OVERVIEW

All of the KEELOQ Encoders share the same standard code word format composed of a basic set of 65 bits of information. Although, the Advanced Encoders (e.g. HCS360, HCS361) supplement that set with extra bits of information along with a pair of CRC bits (Cyclic Redundancy Check). This Technical Brief presents a routine in PICmicro® microcontroller (MCU) assembly language that implements the CRC polynomial as used by Advanced KEELOQ Encoders and allows a KEELOQ receiver to verify the received code word.

KEELOQ CRC EQUATIONS

The CRC bits are calculated on the 65 previous transmitted bits using the following equations:

\[ CRC[1]_{n+1} = CRC[0]_n \oplus D_i \]

and

\[ CRC[0]_{n+1} = (CRC[0]_n \oplus D_i) \oplus CRC[1]_n \]

with

\[ CRC[0]_0 = 0; \quad CRC[1]_0 = 0; \]

where \( D_i \) represents the \( n \)th transmission bit of the code word with 0<\( n \)<64

PICmicro® MCU IMPLEMENTATION

The CRC calculation would be ideally done on the fly by the same routine that is receiving the data from the radio input. Therefore, it would compute a new CRC value as every data bit gets shifted in. After receiving the first 65 bits, the computed CRC value is ready to be compared with the transmitted CRC bits that are following in the data stream. If the CRC calculation is further extended to include the transmitted CRC bits too (all first 67 bits), a valid transmission will result in a CRC value of 00.

The assembler implementation of the CRC equations can be optimized by the use of the decrement instruction. The core code segment can be expressed simply as follows:

```
DECF CRC,W
BTFSC CRC,1
XORLW 3
BTFSS STATUS,C
XORLW 3
MOVWF CRC
```

For completeness and testing purposes, the code presented in Appendix A shows a complete stand-alone routine implementing the CRC calculation on a buffer containing 9 bytes (65 + 2 bits) of data.
The core segment is composed of the few lines of code from Listing 1 and is therefore suitable for inline insertion in any standard receive routine.

The stand-alone version includes a brief initialization code and a loop (67 times) around the CRC code, while shifting all the bits out of the buffer. The return value in the CRC variable (2 LSb) contains the CRC check result.

The source code presented in Appendix A can be compiled using MPASM™ 2.50 for any PICmicro MCU with 12-bit and 14-bit cores (PIC16C5X and PIC16CXXX family).

The only modification required for use with PIC17CXXX and PIC18CXXX families (16-bit and enhanced PICmicro MCU cores) is related to the use of the RRF instruction whose mnemonic changes in RRCF.

**C LANGUAGE IMPLEMENTATION**

Using the C programming language, the CRC equations can be effectively expressed in the following three lines of code:

```
TEMP = CRC1;
CRC1 = CRC0 ^ Di;
CRC0 = CRC1 ^ TEMP;
```

where \(D_i\) represent the \(i\)th transmission bit of the code word and \(TEMP\) is a temporary variable.

The compiler specific efficiency might actually turn out to have quite an impact on the performance results of such implementation depending on the compiler ability to manipulate bit variables.

**MEMORY USAGE**

(Assembly implementation only)

Program memory: 6 words

RAM: 1 byte

**REFERENCES**

- HCS360 Data Sheet DS40152
- HCS361 Data Sheet DS40146
- HCS362 Data Sheet DS40189
- HCS365 Data Sheet DS41109
- HCS410 Data Sheet DS40158
- HCS412 Data Sheet DS41099
- HCS473 Data Sheet DS40035
- AN730 CRC Generating and Checking DS00730
- TB001 Secure Learning RKE Systems using KEEL0Q Technology DS91000
- TB003 Introduction to KEEL0Q Technology DS91002
- TB030 KEEL0Q Decryption & IFF Algorithms DS91030
- TB041 KEEL0Q Decryption Routines in C DS90041
- TB042 Interfacing a KEEL0Q Encoder to a PLL Circuit DS90042

**KEYWORDS**

KEEL0Q, CRC, Receiver
APPENDIX A: SOURCE CODE

;---------------------------------------------
; Keeloq CRC verification routine
;
; version 1.0  01/10/2001 Lucio Di Jasio, Myron Loewen
;
; INPUT:
; Buffer[0..8]  67+ bit HCS code word (left aligned)
;
; OUTPUT:
; CRC            computed CRC check result (two lsb bits only)
;
; USES:
; FSR            indirect pointer
; Aux            shift out data bits
; Count          loop counter
;
; CRCcheck
clr CRC           ; start with 0,0
movlw Buffer0    ; point to first byte
clr Count
movf INDF,W
movwf Aux

CRCLoop
rrf Aux,F        ; rotate data bit in CARRY

;---- this segment can be inline in receive routine

decf CRC,W        ; apply the CRC equations
btfsc CRC,1
xorlw 3
btfss STATUS,C
xorlw 3
movwf CRC

;--------

incf Count,F     ; count the bit
movlw 7
andwf Count,W
BNZ CRCLE
incf FSR,F

movf INDF,W
movwf Aux

CRCLE
movlw .67        ; repeat 65 times
subwf Count,W
SKPZ
goto CRCLoop
retlw 0

Software License Agreement

The software supplied herewith by Microchip Technology Incorporated (the “Company”) for its PICmicro® Microcontroller is intended and supplied to you, the Company’s customer, for use solely and exclusively on Microchip PICmicro Microcontroller products.

The software is owned by the Company and/or its supplier, and is protected under applicable copyright laws. All rights are reserved. Any use in violation of the foregoing restrictions may subject the user to criminal sanctions under applicable laws, as well as to civil liability for the breach of the terms and conditions of this license.

THIS SOFTWARE IS PROVIDED IN AN “AS IS” CONDITION. NO WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE APPLY TO THIS SOFTWARE. THE COMPANY SHALL NOT, IN ANY CIRCUMSTANCES, BE LIABLE FOR SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES, FOR ANY REASON WHATSOEVER.
“All rights reserved. Copyright © 2001, Microchip Technology Incorporated, USA. Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights. The Microchip logo and name are registered trademarks of Microchip Technology Inc. in the U.S.A. and other countries. All rights reserved. All other trademarks mentioned herein are the property of their respective companies. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.”

**Trademarks**

The Microchip name, logo, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, KEELOQ, SEEVAL, MPLAB and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

Total Endurance, ICSP, In-Circuit Serial Programming, FilterLab, MXDEV, microID, FlexROM, fuzzyLAB, MPASM, MPLINK, MPLIB, PICDEM, ICEPIC, Migratable Memory, FanSense, ECONOMONITOR, SelectMode and microPort are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Term Programming (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, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company’s quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip’s quality system for the design and manufacture of development systems is ISO 9001 certified.