EE 472 – Embedded Systems

Dr. Shwetak Patel
Assistant Professor
Computer Science & Engineering
Electrical Engineering

Agenda

- Lab1
- Finish C programming
  - Bitwise operations
- Lab time

Questions?

- Pointers
- Function pointers
- Arrays
- Structs

Bitwise Operations

- & bitwise AND
- | bitwise OR
- ^ bitwise XOR
- << left shift
- >> right shift
- ~ ones complement
Bitwise Operations

- Important in embedded systems
- Manipulating registers
  - Setting and observing
- Sometimes faster for certain arithmetic operations

Bitwise Operations

- & bitwise AND
- | bitwise OR
- ^ bitwise XOR
- << left shift
- >> right shift
- ~ ones complement

Class Exercise

```c
// Use of bitwise logical operators */
#define Bit_Zero 0x01
#define Bit_One 0x02
#define Bit_Two 0x04
#define Bit_Three 0x08
#define Bit_Four 0x10
#define Bit_Five 0x20
#define Bit_Six 0x40

// etc etc etc...

int x = 0x01;  // x = [...00001001]
int y = 0x01;  // y = [...00001001]
int z = x < 2;  // z = [...00101000]
int z1 = y ^ Bit_Three;  // z1 = [...00001000]
int z2 = z | Bit_Four;  // z2 = [...00111100]
```

XOR

- Can hide two pieces of data in one variable
  - `variable2 = variable1 ^ Feature`
  - `variable3 = variable2 ^ Feature`
  - (variable3 and variable1 are the same now)
New swap function

- Use XOR to redo swap without a temp variable

XOR Swap

```c
void swap(int *a, int *b){
    *a = *a ^ *b;  //hide *b in *a
    *b = *a ^ *b;  //pull the original *a out
    *a = *a ^ *b;  //pull the original *b out
}
```

Class Exercise

- Write a C function using bitwise logical operators to print the binary value of a 16 bit unsigned integer

Example solution

```c
void bin_display(unsigned int x){
    unsigned int i,j;
    for(i=0;i<16;i++){
        j = 1<i<15;i++>
        if(j & x){ printf("1"); }
        else { printf("0"); }
    }
}
```
Questions?

- Next time:
  - Software Design
  - UML

Class Exercise

```java
// Convert 0.0625 to decimal: ___0.0625___

// Convert 0.0625 to decimal: ____0.0625_____

// Convert 0.0625 to float: ______0.0625_____

// Convert 0.0625 to float: ______0.0625_____

/* How many digits of the least significant 9 bits of the following values: */
set a = 11001001; // a = [1 1 0 0 1 0 0 1]
set b = 11001100; // b = [1 1 0 0 1 1 0 0]
set c = 0.0625; // c = [0.0000000000000001]
set d = (int) c; // d = [0]

set e = 1.0; // e = [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
```

Class Exercise