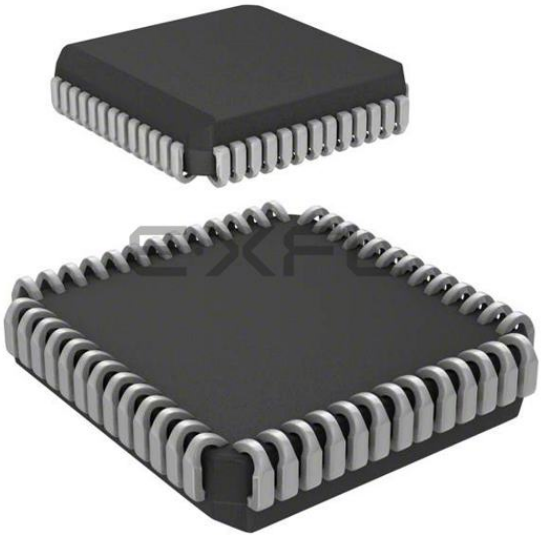


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	HC11
Core Size	8-Bit
Speed	3MHz
Connectivity	SCI, SPI
Peripherals	POR, WDT
Number of I/O	38
Program Memory Size	20KB (20K x 8)
Program Memory Type	OTP
EEPROM Size	512 x 8
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	52-LCC (J-Lead)
Supplier Device Package	52-PLCC (19.1x19.1)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc68711e20cfne3

List of Chapters

Chapter 5
Resets and Interrupts

Chapter 9
Timing Systems

Chapter 10
Electrical Characteristics

- Computer operating properly (COP) watchdog system
- 38 general-purpose input/output (I/O) pins:
  - 16 bidirectional I/O pins
  - 11 input-only pins
  - 11 output-only pins
- Several packaging options:
  - 52-pin plastic-leaded chip carrier (PLCC)
  - 52-pin windowed ceramic leaded chip carrier (CLCC)
  - 52-pin plastic thin quad flat pack, 10 mm x 10 mm (TQFP)
  - 64-pin quad flat pack (QFP)
  - 48-pin plastic dual in-line package (DIP), MC68HC811E2 only
  - 56-pin plastic shrink dual in-line package, .070-inch lead spacing (SDIP)

1.3 Structure

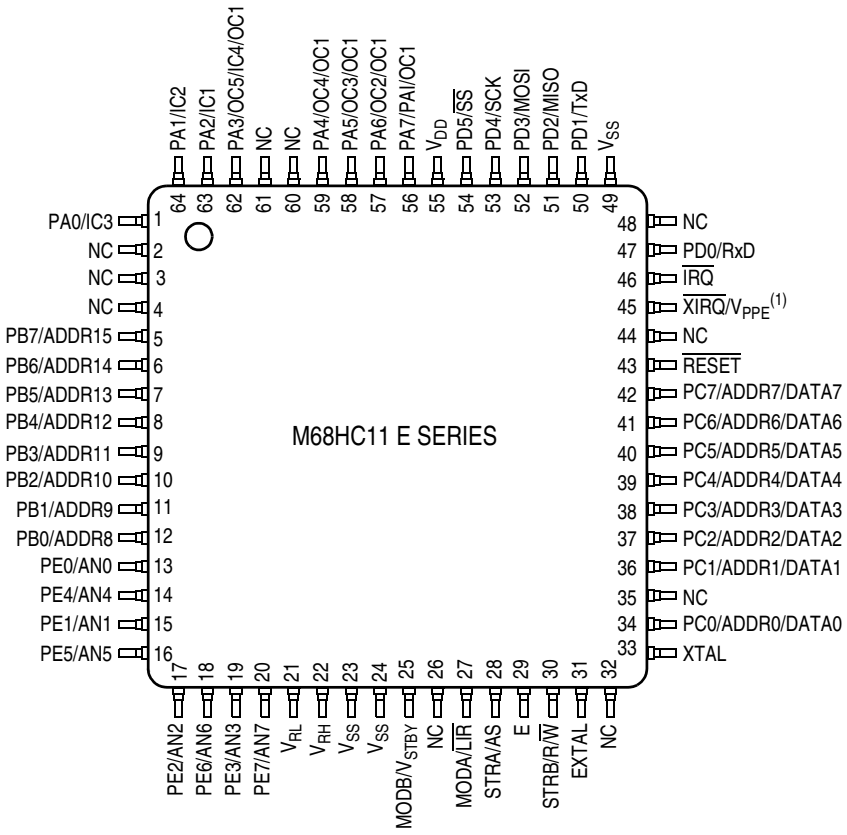
See Figure 1-1 for a functional diagram of the E-series MCUs. Differences among devices are noted in the table accompanying Figure 1-1.

1.4 Pin Descriptions

M68HC11 E-series MCUs are available packaged in:

- 52-pin plastic-leaded chip carrier (PLCC)
- 52-pin windowed ceramic leaded chip carrier (CLCC)
- 52-pin plastic thin quad flat pack, 10 mm x 10 mm (TQFP)
- 64-pin quad flat pack (QFP)
- 48-pin plastic dual in-line package (DIP), MC68HC811E2 only
- 56-pin plastic shrink dual in-line package, .070-inch lead spacing (SDIP)

Most pins on these MCUs serve two or more functions, as described in the following paragraphs. Refer to Figure 1-2, Figure 1-3, Figure 1-4, Figure 1-5, and Figure 1-6 which show the M68HC11 E-series pin assignments for the PLCC/CLCC, QFP, TQFP, SDIP, and DIP packages.



1. V_{PPE} applies only to devices with EPROM/OTPROM.

Figure 1-3. Pin Assignments for 64-Pin QFP

Addr.	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
$1031	Analog-to-Digital Results Register 1 (ADR1) See page 64.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Indeterminate after reset							
$1032	Analog-to-Digital Results Register 2 (ADR2) See page 64.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Indeterminate after reset							
$1033	Analog-to-Digital Results Register 3 (ADR3) See page 64.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Indeterminate after reset							
$1034	Analog-to-Digital Results Register 4 (ADR4) See page 64.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Indeterminate after reset							
$1035	Block Protect Register (BPROT) See page 52.	Read:				PTCON	BPRT3	BPRT2	BPRT1	BPRT0
		Write:								
		Reset:	0	0	0	1	1	1	1	1
$1036	EPROM Programming Control Register (EPROG)(1) See page 53.	Read:	MBE		ELAT	EXCOL	EXROW	T1	T0	PGM
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$1037	Reserved		R	R	R	R	R	R	R	R

1. MC68HC711E20 only

$1038	Reserved		R	R	R	R	R	R	R	R
$1039	System Configuration Options Register (OPTION) See page 46.	Read:	ADPU	CSEL	IRQE(1)	DLY(1)	CME		CR1(1)	CR0(1)
		Write:								
		Reset:	0	0	0	1	0	0	0	0
$103A	Arm/Reset COP Timer Circuitry Register (COPRST) See page 81.	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$103B	EPROM and EEPROM Programming Control Register (PPROG) See page 49.	Read:	ODD	EVEN	ELAT(2)	BYTE	ROW	ERASE	EELAT	EPGM
		Write:								
		Reset:	0	0	0	0	0	0	0	0
$103C	Highest Priority I Bit Interrupt and Miscellaneous Register (HPRIO) See page 41.	Read:	RBOOT	SMOD	MDA	IRV(NE)	PSEL3	PSEL2	PSEL1	PSEL0
		Write:								
		Reset:	0	0	0	0	0	1	1	0
$103D	RAM and I/O Mapping Register (INIT) See page 45.	Read:	RAM3	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0
		Write:								
		Reset:	0	0	0	0	0	0	0	1

= Unimplemented R = Reserved U = Unaffected

I = Indeterminate after reset

Figure 2-7. Register and Control Bit Assignments (Sheet 5 of 6)

Table 2-4. RAM Mapping

RAM[3:0]	Address
0000	$0000–$0xFF
0001	$1000–$1xFF
0010	$2000–$2xFF
0011	$3000–$3xFF
0100	$4000–$4xFF
0101	$5000–$5xFF
0110	$6000–$6xFF
0111	$7000–$7xFF
1000	$8000–$8xFF
1001	$9000–$9xFF
1010	$A000–$AxFF
1011	$B000–$BxFF
1100	$C000–$CxFF
1101	$D000–$DxFF
1110	$E000–$ExFF
1111	$F000–$FxFF

Table 2-5. Register Mapping

REG[3:0]	Address
0000	$0000–$003F
0001	$1000–$103F
0010	$2000–$203F
0011	$3000–$303F
0100	$4000–$403F
0101	$5000–$503F
0110	$6000–$603F
0111	$7000–$703F
1000	$8000–$803F
1001	$9000–$903F
1010	$A000–$A03F
1011	$B000–$B03F
1100	$C000–$C03F
1101	$D000–$D03F
1110	$E000–$E03F
1111	$F000–$F03F

2.3.3.3 System Configuration Options Register

The 8-bit, special-purpose system configuration options register (OPTION) sets internal system configuration options during initialization. The time protected control bits, IRQE, DLY, and CR[1:0], can be written only once after a reset and then they become read-only. This minimizes the possibility of any accidental changes to the system configuration.

Address: $1039

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	ADPU	CSEL	IRQE(1)	DLY(1)	CME		CR1(1)	CR0(1)
Reset:	0	0	0	1	0	0	0	0

1. Can be written only once in first 64 cycles out of reset in normal modes or at any time during special modes.

= Unimplemented

Figure 2-13. System Configuration Options Register (OPTION)

ADPU — Analog-to-Digital Converter Power-Up Bit

Refer to Chapter 3 Analog-to-Digital (A/D) Converter.

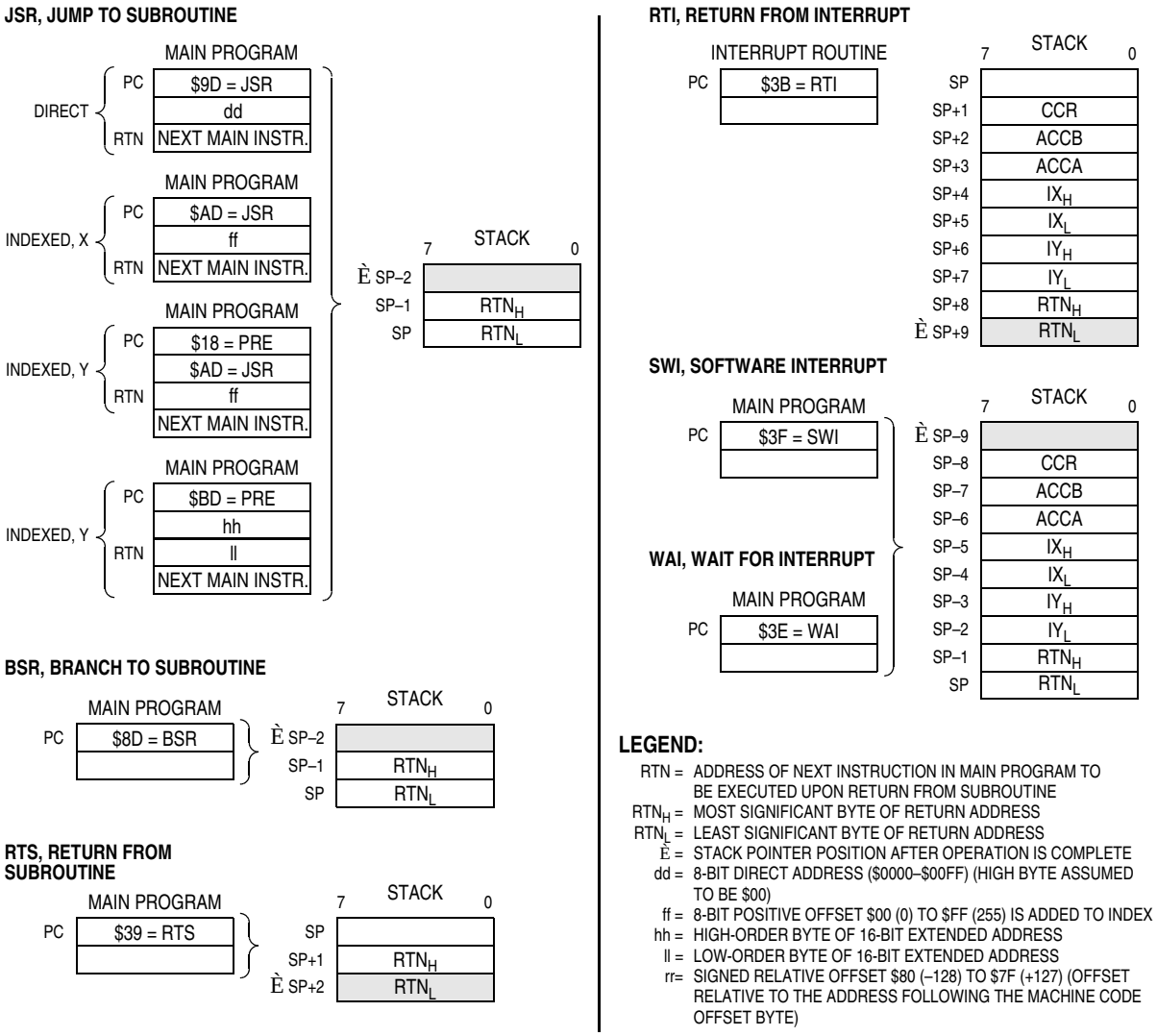
CSEL — Clock Select Bit

Selects alternate clock source for on-chip EEPROM charge pump. Refer to 2.5.1 EEPROM and CONFIG Programming and Erasure for more information on EEPROM use.

CSEL also selects the clock source for the A/D converter, a function discussed in Chapter 3 Analog-to-Digital (A/D) Converter.

At the end of the interrupt service routine, an return-from interrupt (RTI) instruction is executed. The RTI instruction causes the saved registers to be pulled off the stack in reverse order. Program execution resumes at the return address.

Certain instructions push and pull the A and B accumulators and the X and Y index registers and are often used to preserve program context. For example, pushing accumulator A onto the stack when entering a subroutine that uses accumulator A and then pulling accumulator A off the stack just before leaving the subroutine ensures that the contents of a register will be the same after returning from the subroutine as it was before starting the subroutine.



LEGEND:

RTN = ADDRESS OF NEXT INSTRUCTION IN MAIN PROGRAM TO BE EXECUTED UPON RETURN FROM SUBROUTINE
RTN_H = MOST SIGNIFICANT BYTE OF RETURN ADDRESS
RTN_L = LEAST SIGNIFICANT BYTE OF RETURN ADDRESS
È = STACK POINTER POSITION AFTER OPERATION IS COMPLETE
dd = 8-BIT DIRECT ADDRESS ($0000–$00FF) (HIGH BYTE ASSUMED TO BE $00)
ff = 8-BIT POSITIVE OFFSET $00 (0) TO $FF (255) IS ADDED TO INDEX
hh = HIGH-ORDER BYTE OF 16-BIT EXTENDED ADDRESS
ll = LOW-ORDER BYTE OF 16-BIT EXTENDED ADDRESS
rr= SIGNED RELATIVE OFFSET $80 (–128) TO $7F (+127) (OFFSET RELATIVE TO THE ADDRESS FOLLOWING THE MACHINE CODE OFFSET BYTE)

Figure 4-2. Stacking Operations

4.4 Opcodes and Operands

The M68HC11 Family of microcontrollers uses 8-bit opcodes. Each opcode identifies a particular instruction and associated addressing mode to the CPU. Several opcodes are required to provide each instruction with a range of addressing capabilities. Only 256 opcodes would be available if the range of values were restricted to the number able to be expressed in 8-bit binary numbers.

A 4-page opcode map has been implemented to expand the number of instructions. An additional byte, called a prebyte, directs the processor from page 0 of the opcode map to one of the other three pages. As its name implies, the additional byte precedes the opcode.

A complete instruction consists of a prebyte, if any, an opcode, and zero, one, two, or three operands. The operands contain information the CPU needs for executing the instruction. Complete instructions can be from one to five bytes long.

4.5 Addressing Modes

Six addressing modes can be used to access memory:

- Immediate
- Direct
- Extended
- Indexed
- Inherent
- Relative

These modes are detailed in the following paragraphs. All modes except inherent mode use an effective address. The effective address is the memory address from which the argument is fetched or stored or the address from which execution is to proceed. The effective address can be specified within an instruction, or it can be calculated.

4.5.1 Immediate

In the immediate addressing mode, an argument is contained in the byte(s) immediately following the opcode. The number of bytes following the opcode matches the size of the register or memory location being operated on. There are 2-, 3-, and 4- (if prebyte is required) byte immediate instructions. The effective address is the address of the byte following the instruction.

4.5.2 Direct

In the direct addressing mode, the low-order byte of the operand address is contained in a single byte following the opcode, and the high-order byte of the address is assumed to be $00. Addresses $00–$FF are thus accessed directly, using 2-byte instructions. Execution time is reduced by eliminating the additional memory access required for the high-order address byte. In most applications, this 256-byte area is reserved for frequently referenced data. In M68HC11 MCUs, the memory map can be configured for combinations of internal registers, RAM, or external memory to occupy these addresses.

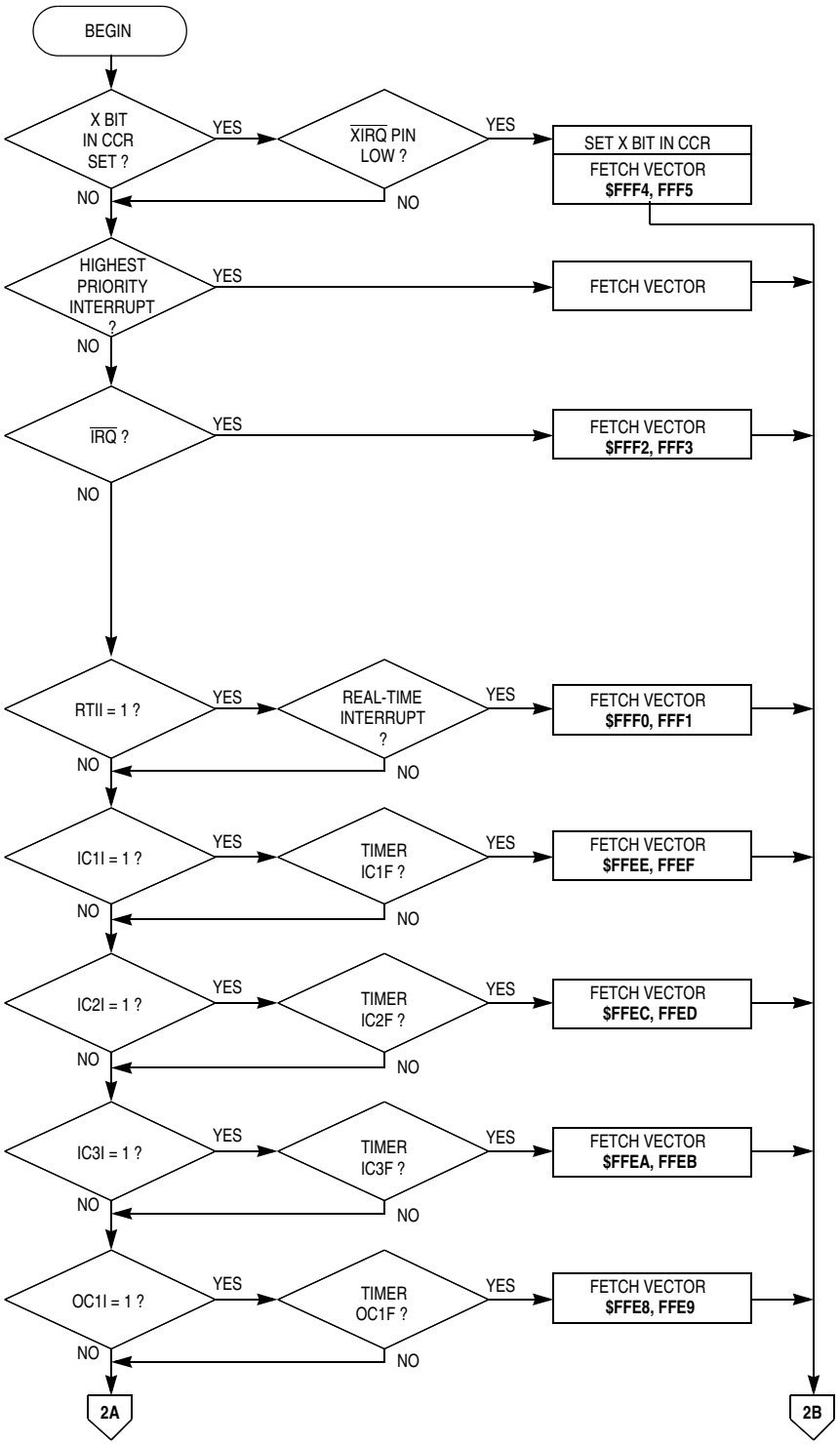


Figure 5-6. Interrupt Priority Resolution (Sheet 1 of 2)

Table 9-1. Timer Summary

Control Bits PR1, PR0	XTAL Frequencies			
	4.0 MHz	8.0 MHz	12.0 MHz	Other Rates
	1.0 MHz	2.0 MHz	3.0 MHz	(E)
	1000 ns	500 ns	333 ns	(1/E)
	Main Timer Count Rates			
0 0 1 count — overflow —	1000 ns 65.536 ms	500 ns 32.768 ms	333 ns 21.845 ms	(E/1) ($E/2^{16}$)
0 1 1 count — overflow —	4.0 µs 262.14 ms	2.0 µs 131.07 ms	1.333 µs 87.381 ms	(E/4) ($E/2^{18}$)
1 0 1 count — overflow —	8.0 µs 524.29 ms	4.0 µs 262.14 ms	2.667 µs 174.76 ms	(E/8) ($E/2^{19}$)
1 1 1 count — overflow —	16.0 µs 1.049 s	8.0 µs 524.29 ms	5.333 µs 349.52 ms	(E/16) ($E/2^{20}$)

9.2 Timer Structure

Figure 9-2 shows the capture/compare system block diagram. The port A pin control block includes logic for timer functions and for general-purpose I/O. For pins PA3, PA2, PA1, and PA0, this block contains both the edge-detection logic and the control logic that enables the selection of which edge triggers an input capture. The digital level on PA[3:0] can be read at any time (read PORTA register), even if the pin is being used for the input capture function. Pins PA[6:3] are used for either general-purpose I/O, or as output compare pins. When one of these pins is being used for an output compare function, it cannot be written directly as if it were a general-purpose output. Each of the output compare functions (OC[5:2]) is related to one of the port A output pins. Output compare one (OC1) has extra control logic, allowing it optional control of any combination of the PA[7:3] pins. The PA7 pin can be used as a general-purpose I/O pin, as an input to the pulse accumulator, or as an OC1 output pin.

9.3 Input Capture

The input capture function records the time an external event occurs by latching the value of the free-running counter when a selected edge is detected at the associated timer input pin. Software can store latched values and use them to compute the periodicity and duration of events. For example, by storing the times of successive edges of an incoming signal, software can determine the period and pulse width of a signal. To measure period, two successive edges of the same polarity are captured. To measure pulse width, two alternate polarity edges are captured.

In most cases, input capture edges are asynchronous to the internal timer counter, which is clocked relative to an internal clock (PH2). These asynchronous capture requests are synchronized to PH2 so that the latching occurs on the opposite half cycle of PH2 from when the timer counter is being incremented. This synchronization process introduces a delay from when the edge occurs to when the counter value is detected. Because these delays offset each other when the time between two edges is being measured, the delay can be ignored. When an input capture is being used with an output compare, there is a similar delay between the actual compare point and when the output pin changes state.

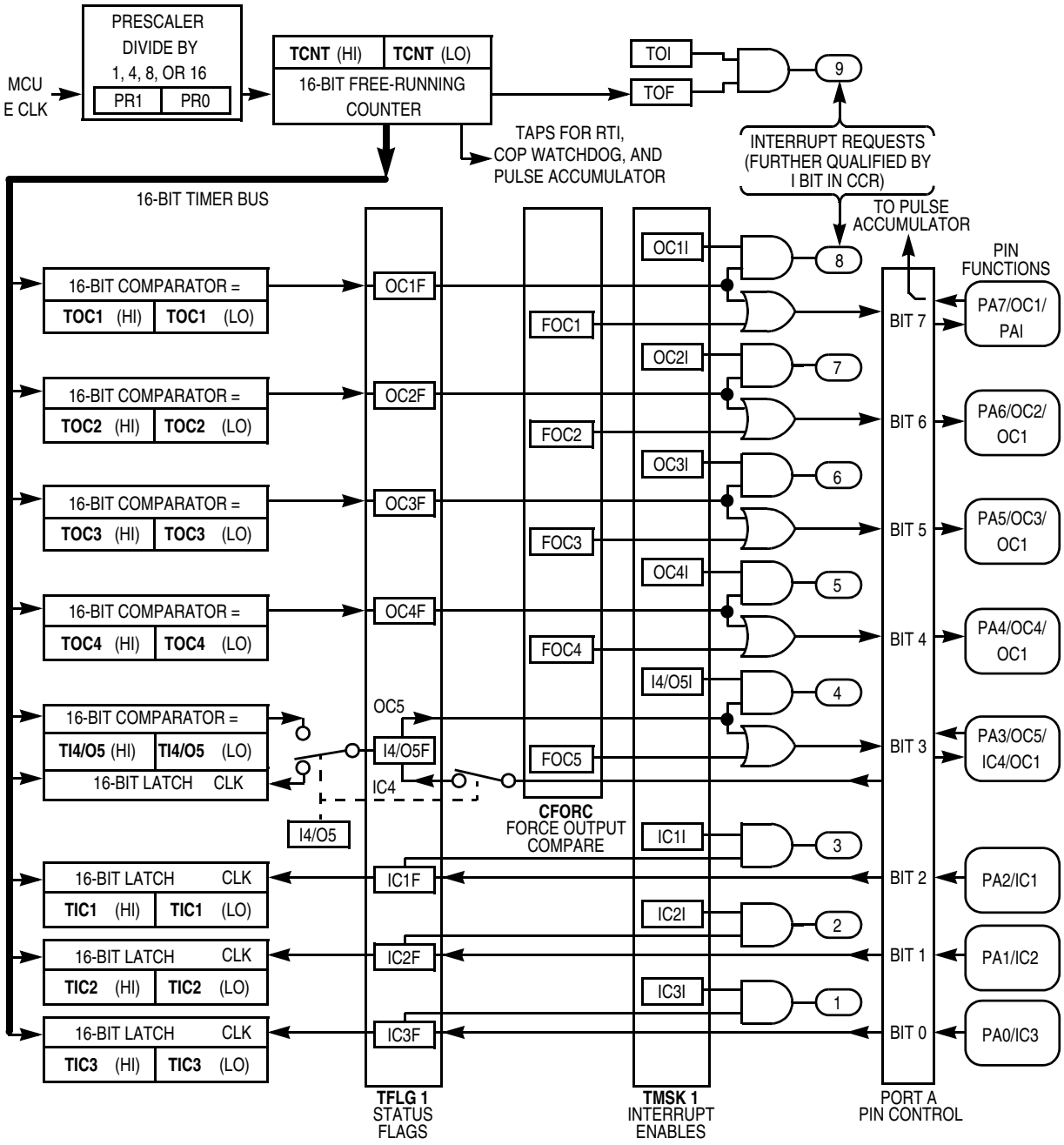


Figure 9-2. Capture/Compare Block Diagram

independent of the software latencies associated with flag clearing and service. For this reason, an RTI period starts from the previous timeout, not from when RTIF is cleared.

Every timeout causes the RTIF bit in TFLG2 to be set, and if RTII is set, an interrupt request is generated. After reset, one entire RTI period elapses before the RTIF is set for the first time. Refer to the 9.4.9 Timer Interrupt Mask 2 Register, 9.5.2 Timer Interrupt Flag Register 2, and 9.5.3 Pulse Accumulator Control Register.

9.5.1 Timer Interrupt Mask Register 2

This register contains the real-time interrupt enable bits.

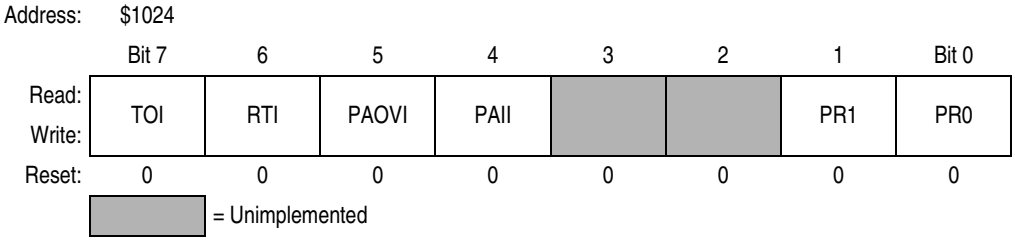


Figure 9-21. Timer Interrupt Mask 2 Register (TMSK2)

TOI — Timer Overflow Interrupt Enable Bit
0 = TOF interrupts disabled
1 = Interrupt requested when TOF is set to 1

RTII — Real-Time Interrupt Enable Bit
0 = RTIF interrupts disabled
1 = Interrupt requested when RTIF set to 1

PAOVI — Pulse Accumulator Overflow Interrupt Enable Bit
Refer to 9.7 Pulse Accumulator.

PAII — Pulse Accumulator Input Edge Bit
Refer to 9.7 Pulse Accumulator.

Bits [3:2] — Unimplemented
Always read 0

PR[1:0] — Timer Prescaler Select Bits
Refer to Table 9-4.

NOTE

Bits in TMSK2 correspond bit for bit with flag bits in TFLG2. Bits in TMSK2 enable the corresponding interrupt sources.

9.5.2 Timer Interrupt Flag Register 2

Bits of this register indicate the occurrence of timer system events. Coupled with the four high-order bits of TMSK2, the bits of TFLG2 allow the timer subsystem to operate in either a polled or interrupt driven system. Each bit of TFLG2 corresponds to a bit in TMSK2 in the same position.

Address: $1025

	Bit 7	6	5	4	3	2	1	Bit 0
Read: / Write:	TOF	RTIF	PAOVF	PAIF				
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 9-22. Timer Interrupt Flag 2 Register (TFLG2)

Clear flags by writing a 1 to the corresponding bit position(s).

TOF — Timer Overflow Interrupt Flag
Set when TCNT changes from $FFFF to $0000

RTIF — Real-Time Interrupt Flag
The RTIF status bit is automatically set to 1 at the end of every RTI period. To clear RTIF, write a byte to TFLG2 with bit 6 set.

PAOVF — Pulse Accumulator Overflow Interrupt Flag
Refer to 9.7 Pulse Accumulator.

PAIF — Pulse Accumulator Input Edge Interrupt Flag
Refer to 9.7 Pulse Accumulator.

Bits [3:0] — Unimplemented
Always read 0

9.5.3 Pulse Accumulator Control Register

Bits RTR[1:0] of this register select the rate for the RTI system. The remaining bits control the pulse accumulator and IC4/OC5 functions.

Address: $1026

	Bit 7	6	5	4	3	2	1	Bit 0
Read: / Write:	DDRA7	PAEN	PAMOD	PEDGE	DDRA3	I4/O5	RTR1	RTR0
Reset:	0	0	0	0	0	0	0	0

Figure 9-23. Pulse Accumulator Control Register (PACTL)

DDRA7 — Data Direction for Port A Bit 7
Refer to Chapter 6 Parallel Input/Output (I/O) Ports.

PAEN — Pulse Accumulator System Enable Bit
Refer to 9.7 Pulse Accumulator.

PAMOD — Pulse Accumulator Mode Bit
Refer to 9.7 Pulse Accumulator.

10.19 EEPROM Characteristics

Characteristic[1]	Temperature Range			Unit
	–40 to 85°C	–40 to 105°C	–40 to 125°C	
Programming time[2] < 1.0 MHz, RCO enabled 1.0 to 2.0 MHz, RCO disabled ≥ 2.0 MHz (or anytime RCO enabled)	 10 20 10	 15 Must use RCO 15	 20 Must use RCO 20	ms
Erase time[2] Byte, row, and bulk	10	10	10	ms
Write/erase endurance	10,000	10,000	10,000	Cycles
Data retention	10	10	10	Years

1. V_{DD} = 5.0 Vdc ±10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H
2. The RC oscillator (RCO) must be enabled (by setting the CSEL bit in the OPTION register) for EEPROM programming and erasure when the E-clock frequency is below 1.0 MHz.

10.20 MC68L11E9/E20 EEPROM Characteristics

Characteristic[1]	Temperature Range –20 to 70°C	Unit
Programming time[2] 3 V, E ≤ 2.0 MHz, RCO enabled 5 V, E ≤ 2.0 MHz, RCO enabled	 25 10	ms
Erase time[2] (byte, row, and bulk) 3 V, E ≤ 2.0 MHz, RCO enabled 5 V, E ≤ 2.0 MHz, RCO enabled	 25 10	ms
Write/erase endurance	10,000	Cycles
Data retention	10	Years

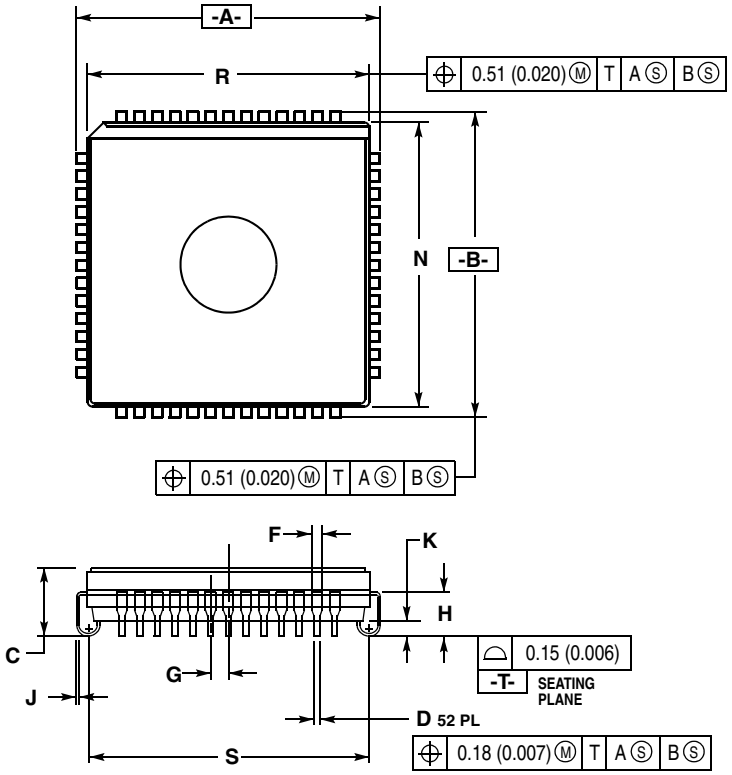
1. V_{DD} = 3.0 Vdc to 5.5 Vdc, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H
2. The RC oscillator (RCO) must be enabled (by setting the CSEL bit in the OPTION register) for EEPROM programming and erasure.

10.21 EPROM Characteristics

Characteristics[1]	Symbol	Min	Typ	Max	Unit
Programming voltage[2]	V_{PPE}	11.75	12.25	12.75	V
Programming current[3]	I_{PPE}	—	3	10	mA
Programming time	t_{EPROG}	2	2	4	ms

1. V_{DD} = 5.0 Vdc ± 10%
2. During EPROM programming of the MC68HC711E9 device, the V_{PPE} pin circuitry may latch-up and be damaged if the input current is not limited to 10 mA. For more information please refer to MC68HC711E9 8-Bit Microcontroller Unit Mask Set Errata 3 (Freescale document order number 68HC711E9MSE3.
3. Typically, a 1-kΩ series resistor is sufficient to limit the programming current for the MC68HC711E9. A 100-Ω series resistor is sufficient to limit the programming current for the MC68HC711E20.

11.6 52-Pin Windowed Ceramic-Leaded Chip Carrier (Case 778B)



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION R AND N DO NOT INCLUDE GLASS PROTRUSION. GLASS PROTRUSION TO BE 0.25 (0.010) MAXIMUM.
4. ALL DIMENSIONS AND TOLERANCES INCLUDE LEAD TRIM OFFSET AND LEAD

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	0.785	0.795	19.94	20.19
B	0.785	0.795	19.94	20.19
C	0.165	0.200	4.20	5.08
D	0.017	0.021	0.44	0.53
F	0.026	0.032	0.67	0.81
G	0.050 BSC		1.27 BSC	
H	0.090	0.130	2.29	3.30
J	0.006	0.010	0.16	0.25
K	0.035	0.045	0.89	1.14
N	0.735	0.756	18.67	19.20
R	0.735	0.756	18.67	19.20
S	0.690	0.730	17.53	18.54

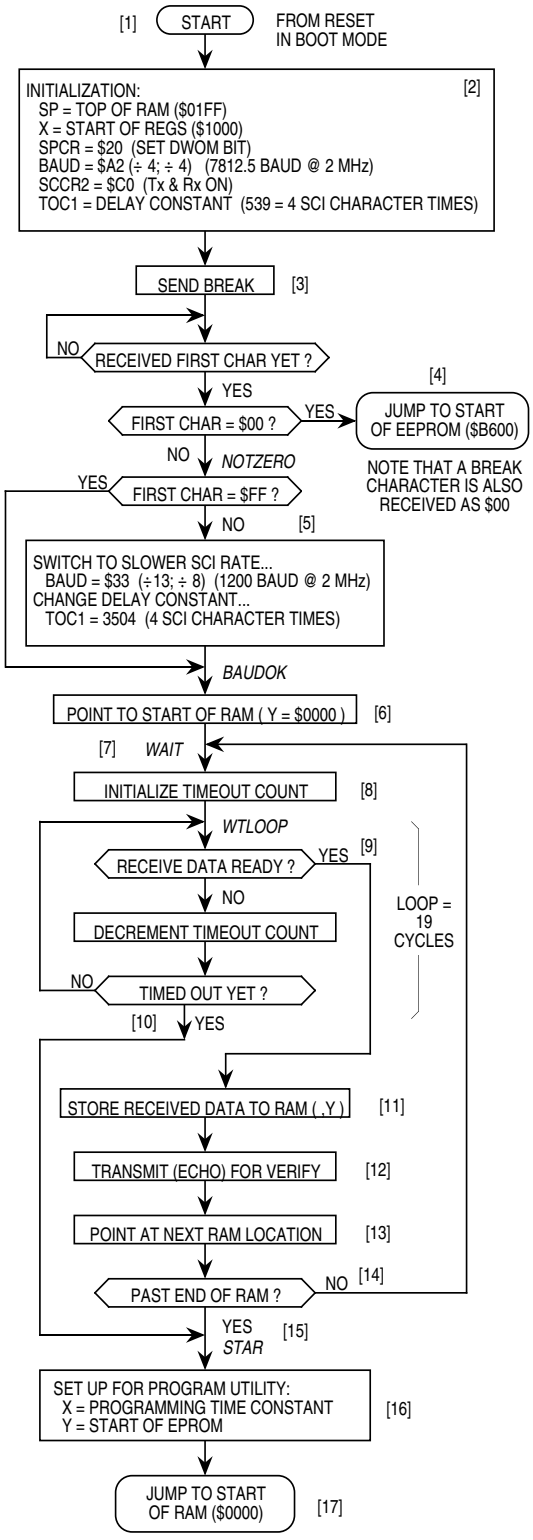


Figure 3. MC68HC711E9 Bootloader Flowchart

The duplicator program in EEPROM clears the DWOM control bit to change port D (thus, TxD) of U3 to normal driven outputs. This configuration will prevent interference due to R9 when TxD from the target MCU (U6) becomes active. Series resistor R9 demonstrates how TxD of U3 can drive RxD of U3[1] and later TxD of U6 can drive RxD of U3 without a destructive conflict between the TxD output buffers.

As the target MCU (U6) leaves reset, its mode pins select bootstrap mode so the bootloader firmware begins executing. A break is sent out the TxD pin of U6. At this time, the TxD pin of U3 is at a driven high so R9 acts as a pullup resistor for TxD of the target MCU (U6). The break character sent from U6 is received by U3 so the duplicator program that is running in the EEPROM of the master MCU knows that the target MCU is ready to accept a bootloaded program.

The master MCU sends a leading $FF character to set the baud rate in the target MCU. Next, the master MCU passes a 3-instruction program to the target MCU and pauses so the bootstrap program in the target MCU will stop the loading process and jump to the start of the downloaded program. This sequence demonstrates the variable-length download feature of the MC68HC711E9 bootloader.

The short program downloaded to the target MCU clears the DWOM bit to change its TxD pin to a normal driven CMOS output and jumps to the EPROM programming utility in the bootstrap ROM of the target MCU.

Note that the small downloaded program did not have to set up the SCI or initialize any parameters for the EPROM programming process. The bootstrap software that ran prior to the loaded program left the SCI turned on and configured in a way that was compatible with the SCI in the master MCU (the duplicator program in the master MCU also did not have to set up the SCI for the same reason). The programming time and starting address for EPROM programming in the target MCU were also set to default values by the bootloader software before jumping to the start of the downloaded program.

Before the EPROM in the target MCU can be programmed, the V_{PP} power supply must be available at the $\overline{\text{XIRQ}}/V_{PPE}$ pin of the target MCU. The duplicator program running in the master MCU monitors this voltage (for presence or absence, not level) at PE7 through resistor divider R14–RI5. The PE7 input was chosen because the internal circuitry for port E pins can tolerate voltages slightly higher than V_{DD}; therefore, resistors R14 and R15 are less critical. No data to be programmed is passed to the target MCU until the master MCU senses that V_{PP} has been stable for about 200 ms.

When V_{PP} is ready, the master MCU turns on the red LED (light-emitting diode) and begins passing data to the target MCU. EPROM Programming Utility explains the activity as data is sent from the master MCU to the target MCU and programmed into the EPROM of the target. The master MCU in the EVBU corresponds to the HOST in the programming utility description and the "PROGRAM utility in MCU" is running in the bootstrap ROM of the target MCU.

Each byte of data sent to the target is programmed and then the programmed location is read and sent back to the master for verification. If any byte fails, the red and green LEDs are turned off, and the programming operation is aborted. If the entire 12 Kbytes are programmed and verified successfully, the red LED is turned off, and the green LED is turned on to indicate success. The programming of all 12 Kbytes takes about 30 seconds.

After a programming operation, the V_{PP} switch (S2) should be turned off before the EVBU power is turned off.

```
8491 '*         DECIMAL TO HEX CONVERSION
8492 '*                 INPUT:  K - INTEGER TO BE CONVERTED
8493 '*                 OUTPUT: HX$ - TWO CHARACTER STRING WITH HEX CONVERSION
8494 '*************************************************************************
8500 IF K > 255 THEN HX$="Too big":GOTO 8530
8510 HX$=MID$(H$,K\16+1,1)            'UPPER NIBBLE
8520 HX$=HX$+MID$(H$,(K MOD 16)+1,1)  'LOWER NIBBLE
8530 RETURN
9499 '******************** BOOT CODE ***************************************
9500 DATA 86, 23                 'LDAA   #$23
9510 DATA B7, 10, 02             'STAA   OPT2    make port C wire or
9520 DATA 86, FE                 'LDAA   #$FE
9530 DATA B7, 10, 03             'STAA   PORTC   light 1 LED on port C bit 0
9540 DATA C6, FF                 'LDAB   #$FF
9550 DATA F7, 10, 07             'STAB   DDRC    make port C outputs
9560 DATA CE, 0F, A0             'LDX    #4000   2msec at 2MHz
9570 DATA 18, CE, E0, 00         'LDY    #$E000  Start of BUFFALO 3.4
9580 DATA 7E, BF, 00             'JMP    $BF00   EPROM routine start address
9590 '*************************************************************************
```

Common Bootstrap Mode Problems

It is not unusual for a user to encounter problems with bootstrap mode because it is new to many users. By knowing some of the common difficulties, the user can avoid them or at least recognize and quickly correct them.

Reset Conditions vs. Conditions as Bootloaded Program Starts

It is common to confuse the reset state of systems and control bits with the state of these systems and control bits when a bootloaded program in RAM starts.

Between these times, the bootloader program is executed, which changes the states of some systems and control bits:

- The SCI system is initialized and turned on (Rx and Tx).
- The SCI system has control of the PD0 and PD1 pins.
- Port D outputs are configured for wire-OR operation.
- The stack pointer is initialized to the top of RAM.
- Time has passed (two or more SCI character times).
- Timer has advanced from its reset count value.

Users also forget that bootstrap mode is a special mode. Thus, privileged control bits are accessible, and write protection for some registers is not in effect. The bootstrap ROM is in the memory map. The DISR bit in the TEST1 control register is set, which disables resets from the COP and clock monitor systems.

Since bootstrap is a special mode, these conditions can be changed by software. The bus can even be switched from single-chip mode to expanded mode to gain access to external memories and peripherals.

```
NEWONE            BF9B *00196  00189
NOTZERO           BF7E *00176  00174
OC1F              0080 *00034  00136 00139
PORTD             0008 *00029  00168
PPROG             003B *00041  00126 00129 00140
PRGROUT           BF13 *00110  00074
PROGDEL           1068 *00063  00205
PROGRAM           BF00 *00074
RAMEND            01FF *00056  00156 00201
RAMSTR            0000 *00055  00184 00207
SCCR2             002D *00038  00162 00167 00169
SCDAT             002F *00040  00091 00118 00122 00145 00172 00197 00199
SCSR              002E *00039  00090 00116 00121 00143 00171 00189
SPCR              0028 *00036  00158
STAR              BFAA *00204  00194
TCNT              000E *00030  00134
TFLG1             0023 *00032  00137 00139
TOC1              0016 *00031  00135 00164 00182 00187
UPLOAD            BF03 *00075
UPLOOP            BF06 *00089  00093
WAIT              BF8E *00186  00202
WAIT1             BF1F *00120  00147
WTLOOP            BF90 *00188  00193

                  Errors: None
                  Labels: 35
     Last Program Address: $BFFF
     Last Storage Address: $0000
            Program Bytes: $0100  256
            Storage Bytes: $0000  0
```