Jonathan W. Valvano March 2, 2005, 1:00pm-1:50pm

(5) Question 1. Value (resolution is 2^{-3}, which equals 1/8) if the integer is 25? 25/8=3.125

(4) Question 2. The debugger itself causes the system to operate abnormally. I) intrusive

(4) Question 3. A type of logic in which the output can be low or off. J)open collector

(4) Question 4. fixes all its inputs to specific values and can be repeated over and over. O) stabilize

(4) Question 5. A type of logic in which the output can be high, low, or off. K) tristate

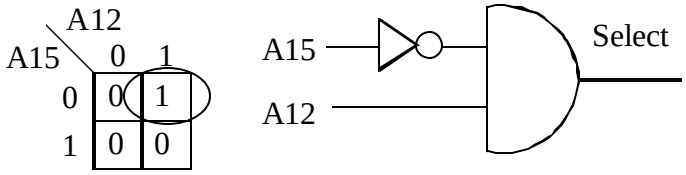
(4) Question 6. Latency is small and bounded. G) real-time

(5) Question 7. A signed fixed-point system has a range of values from -10.00 to 10.00 with a resolution of 10^{-2}. Note: 10^{-2} equals 0.01. With which of the following **data types** should the software variables be allocated? Precision is 2000, signed **F) `short`**

(20) Question 8. Design a minimal-cost positive-logic address decoder for **YourDevice** in the following system.

RAM	\$6000-\$6FFF	**`0110,XXXX,XXXX,XXXX`**	needs A12
YourDevice	\$7000-\$7FFF	**`0111,XXXX,XXXX,XXXX`**	
ROM	\$E000-\$FFFF	**`111X,XXXX,XXXX,XXXX`**	needs A15

Select = not(A15)*A12



(25) Question 9. There is a 32k by 8 bit PROM interfaced to a 6811 running at 2 MHz as shown below

RDA =(?/CE+120, ?/CE+20)
=(t_2+t_4+5+120, t_1+5+20)
=(250+5+120, 500+5+20)
=(375, 525)

RDR = (t_1 - t_4 - t_{17}, t_1 + t_{18A})
= (500 - 10 - 30, 500 + 10)
= (450, 510)

(25) Question 10. Write one function that receives a single ASCII character from the SCI with echo.

(10) Part a) Prototypes of public functions go in the **`SCI.h`** file

```
//************SCI_InOut************
// Receive from SCI with echo
// Inputs: none
// Outputs: returns ASCII code of new received data
unsigned char SCI_InOut(void);
```

(15) Part b) Implementations and any private variables (none needed here) go in the **`SCI.c`** file

```
//************SCI_InOut************
// Receive from SCI with echo
// Inputs: none
// Outputs: returns ASCII code of new received data
unsigned char SCI_InOut(void){
unsigned char data;
  while((SCISR1 & RDRF) == 0){};  // 1) wait receive
  data = SCIDRL;                  // 2) receive
  while((SCISR1 & TDRE) == 0){};  // 3) wait trans
  SCIDRL = data;                  // 4) transmit
  return data;                    // 5) return
}
```