7.15; 7.26; 7.28; 7.32; 7.34; 7.36; 7.37; 7.40; 7.46; 7.47; 7.48; 7.50

7.26
True.

7.28
True, because each 8-bit character also has a start bit and a stop bit for a total of 10 bits. Therefore, the maximum rate is about one character per millisecond.

7.32
True. The ISR must distinguish which device caused the interrupt by testing the flags for the devices that could cause an interrupt.

7.34
The microprocessor’s R/W signal is used to distinguish between the two devices. While the microprocessor is reading, the data comes from the receiver; while writing, the data goes to the transmitter.

7.36
True. Therefore, the TC flag is normally used only for detecting that all the bits of the last character in a message or string of characters have been transmitted.

7.37
True. Using 9-bit characters is especially useful when transmitting 8-bit data numbers with parity.

7.40
Receiver. Framing errors are caused by improper signals that are generated outside the computer, usually due to electrical interference with the signals on the cable.

7.46
False. The A/D converter cannot cause an interrupt because its conversion time is much less than the time required to process an interrupt.

7.47
Four. In the multiple-channel reading mode, the A/D converter always reads four channels and this operation cannot be changed.

7.48
The SCI device cannot be powered down, but the A/D converter can be powered down. The A/D converter is powered down by a reset of the 68HC 11 chip, so the program must power up the A/D converter in order to use it.

7.50
Nothing has to be done because the port E pins work as both analog inputs and digital inputs. To read the digital inputs, the program must read the PORTE register at address 100A.

7.15
Next page
CLRA

CLRB

* TURN ON INTERRUPT SYSTEM

CLI

* ----------------------------------------------

* MAIN PROGRAM LOOP

* ELAPSED TIME OVER LIMIT?

HERE

LDX PLANE

CPD TLIMIT

BHI AHEAD BRANCH ON YES

* SET TIME OK INDICATOR

LDAA $BIT0

1 IN BIT0 = OK

STA TINC

BRA AGAIN

* SET TIME OVERLIMIT INDICATOR

AHEAD

LDAA $BIT1

1 IN BIT1 = OVERLIMIT

AGAIN

BRA HERE

*

**********

* INTERRUPT SERVICE ROUTINE

**********

* INTERRUPT POLLING CHAIN

**********

* INTERRUPT FROM IC3?

ICINT

LDA TINC REG

BNEIR FROM IC3

... INTERRUPT

* SERVICE INPUT CAPTURE 3

* CLEAR IC3 FLAG

LDAA $BIT0

STORE 1 TO CLEAR FLAG!

STA TINC REG, X

...ERRORS DO NOTHING

* BEGINNING OF PULSE?

BNEIR FROM IC3

...ERRORS DO NOTHING

* SAVE CURRENT CAPTURE TIME AT BEGINNING OF PULSE

LDX TIC3 REG, X

STD PULSEN

* SET UP FOR CAPTURE ON HIGH-TO-LOW

BNEIR FROM IC3

...ERRORS DO NOTHING

* CALCULATE PULSE LENGTH TIME

PULSEN

LDX TIC3 REG, X

STD PLENTH

* SET UP FOR CAPTURE ON LOW-TO-HIGH

BSET TIC3 REG, X

...ERRORS DO NOTHING

* RETURN FROM INTERRUPT

RTIC3

RTX

END