

CSE460

Switching Theory

Lecture7

John Lockwood

Washington University

Spring 2006

<http://www.arl.wustl.edu/~lockwood/class/coe460/>

Programming Assignment

- You are free to use any language or programming environment to implement homework.
- You must implement your own code for the assignments and submit your original source code.

Programming

- SOP Input Format:

- n : Number of variables (1-32)
- ###..## = n-size Array of {'0','1','-'}
 - 0 : Zero
 - 1 : One
 - - : Unspecified

- Minterm Output Format

- Individually Listed Terms
- Let K=Number of unspecified terms
- 2^k elements of n-size Array of {'0','1'}

- Example

- Input
8
1110-0--
- Output
11100000
11100001
11100010
11100011
11101000
11101001
11101010
11101011

Helpful Data Structure

```
class sop_term {  
public:  
    unsigned int value; // up to 32-bits: 0=false, 1=true  
    unsigned int mask; // up to 32-bits: 0=Care, 1=DontCare  
};
```

- Example

```
N=5  
Input = 10-0-  
  
Value = 10101  
Mask = 00101
```

Bit-wise Operators

- The following C/C++ operators may be helpful:
 - Logical left shift
 - `Value << bits`
 - Logical right shift
 - `Value >> bits`
 - Bit-wise AND
 - `X & Y` [Note that `&&` is for logical AND]
 - Bit-wise OR
 - `X | Y` [Note that `||` is for local OR]

More Bit-wise operators in C/C++

- C/C++ and 80x86 Assembly Equivalent Operations
 - `X &= Y` → AND X,Y
 - `X |= Y` → OR X,Y
 - `X ^= X` → NOT X
 - `Y = 1 << X` → SHL 1,X
- Scanning ASCII Arrays
 - If `(linebuffer[0]=='0')` ...
- Scanning Bit Array
 - If `((x & 0x00000001) == 0)` ...
- Bit Masking Operation Example
 - Turn on a bit in the kth position
 - In-class example

Methods (Example)

```
#define MaxLineSize 200

int n; // Number of variables in function
char linebuffer[MaxLineSize]; // Buffer to hold input

class sop_term {
public:
    unsigned int value;
        // up to 32-bits: 0=false, 1=true
    unsigned int mask;
        // up to 32-bits: 0=Care, 1=DontCare
public:
    int  sop_term::read_value();
    void sop_term::display_var(int i);
    void sop_term::display();
    void sop_term::display_minterms();
};
```

Methods to Read in Data Values

```
int sop_term::read_value(){
    int i;
    int read_error=0;

    value=0;
    mask =0;

    scanf("%s\n",&linebuffer);

    // Read ASCII input, covert to value, mask
    // for bits 0..n-1
    ...
}
```