Microcontroller and Microprocessors
Experiment - 3
Data Transfer And Calculator

Varun Singh Inda
16BEE0023

Aim :-To transfer date from i) RAM to EXT ii) RAM to RAM iii) EXT to EXT iv) ROM to EXT

Procedure :-

1. Launch Keil uVision
2. Create a new project and load the NXP microcontroller P895V1RD2
3. Create a blank document and save the file with extension .asm
4. Load the saved .asm file under the target folder for the ongoing project.
5. Type the code and save the file once again
6. Build the target to check if there are any errors or warnings
7. If there are any errors, debug them and continue with step 8
8. Debug the program and note the values of accumulator and corresponding registers
9. Run the program and verify the updated values of the accumulator and corresponding registers

Logic :-

For RAM to EXT , EXT to EXT, RAM to RAM:
- Use MOV command to locate the data to be entered in Data Pointer and R1 and set value for R5 to act as a timer
- Use MOV command to copy data at location R0 to A (In case of EXT memory, use MOVX command to copy data from EXT location to A)
- Use MOVX command to copy data from A to EXT location (In case of RAM use MOV command to copy data from A to RAM location)
- Use INC command to change the location of R0 and R1 to the next location
- Use DJNZ command to Jump to loop L until R5 becomes zero and decrement value of R5 by 1 as it jumps

For ROM to EXT :
- Use MOV command to locate the data to be entered in R0 and R1 and set value for R5 to act as a timer
- Use MOVC command to copy data from data pointer to A
- Use MOVX command to copy data from A to EXT location
- Use INC command to change the location of Data Pointer and R1 to the next location
- Use DJNZ command to Jump to loop L until R5 becomes zero and decrement value of R5 by 1 as it jumps.

Code :-

i) RAM to EXT :

```
MOV R5, #05H
MOV R0, #20H
MOV R1, #30H
L : MOV A, @R0
MOVX @R1, A
INC R0
INC R1
DJNZ R5, L
END
```

ii) RAM to RAM :

```
MOV R5, #05H
MOV R0, #20H
MOV R1, #30H L :
MOV A, @R0
MOV @R1, A
INC R0
INC R1
DJNZ R5, L
END
```

iii) EXT to EXT :

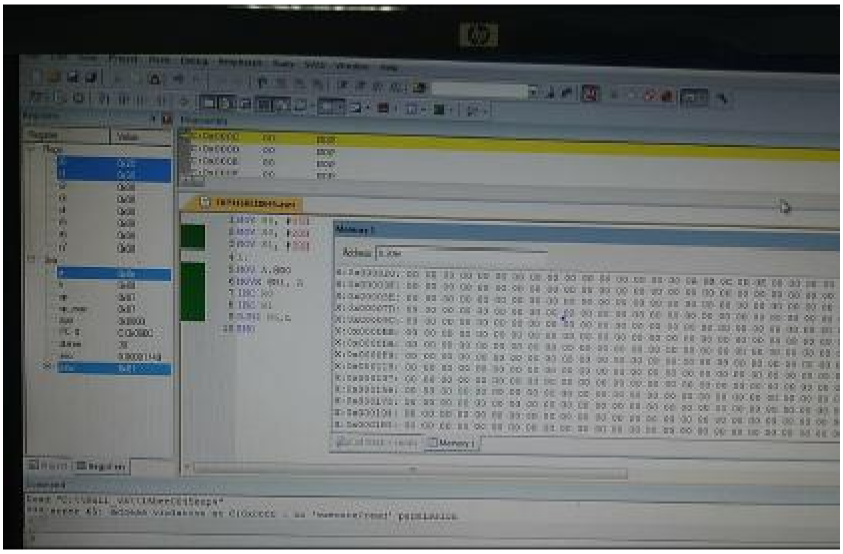
```
MOV R5, #05H
MOV R0, #20H
MOV R1, #30H
L : MOVX A, @R0
MOVX @R1, A
INC R0
INC R1
DJNZ R5, L
END
```

iv) ROM to EXT :

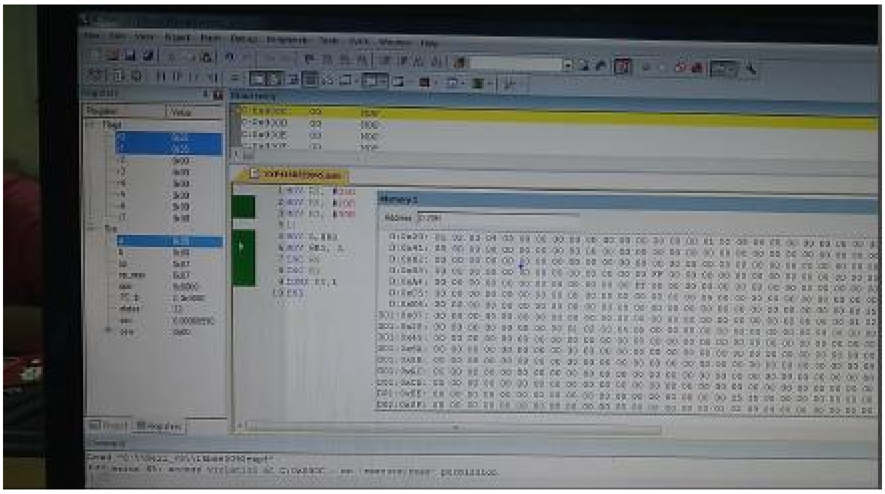
```
MOV R5, #05H
MOV DPTR, #20H
MOV R1, #30H
L : CLR A
MOVC A, @A+DPTR
MOVX @R1, A
INC DPTR
INC R1
DJNZ R5, L
END
```

Pictures

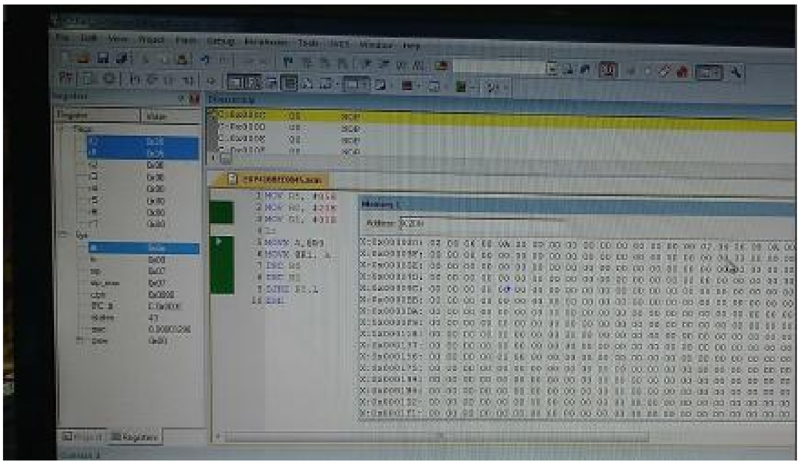
RAM to EXT



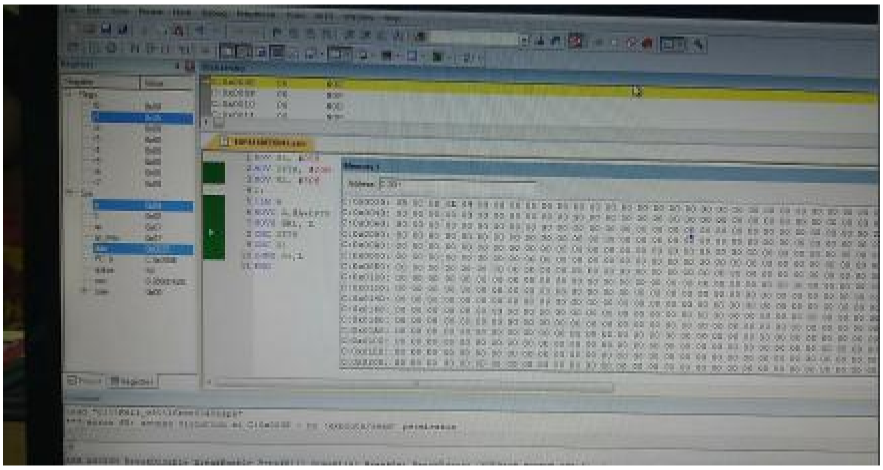
RAM to RAM



EXT to EXT



ROM to EXT



<u>Conclusion :- </u>The experiment was successfully performed with data being transferred from I) RAM to RAM II) RAM TO ROM III)RQM TO EXT IV)EXT TO EXT

<u>Aim:-</u> To make the 8051-processor work like a calculator

<u>Procedure:-</u>

1. Launch Keil uVision
2. Create a new project and load the NXP microcontroller P895V1RD2
3. Create a blank document and save the file with extension .asm
4. Load the saved .asm file under the target folder for the ongoing project.
5. Type the code and save the file once again
6. Build the target to check if there are any errors or warnings
7. If there are any errors, debug them and continue with step 8
8. Debug the program and note the values of accumulator and corresponding registers
9. Run the program and verify the updated values of the accumulator and corresponding registers

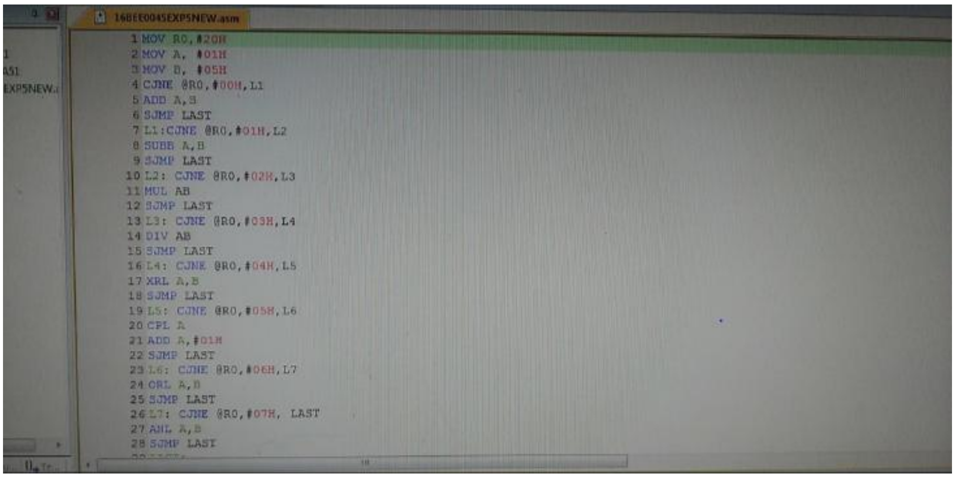
<u>Logic:-</u>

- Use MOV command to direct R0 to location 20h and to copy values 1 to R2 and 5 to R3.
- Use CJNE command to compare jump to next operation if not equal to a certain value for register R0 to check the input and perform the operation accordingly.
- Use SJMP to jump directly to the end of the program without comparing further.

<u>Code:-</u>

```
MOV R0, 20H
MOV R2, #01H
MOV R3, #05H
CJNE @R0, #00H, L
MOV A, R2
MOV B, R3
ADD A,B
SJMP LAST
L :
CJNE @R0, #01H, L1
MOV A, R2
MOV B, R3
SUBB A, B
SJMP LAST
L1 :
CJNE @R0, #02H, L2
MOV A, R2
MOV B, R3
XRL A, B
SJMP
LAST L2:
CJNE @R0, #03H, L3
MOV A, R2
CPL A
ADD A, #01H
SJMP
LAST L3:
CJNE @R0, #04H, L4
MOV A, R2
MOV B, R3
DIV AB
SJMP
LAST L4:
CJNE @R0, #05H, L5
MOV A, R2
MOV B, R3
ORL A, B
SJMP LAST L5:
CJNE @R0, #06H, L6
MOV A, R2
MOV B, R3
MUL AB SJMP
LAST L6:
CJNE @R0, #07H, LAST
MOV A, R2
MOV B, R3
ANL A, B
SJMP
LAST
LAST: END
```

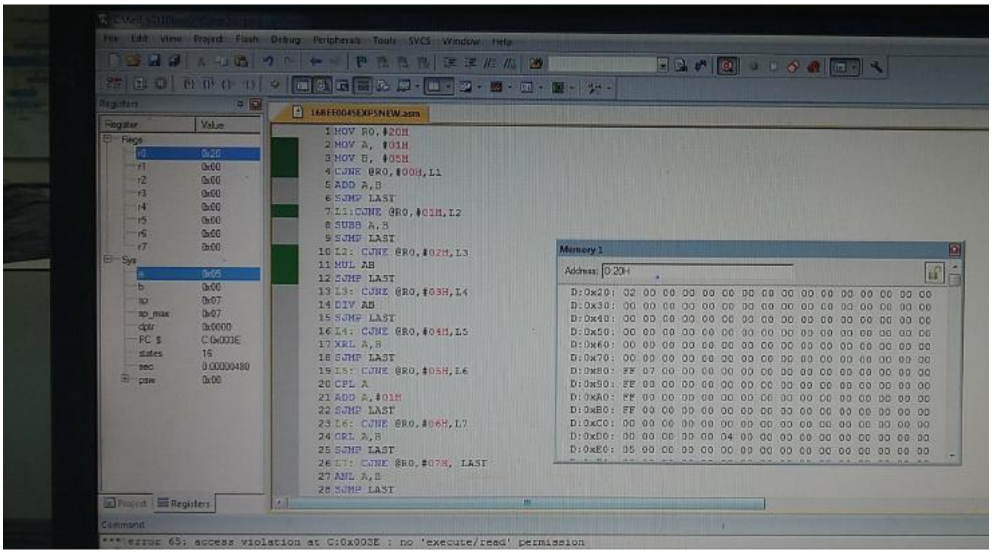
Pictures



16BEE0045EXP5NEW.asm

```
MOV R0,#20H
MOV A, #01H
MOV B, #05H
CJNE @R0,#00H,L1
ADD A,B
SJMP LAST
L1:CJNE @R0,#01H,L2
SUBB A,B
SJMP LAST
L2: CJNE @R0,#02H,L3
MUL AB
SJMP LAST
L3: CJNE @R0,#03H,L4
DIV AB
SJMP LAST
L4: CJNE @R0,#04H,L5
XRL A,B
SJMP LAST
L5: CJNE @R0,#05H,L6
CPL A
ADD A,#01H
SJMP LAST
L6: CJNE @R0,#06H,L7
ORL A,B
SJMP LAST
L7: CJNE @R0,#07H, LAST
ANL A,B
SJMP LAST
```

Multiply



File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
Project
Registers
Command

Register	Value
Regs	
r0	0x20
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
Sys	
a	0x05
b	0x00
sp	0x07
sp_max	0x07
dptr	0x0000
PC $	C:0x003E
states	16
sec	0.00000480
psw	0x00

16BEE0045EXP5NEW.asm

```
MOV R0,#20H
MOV A, #01H
MOV B, #05H
CJNE @R0,#00H,L1
ADD A,B
SJMP LAST
L1:CJNE @R0,#01H,L2
SUBB A,B
SJMP LAST
L2: CJNE @R0,#02H,L3
MUL AB
SJMP LAST
L3: CJNE @R0,#03H,L4
DIV AB
SJMP LAST
L4: CJNE @R0,#04H,L5
XRL A,B
SJMP LAST
L5: CJNE @R0,#05H,L6
CPL A
ADD A,#01H
SJMP LAST
L6: CJNE @R0,#06H,L7
ORL A,B
SJMP LAST
L7: CJNE @R0,#07H, LAST
ANL A,B
SJMP LAST
```

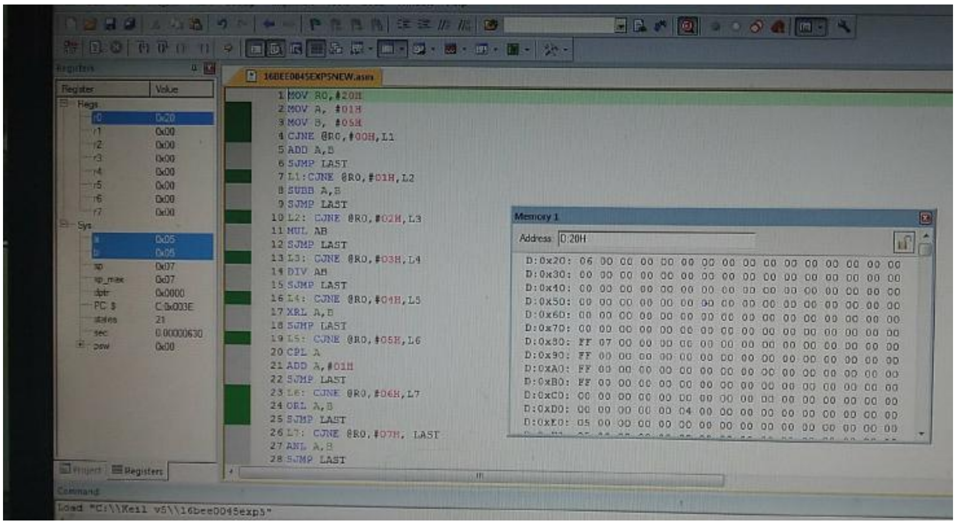
Memory 1

Address: D:20H

```
D:0x20: 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x80: FF 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x90: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xA0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xB0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xC0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xD0: 00 00 00 00 00 04 00 00 00 00 00 00 00 00 00 00
D:0xE0: 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
*** error 65: access violation at C:0x003E : no 'execute/read' permission
```

OR Gate



Register	Value
Regs	
r0	0x20
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
Sys	
a	0x05
b	0x05
sp	0x07
sp_max	0x07
dptr	0x0000
PC $	C:0x003E
states	21
sec	0.00000630
psw	0x00

16BEE0045EXP5NEW.asm

```
MOV R0,#20H
MOV A, #01H
MOV B, #05H
CJNE @R0,#00H,L1
ADD A,B
SJMP LAST
L1:CJNE @R0,#01H,L2
SUBB A,B
SJMP LAST
L2: CJNE @R0,#02H,L3
MUL AB
SJMP LAST
L3: CJNE @R0,#03H,L4
DIV AB
SJMP LAST
L4: CJNE @R0,#04H,L5
XRL A,B
SJMP LAST
L5: CJNE @R0,#05H,L6
CPL A
ADD A,#01H
SJMP LAST
L6: CJNE @R0,#06H,L7
ORL A,B
SJMP LAST
L7: CJNE @R0,#07H, LAST
ANL A,B
SJMP LAST
```

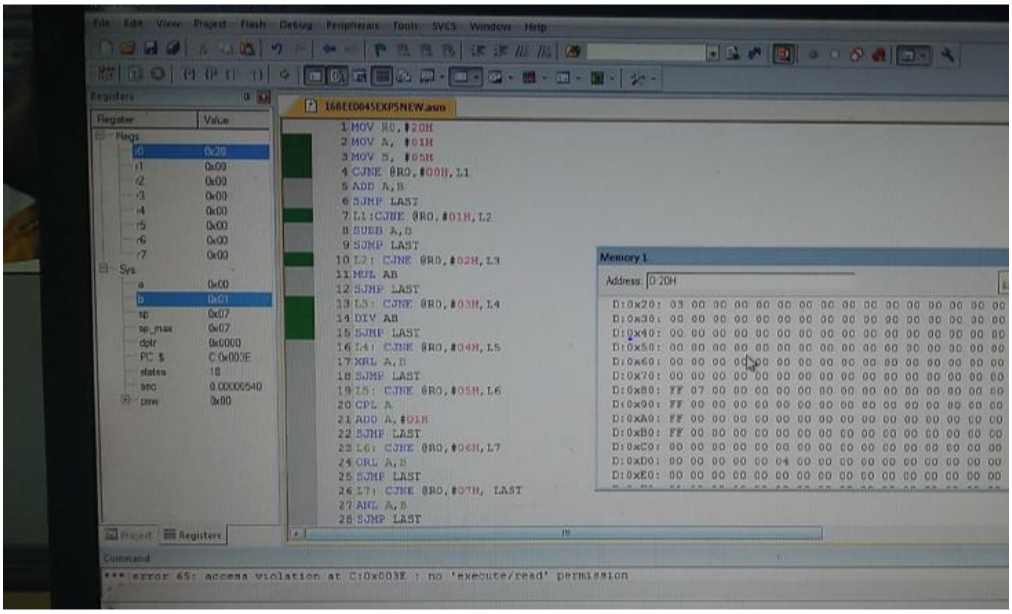
Memory 1

Address: D:20H

```
D:0x20: 06 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x80: FF 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x90: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xA0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xB0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xC0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xD0: 00 00 00 00 00 04 00 00 00 00 00 00 00 00 00 00
D:0xE0: 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
Load "C:\\Keil v5\\16bee0045exp5"
```

Division



File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
16BEE0045EXP5NEW.asm
Project
Registers
Command

Register	Value
Regs	
r0	0x20
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
Sys	
a	0x00
b	0x01
sp	0x07
sp_max	0x07
dptr	0x0000
PC $	C:0x003E
states	18
sec	0.00000540
psw	0x00

```
MOV R0,#20H
MOV A, #01H
MOV B, #05H
CJNE @R0,#00H,L1
ADD A,B
SJMP LAST
L1:CJNE @R0,#01H,L2
SUBB A,B
SJMP LAST
L2: CJNE @R0,#02H,L3
MUL AB
SJMP LAST
L3: CJNE @R0,#03H,L4
DIV AB
SJMP LAST
L4: CJNE @R0,#04H,L5
XRL A,B
SJMP LAST
L5: CJNE @R0,#05H,L6
CPL A
ADD A,#01H
SJMP LAST
L6: CJNE @R0,#06H,L7
ORL A,B
SJMP LAST
L7: CJNE @R0,#07H, LAST
ANL A,B
SJMP LAST
```

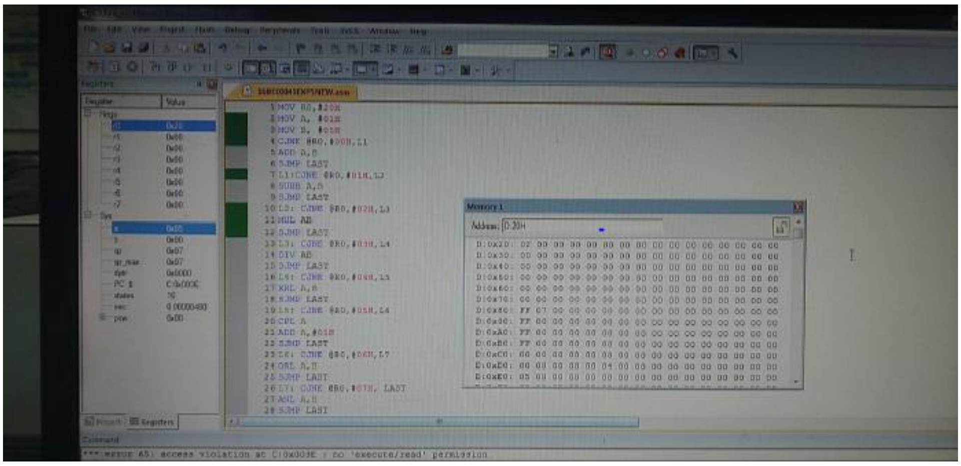
Memory 1

Address: D:20H

```
D:0x20: 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x80: FF 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x90: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xA0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xB0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xC0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xD0: 00 00 00 00 00 04 00 00 00 00 00 00 00 00 00 00
D:0xE0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
*** error 65: access violation at C:0x003E : no 'execute/read' permission
```

Multiplication



File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
16BEE0045EXP5NEW.asm
Project
Registers
Command

Register	Value
Regs	
r0	0x20
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
Sys	
a	0x05
b	0x00
sp	0x07
sp_max	0x07
dptr	0x0000
PC $	C:0x003E
states	16
sec	0.00000480
psw	0x00

```
MOV R0,#20H
MOV A, #01H
MOV B, #05H
CJNE @R0,#00H,L1
ADD A,B
SJMP LAST
L1:CJNE @R0,#01H,L2
SUBB A,B
SJMP LAST
L2: CJNE @R0,#02H,L3
MUL AB
SJMP LAST
L3: CJNE @R0,#03H,L4
DIV AB
SJMP LAST
L4: CJNE @R0,#04H,L5
XRL A,B
SJMP LAST
L5: CJNE @R0,#05H,L6
CPL A
ADD A,#01H
SJMP LAST
L6: CJNE @R0,#06H,L7
ORL A,B
SJMP LAST
L7: CJNE @R0,#07H, LAST
ANL A,B
SJMP LAST
```

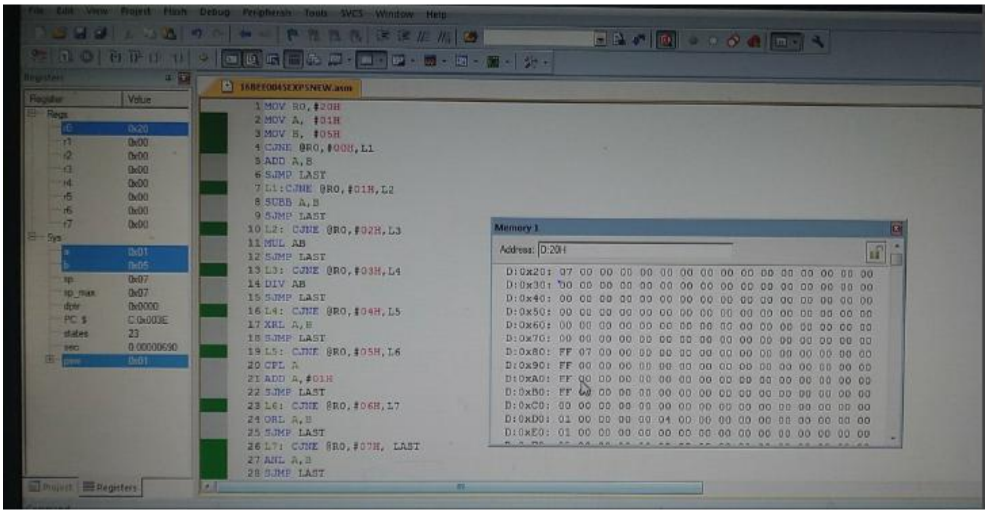
Memory 1

Address: D:20H

```
D:0x20: 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x80: FF 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x90: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xA0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xB0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xC0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xD0: 00 00 00 00 00 04 00 00 00 00 00 00 00 00 00 00
D:0xE0: 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
*** error 65: access violation at C:0x003E : no 'execute/read' permission
```

AND Gate



File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
16BEE0045EXP5NEW.asm
Project
Registers

Register	Value
Regs	
r0	0x20
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
Sys	
a	0x01
b	0x05
sp	0x07
sp_max	0x07
dptr	0x0000
PC $	C:0x003E
states	23
sec	0.00000690
psw	0x01

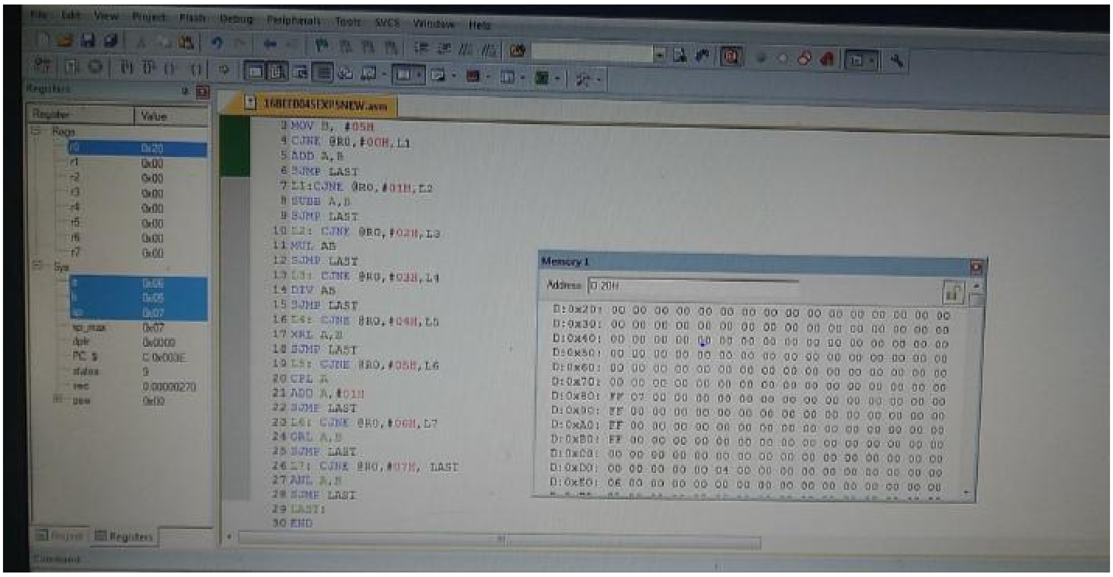
```
MOV R0,#20H
MOV A, #01H
MOV B, #05H
CJNE @R0,#00H,L1
ADD A,B
SJMP LAST
L1:CJNE @R0,#01H,L2
SUBB A,B
SJMP LAST
L2: CJNE @R0,#02H,L3
MUL AB
SJMP LAST
L3: CJNE @R0,#03H,L4
DIV AB
SJMP LAST
L4: CJNE @R0,#04H,L5
XRL A,B
SJMP LAST
L5: CJNE @R0,#05H,L6
CPL A
ADD A,#01H
SJMP LAST
L6: CJNE @R0,#06H,L7
ORL A,B
SJMP LAST
L7: CJNE @R0,#07H, LAST
ANL A,B
SJMP LAST
```

Memory 1

Address: D:20H

```
D:0x20: 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x80: FF 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x90: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xA0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xB0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xC0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xD0: 01 00 00 00 00 04 00 00 00 00 00 00 00 00 00 00
D:0xE0: 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Addition



File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
16BEE0045EXP5NEW.asm
Memory 1
Registers

```
MOV B, #05H
CJNE @R0,#00H,L1
ADD A,B
SJMP LAST
L1:CJNE @R0,#01H,L2
SUBB A,B
SJMP LAST
L2: CJNE @R0,#02H,L3
MUL AB
SJMP LAST
L3: CJNE @R0,#03H,L4
DIV AB
SJMP LAST
L4: CJNE @R0,#04H,L5
XRL A,B
SJMP LAST
L5: CJNE @R0,#05H,L6
CPL A
ADD A,#01H
SJMP LAST
L6: CJNE @R0,#06H,L7
ORL A,B
SJMP LAST
L7: CJNE @R0,#07H, LAST
ANL A,B
SJMP LAST
LAST:
END
```

Register	Value
Regs	
r0	0x20
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
Sys	
a	0x06
b	0x05
sp	0x07
sp_max	0x07
dptr	0x0000
PC $	C:0x003E
states	9
sec	0.00000270
psw	0x00

Address: D:20H

```
D:0x20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x80: FF 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x90: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xA0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xB0: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xC0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0xD0: 00 00 00 00 00 04 00 00 00 00 00 00 00 00 00 00
D:0xE0: 06 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Conclusion:- A calculator was successfully coded using 8051 microcontroller.