Assignment 1

Due Friday, October 20 2001

Question 1

Convert the following decimal numbers to binary and hexadecimal. Express the hexadecimal numbers as C constants (use a `'0x'` prefix).

1. `8`
2. `7`
3. `16`
4. `15`
5. `256`
6. `255`
7. `237`

Question 2

Convert the following binary numbers to hexadecimal and decimal.

1. `1011`
2. `1011 1011`
3. `1000 0000`
4. `11 1100`
5. `0011 1100`

Question 3

Convert the following hexadecimal numbers to binary and decimal.

1. `0x0e`
2. `0xe`
3. `0xAA`
4. `0xFA`
5. `0x40`
6. `0x18`

Question 4

What are the values of the following expressions? Give your answer in hex (hexadecimal) notation.

1. `( 0xaa & 0x0f )`
2. `( 0x3c & 0xf0 ) | ( 0x3c & 0x0f )`
3. `3 * ( 0xf0 && 0x0f )`
4. `( 0x3c ^ 0xff ) + ( 1 < 3 )`
5. `~ ( 128 | ' ' )`
6. `128 || ( ' ' == 0x20 )`

Question 5

Write a function called `printbin` that prints the binary value of an integer whose value can be assumed to be less than 32768. This function should take one integer argument and not return a value. *Hint: use a loop which computes the values of powers of 2 starting at 16384 and going down to 1. Use the conversion algorithm described in the class notes.*

Question 6

Write a function with the name `len` that takes an array of characters as an argument and returns an integer value which is the number of characters in the array. The last character in the array has a value zero. This character should not be included in the count. For example, given the following sequence of statements, the value of the last expression would be 2:

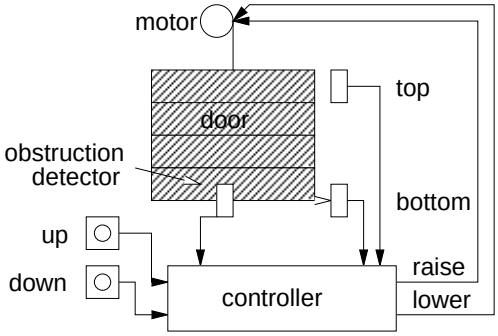
```
char s[3] ;

s[0]='H' ;
s[1]='i' ;
s[2]=0 ;

len(s) ;
```

Question 7

Consider a controller for a simple garage door opener as shown in the following diagram:



The controller has five inputs: two pushbuttons to request that the door be opened (up) or closed (down), two switches that indicate when the door is at the top or bottom of its range and a switch that indicates that the door has hit an obstruction. The controller has two outputs that control a motor: one to raise and one to lower the door.

The operation of the controller is fairly simple: if the up or down switch is pressed (even momentarily) the door should be moved up or down until it is fully open or closed. If an obstruction is sensed at any time the door should be moved up until it is fully open. It should be possible to reverse the direction of the door while it is moving by pressing the appropriate button.

Design a state machine controller for the garage door opener. List the inputs and outputs. List a sufficient number of states and a name for each state. Give a table giving the output conditions for each state. Draw a state transition diagram showing the states and the logical conditions that cause transitions between them. Write out a tabular description of the state machine with the following columns: starting state, input, next state.

The state transition table may use 'X' to indicate a "don't-care" input (i.e. that particular state transition happens for both values of that particular input). However, your table must be unambiguous: the same combination of state and input conditions cannot appear on two different lines.

Derive the logic equations for all of the logic functions required to implement the controller. Draw a schematic diagram of the controller using AND/OR/NOT gates and D flip–flops.

You may assume a 1 kHz clock signal is available (show where it would be connected in the circuit).

What minimum duration of button push is required to ensure the controller will act on it?

Question 8

Design a controller for a coin-operated candy vending machine. The machine has a coin detector that can detect nickels, dimes and quarters and an electrically-driven candy release mechanism. A candy costs 15 cents. The machine does not give change or have a coin return.

The coin detector has two output lines encoded as follows:

00	no coin detected
01	nickel
10	dime
11	quarter

The candy release mechanism will deliver one candy for each time its control line goes from zero to one.

Identify the inputs and outputs for this controller. Identify a set of states and outputs for the controller and draw the state transition diagram. Write out a tabular description of the state machine (you may use symbolic rather than binary encodings for the states).