Support", click on "Customer Change Notification" and follow the registration instructions.

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

**Technical support is available through the web site at: <http://microchip.com/support>**

## READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

TO: Technical Publications Manager Total Pages Sent \_\_\_\_\_

RE: Reader Response

From: Name \_\_\_\_\_

Company \_\_\_\_\_

Address \_\_\_\_\_

City / State / ZIP / Country \_\_\_\_\_

Telephone: (\_\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_ FAX: (\_\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_

Application (optional):

Would you like a reply?  Y  N

Device: HCS320

Literature Number: DS41097D

Questions:

1. What are the best features of this document?

\_\_\_\_\_  
\_\_\_\_\_

2. How does this document meet your hardware and software development needs?

\_\_\_\_\_  
\_\_\_\_\_

3. Do you find the organization of this document easy to follow? If not, why?

\_\_\_\_\_  
\_\_\_\_\_

4. What additions to the document do you think would enhance the structure and subject?

\_\_\_\_\_  
\_\_\_\_\_

5. What deletions from the document could be made without affecting the overall usefulness?

\_\_\_\_\_  
\_\_\_\_\_

6. Is there any incorrect or misleading information (what and where)?

\_\_\_\_\_  
\_\_\_\_\_

7. How would you improve this document?

\_\_\_\_\_  
\_\_\_\_\_

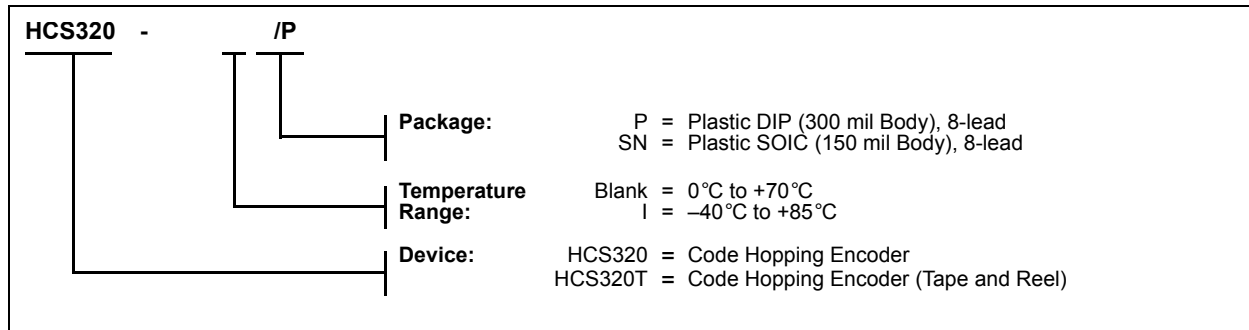
# HCS320

---

---

## HCS320 PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.



---

**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, PIC<sup>32</sup> logo, rPIC 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, 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, rLAB, 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.

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

 Printed on recycled paper.

ISBN: 978-1-61341-229-9

*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.*

---

**QUALITY MANAGEMENT SYSTEM  
CERTIFIED BY DNV  
== ISO/TS 16949:2002 ==**



# 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](http://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



Компания «ЭлектроПласт» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов;
- Поставка более 17-ти миллионов наименований электронных компонентов;
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- Лицензия ФСБ на осуществление работ с использованием сведений, составляющих государственную тайну;
- Поставка специализированных компонентов (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Aeroflex, Peregrine, Syfer, Eurofarad, Texas Instrument, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Помимо этого, одним из направлений компании «ЭлектроПласт» является направление «Источники питания». Мы предлагаем Вам помощь Конструкторского отдела:

- Подбор оптимального решения, техническое обоснование при выборе компонента;
- Подбор аналогов;
- Консультации по применению компонента;
- Поставка образцов и прототипов;
- Техническая поддержка проекта;
- Защита от снятия компонента с производства.



#### Как с нами связаться

**Телефон:** 8 (812) 309 58 32 (многоканальный)

**Факс:** 8 (812) 320-02-42

**Электронная почта:** [org@eplast1.ru](mailto:org@eplast1.ru)

**Адрес:** 198099, г. Санкт-Петербург, ул. Калинина, дом 2, корпус 4, литера А.