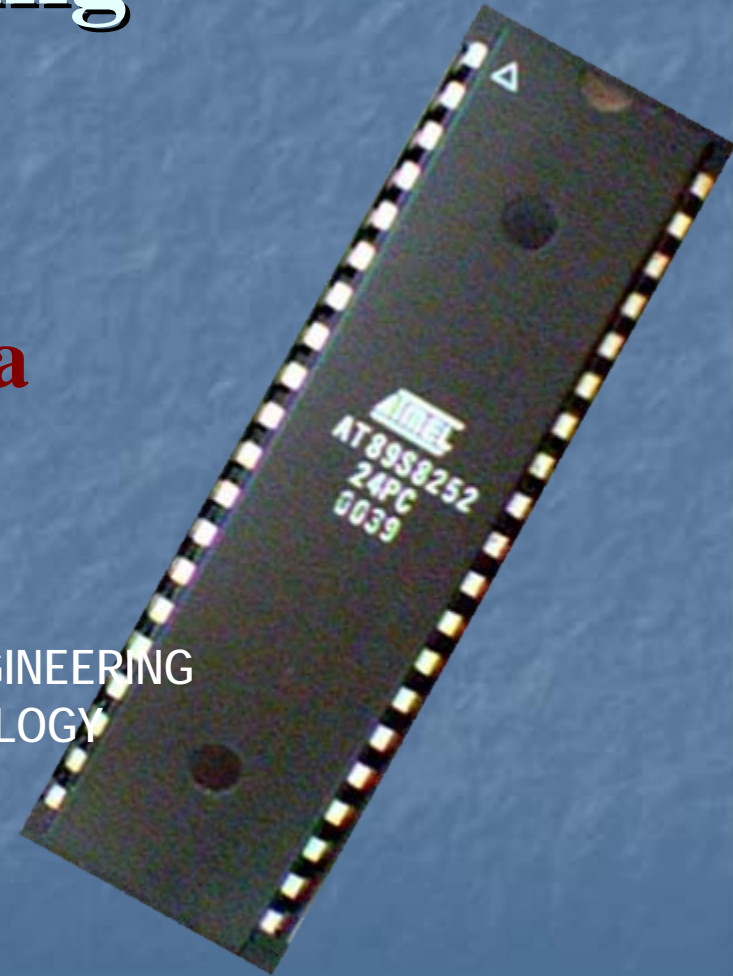


The 8051 Architecture and Programming

Supratim Gupta

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY
KHARAGPUR



Programming Issues

❖ Hardware Information

- Architecture of the processor

❖ Software Instructions

- Complex Instruction Set Computer (CISC)
- 135 Instructions for 8051

8051 Architecture

Specific Features

1. 8-bit CPU with registers A (accumulator) and B
2. 16-bit program counter (PC) and data pointer (DPTR)
3. 8-bit program status word (PSW)
4. 8-bit stack pointer (SP)
5. Internal ROM or EPROM of 4k (8051)
6. Internal RAM of 128 bytes
 - 4 register banks each containing 8 registers of 8-bit
 - 16-bytes bit addressable registers
 - 80 bytes general purpose data memory

8051 Architecture

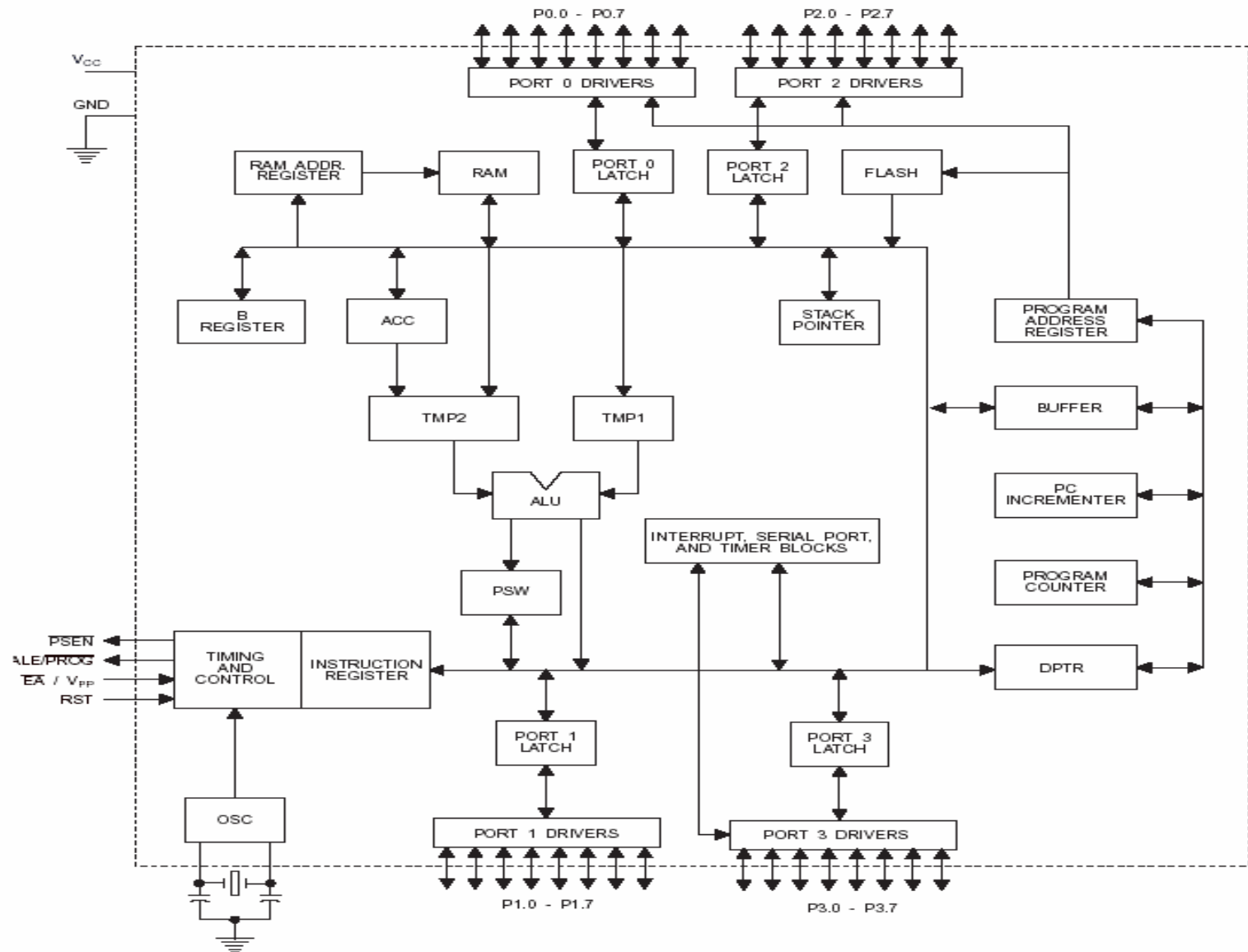
Specific Features

7. 32 input-output pins arranged as 4 eight-bit ports (P0 – P3)
8. Two 16 bit timer/counters : T0 and T1
9. Full duplex serial data receiver/transmitter
10. Control registers : TCON, TMOD, SCON, PCON, IP, and IE
11. 2 external and 3 internal interrupt sources
12. Oscillator and clock circuit

8051 Architecture

Cntd...

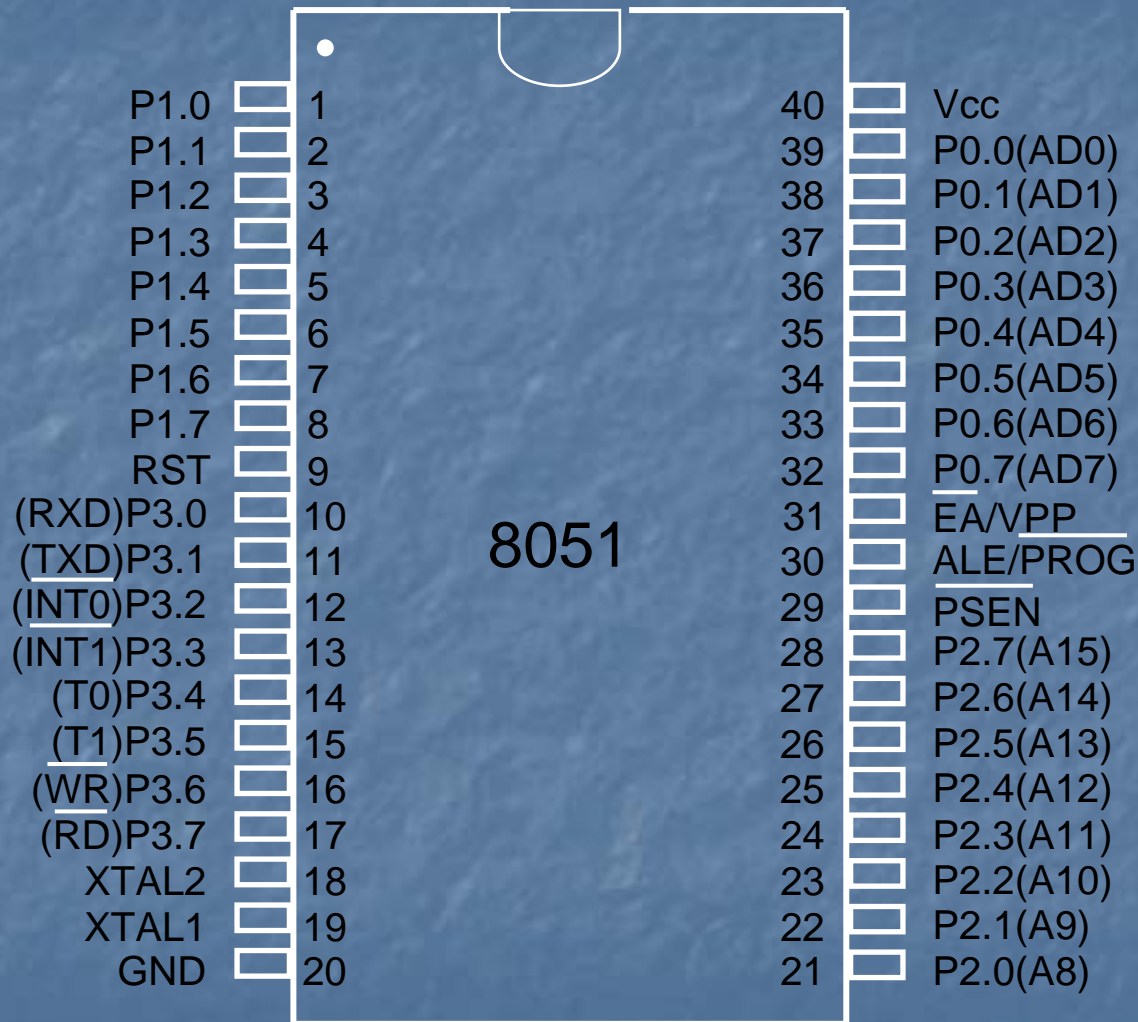
Block Diagram



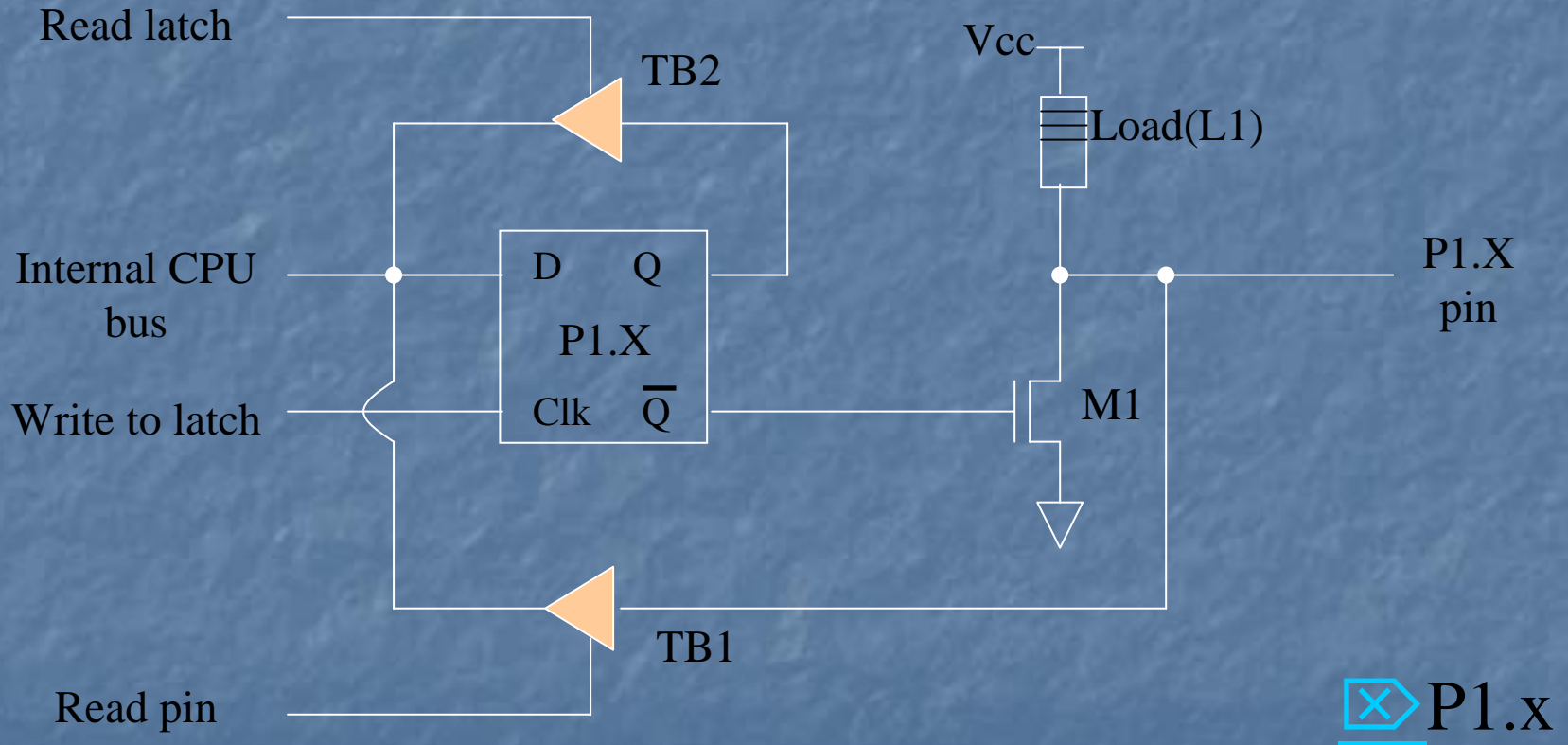
8051 Programming Model



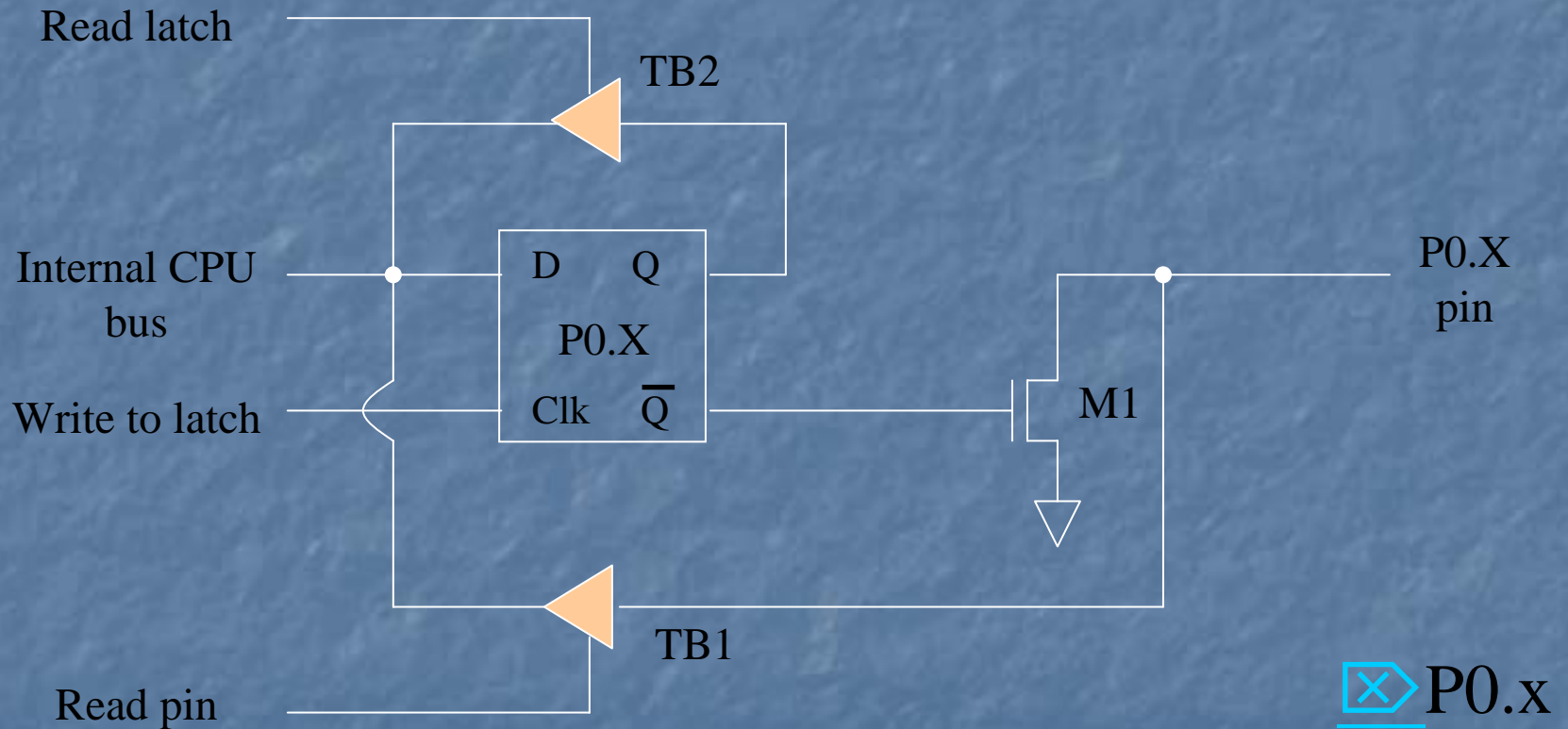
Pin Description of the 8051



A Pin of Port 1

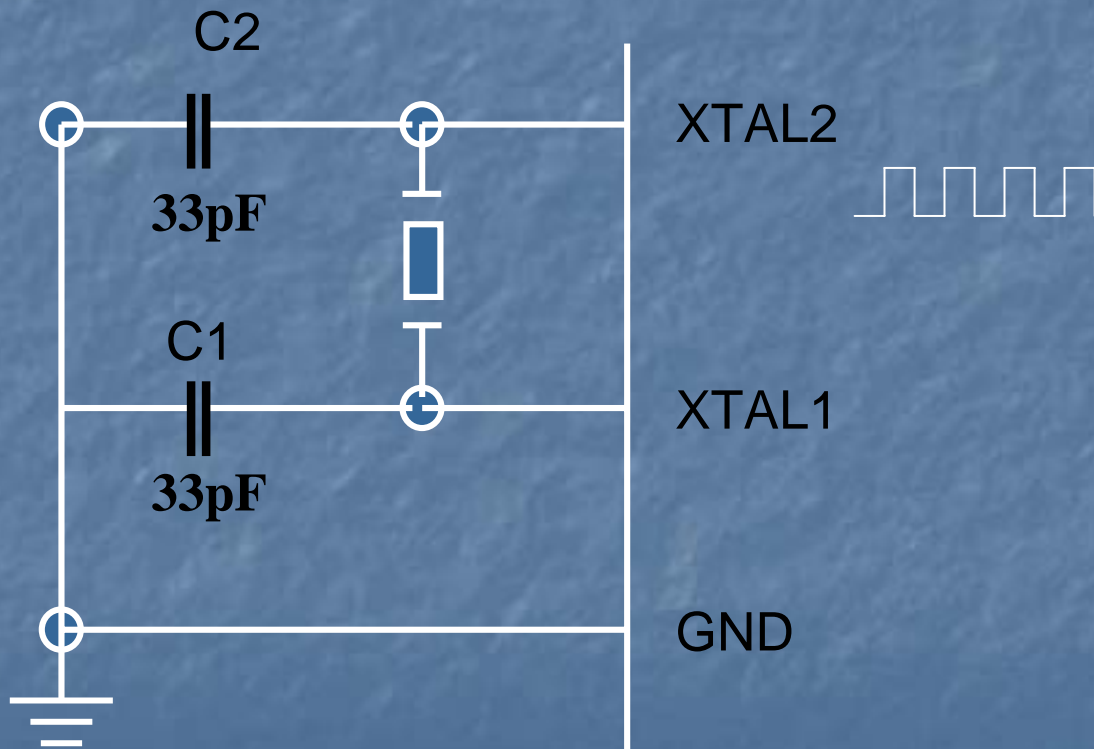


A Pin of Port 0



Crystal Oscillator connection

- Using a quartz crystal oscillator
- We can observe the frequency on the XTAL2 pin.



Port 3 Alternate Functions

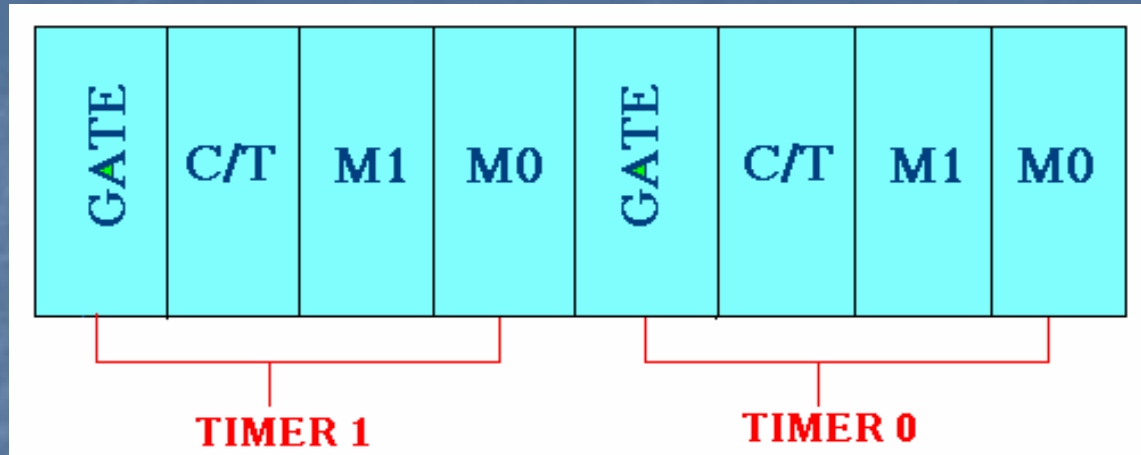
| P3 Bit | Function | Pin |
|--------|--------------------------|-----|
| P3.0 | RxD | 10 |
| P3.1 | TxD | 11 |
| P3.2 | $\overline{\text{INT0}}$ | 12 |
| P3.3 | $\overline{\text{INT1}}$ | 13 |
| P3.4 | T0 | 14 |
| P3.5 | T1 | 15 |
| P3.6 | $\overline{\text{WR}}$ | 16 |
| P3.7 | $\overline{\text{RD}}$ | 17 |

RESET Value of Some 8051 Registers

| Register | Reset Value |
|----------|-------------|
| PC | 0000 |
| ACC | 0000 |
| B | 0000 |
| PSW | 0000 |
| SP | 0007 |
| DPTR | 0000 |

RAM are all zero.

TMOD Register



- Gate : When set, timer only runs while INT(0,1) is high.
- C/T : Counter/Timer select bit. 0 to set it in timer mode
- M1 : Mode bit 1.
- M0 : Mode bit 0.

| M1 | M0 | MODE |
|----|----|------------------------|
| 0 | 0 | 13-bit timer mode |
| 0 | 1 | 16-bit timer mode |
| 1 | 0 | 8-bit auto-reload mode |
| 1 | 1 | split mode |

TCON Register

| | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|
| TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |
|-----|-----|-----|-----|-----|-----|-----|-----|

- TF1: Timer 1 overflow flag.
- TR1: Timer 1 run control bit.
- TF0: Timer 0 overflow flag.
- TR0: Timer 0 run control bit.
- IE1: External interrupt 1 edge flag.
- IT1: External interrupt 1 type flag.
- IE0: External interrupt 0 edge flag.
- IT0: External interrupt 0 type flag.

Interrupt Enable Register

| | | | | | | | |
|----|---|-----|----|-----|-----|-----|-----|
| EA | — | ET2 | ES | ET1 | EX1 | ET0 | EX0 |
|----|---|-----|----|-----|-----|-----|-----|

- EA : Global enable/disable.
- --- : Undefined.
- ET2 :Enable Timer 2 interrupt.
- ES :Enable Serial port interrupt.
- ET1 :Enable Timer 1 interrupt.
- EX1 :Enable External 1 interrupt.
- ET0 : Enable Timer 0 interrupt.
- EX0 : Enable External 0 interrupt.

Timer Interrupt Generation



Interrupt Interval:

$$t = \left[R_{\max} + 1 - R_{\min} \right]_d \times \frac{12_d}{f_{osc}}$$

$R_{\max} = \text{FFFF}$ (for 16-bit mode)

$= \text{FF}$ (for 8-bit mode)

$R_{\min} = \text{User settable}$

Interrupt Vector Address

| Type of Interrupt | Address |
|-------------------|---------|
| IE0 | 0003H |
| TF0 | 000BH |
| IE1 | 0013H |
| TF1 | 001BH |
| SERIAL | 0023H |

Assignment

Generate Square wave of two frequencies at one port pin of microcontroller on the basis of status of a toggle switch.

- Design the hardware and explain the scheme.

Programming the 8051

Modules:

❖ Initialization module

- Data Initializations (for RAM or SFRs)
- Configuration of Control registers
- Address Vectoring

❖ Run Module

- Main program for performing a specific task or group of tasks

Programming the 8051

Problem Statement:

State and Explain an algorithm for square wave generation

Programming the 8051

One Simple Solution:

Step 1: Make a port pin low

Step 2: Give necessary delay

Step 3: Toggle the port pin

Step 4: Go to Step 2.

Note: Multi-task is not possible with this algorithm

RTOS in the 8051

Concept:

All the tasks are time-multiplexed

Requirements:

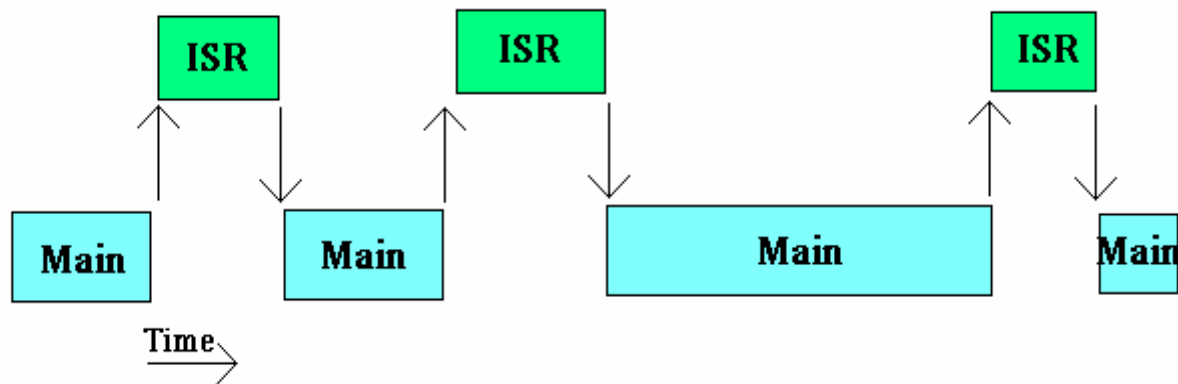
- Specified Time interval generation
- Overall tasks divided into two broad categories
 - Background tasks (Time critical job)
 - Foreground tasks (Non time critical job)

Interrupt

Program execution without interrupts :



Program execution with interrupts :



ISR : Interrupt Service Routine

RTOS in the 8051

Initialization Module

Resource for time interval generation:

Timer0 or Timer1

Design Issues:

- Mode of timer (16 or 8-bit)

```
tim_mod .equal 02h
```

```
mov TMOD, #tim_mod
```

- Computation of content to be used for initialization of timer

RTOS in the 8051

Design Issues:

- Generation of different time intervals using same timer interrupt

Use of multiplying factor with basic timer interrupt interval

```
TM0:    push ACC
        push PSW
;-----Basic time interval-----
```

```
tm_0:   mov A,tickN
        cjne A,#1,tm_1
```

```
;-----Basic time interval*tickN-----
        sjmp back
```

```
tm_1:   dec ACC
        mov tick10,A
```

```
back:   pop PSW
        pop ACC
        reti
```

RTOS in the 8051

Design Issues:

- Configuration of ports as input or output

`mov P0, #FFh ; Set port0 as input port`

`mov P1, #00h ;Set port1 as output`

- Address Vectoring

`.org 0`

`ajmp main`

`.org 000bh`

`ajmp TM0 ;jump to timer0 routine`

RTOS in the 8051

Run Module

Consists of infinite loop

```
main:  acall level1    ;jump to service routine for foreground task  
       sjmp main
```

Assignment Revisited

Generate Square wave of two frequencies at one port pin of microcontroller on the basis of status of a toggle switch.

- Write the code in assembly language for above problem

Implementation

Required Tools:

❖ Hardware: Universal Microcontroller Programmer

❖ Software:

- Assembler for 8051

Convert *.asm file to *.obj

- Linker for 8051

Links several *.obj files and converts the whole to *.hex

- Hex code downloader application program

Provided with universal programmer

Download the hex code into the microcontroller

All The Best