Last Name: _____ First Name: _____

(3) Question 1. circle one	A B C
(3) Question 2. circle one	A B C
(3) Question 3. circle one	A B C
(3) Question 4. circle one	A B C
(3) Question 5. circle one	A B C
(10) Question 6. circle one	A B C D
(5) Question 7. give an integer	
(5) Question 8. give a value	
(5) Question 9. give two values	smallest= largest=
(5) Question 10. circle one	A B C D E F G H
(5) Question 11. circle one	A B C
(2) Question 12. give A-UU	
(2) Question 13. give A-UU	
(2) Question 14 give A-UU	
(2) Question 15. give A-UU	
(2) Question 16. give A-UU	
(2) Question 17. give A-UU	
(2) Question 18. give A-UU	
(2) Question 19. give A-UU	
(2) Question 20. give A-UU	
(2) Question 21. give A-UU	

(15) Question 22. Part a) Part b)	**(15) Question 23.**

Jonathan Valvano March 5, 2003, 1:00pm-1:50pm

This is a closed book exam. No notes or calculators are allowed, just a pencil and eraser. You must put your answers on the special answer pages only, do not turn in these exam questions. You have 50 minutes, so please allocate your time accordingly. ***Please read the entire quiz before starting.***

Consider the following simple C program.

```
const short aa=1000;
static short bb=1000;
short add3(const short cc){static short dd;
  dd = bb+cc;
  return(dd);}
void main(void){ short ee;
  ee = add3(aa);}
```

Where in memory are each of the variables allocated?

A) EEPROM
B) global RAM
or C) stack RAM

(3) Question 1. `aa`
(3) Question 2. `bb`
(3) Question 3. `cc`
(3) Question 4. `dd`
(3) Question 5. `ee`

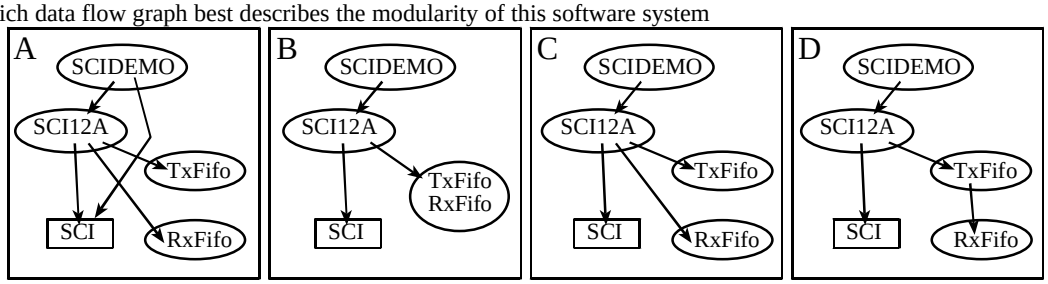
(10) Question 6. Consider the following program stubs.

```
// ******* SCIDEMO.C **************
#include "SCI12.H"
void main(void){
// stuff
#include "SCI12A.C"
// ++++++end of SCIDEMO.C +++++++++
// ******* SCI12A.C **************
#include "RxFifo.H"
#include "TxFifo.H"
// stuff
#include "RxFifo.C"
#include "TxFifo.C"
// ++++++end of SCI12A.C +++++++++
```

```
// ******* RxFifo.C **************
//
// stuff
//
// ++++++end of RxFifo.C +++++++++

// ******* TxFifo.C **************
//
// stuff
//
// ++++++end of TxFifo.C +++++++++
```

Which data flow graph best describes the modularity of this software system



(5) Question 7. What integer is stored in the computer when the value 1.5 is stored in 8-bit unsigned binary fixed-point, with a resolution of 2^{-5}? Note: 2^{-5} equals 1/32.

(5) Question 8. What is the value of a 16-bit signed decimal fixed-point number (resolution is 10^{-5}, which equals 0.00001) if the integer stored in memory is –10542?

(5) Question 9. What is the range (smallest value to largest value) for an 8-bit signed binary fixed-point number system with a resolution of 2^{-6}? Note: 2^{-6} equals 1/64.

(5) Question 10. An unsigned fixed point system has a range of 0 to 10 with a resolution of 0.1. With which of the following data types should the software variables be allocated?

A) `unsigned char`	D) `char`	G) `float`
B) `unsigned short`	E) `short`	H) `double`
C) `unsigned long`	F) `long`	

(5) Question 11. What is the purpose of the following code, which occurs in the file `rti.c`?

```
#pragma abs_address:0xfff0
void (*RTI_vector[])() = { RTIHan };
#pragma end_abs_address
```

A) specifies the location to jump to when a RTI interrupt is serviced
B) specifies that `RTIHan` is an interrupt handler and should return with an `rti` instruction
or C) specifies the start location of the program

For questions 12-21, the definition is given and you are asked to give the correct term described by that definition. Since there are more terms than definitions, not all terms will be used. Answer each as A through UU.

(2) Question 12. An instrument used to produce a side effect without halting execution.
(2) Question 13. Modify a software system fixing all its inputs.
(2) Question 14. Measure the timing characteristics or execution pattern of a program.
(2) Question 15. A software-hardware synchronization method where the software continuously reads the hardware status waiting for the hardware operation to complete.
(2) Question 16. A system that can guarantee an upper bound on latency.
(2) Question 17. The act of clearing the flag that requested the interrupt.
(2) Question 18. The place in the computer where division is performed.
(2) Question 19. A condition where information is lost when power is removed.
(2) Question 20. Output current when the signal is low.
(2) Question 21. The characteristic of a debugger that allows the software/hardware system to operate normally as if the debugger did not exist.

A) address bus
B) ALU
C) atomic
D) BIU
E) break
F) busy waiting
G) control bus
H) CPU
I) critical section
J) CU
K) data bus
L) desk check
M) embedded computer
N) friendly
O) functional debugging
P) I_{IH}
Q) I_{IL}
R) instrument
S) interrupt acknowledge
T) interrupt arm
U) interrupt enable
V) interrupt vector
W) invasiveness
X) I_{OH}
Y) I_{OL}
Z) latency
AA) non-intrusiveness
BB) nonvolatile
CC) open collector
DD) performance debugging
EE) processor
FF) profile
GG) RAM
HH) real-time
II) reentrant
JJ) registers
KK) ROM
LL) scan
MM) stabilize
NN) thread
OO) tristate
PP) V_{IH}
QQ) V_{IL}
RR) V_{OH}
SS) V_{OL}
TT) volatile
UU) write cycle

(15) Question 22. A scope is attached to PT6, and will be used to visualize software activity. The answers to this problem do not need to be complete functions, just the C code fragments.
Part a) Write friendly C code that makes PT6 an output.
Part b) Write friendly C code that toggles PT6 (changes 0 to 1 or changes 1 to 0).

(15) Question 23. Assume PH5 is an output and the other PORTH bits are input. Write friendly C code (just a fragment) that sets PH5 high if PH1 is low and PH2 is high. This code does not clear PH5 low if false.