

# **Chimera-2018-A Emulator Assignment**

## **Practical 5 - Arithmetic**

CANS Tech INC

**It is the easy stuff now...**

**You have already done some arithmetic**

## Implementing the SBB Instruction

Once again inside the Group\_1 function switch add

```
case 0x20: // SBB  
    CODE HERE  
    break;
```

SBB		Addressing	Opcode
Register subtracted to Accumulator with Carry		A-B	0x20
		A-C	0x21
		A-D	0x22
Flags:	T T - - - T T	A-E	0x23
notes		A-F	0x24

**SBB is ADD with a '-' instead of a '+'**

## Implementing the OR Instruction

Once again inside the Group\_1 function switch add

```
case 0x40: // OR  
    CODE HERE  
    break;
```



OR		Addressing	Opcode
Register bitwise inclusive or with Accumulator		A-B	0x40
		A-C	0x41
		A-D	0x42
Flags:	T - - - - T -	A-E	0x43
notes		A-F	0x44

**OR is AND with a '|' instead of a '&'**

**Remember there is 15 marks for quility of code!**

**Think about the similarities between some instructions**

## **Implementing the NOTA Instruction**

Once again inside the Group\_1 function switch add

```
case 0xA8: // NOTA  
    CODE HERE  
    break;
```

NOTA		Addressing	Opcode
Negate Memory or Accumulator		A	0xA8
Flags:	T - - - - T T		
notes			

COMA simply takes the 1's complement of Register A

1's complement inverts each of the bits

One way is to invert the bits in Register A is to XOR it with 0xFF

Another way would be to use the c operator

Now you can implement

EOR, TST, DEC, ASL, ASR, LSR, NEG, ADCP, SBCP, XCHG, SWI,  
RTI,



**Questions?**