CRITERIA FOR SELECTING A MICROCONTROLLER

[illegible]

[illegible] analog & digital [illegible]

[illegible] the software architecture

The amount of computing power required will be one of the biggest requirements for the architecture & frequency of microcontroller

[illegible] the architecture

([illegible] or 32 bit)

Identify memory needs

[illegible] sure that you don't run out of [illegible] highest priority.

Start searching for microcontroller

Examine costs & power constraints

If the device will be powered from a battery [illegible] then making sure the parts are low power is absolutely [illegible]

Check part availability (lead time)

Select a development kit

Finding a development kit to play with & learn [illegible] of microcontroller [illegible] tools [illegible]

[illegible]

[illegible]

Like ECG LCD display [illegible] cell [illegible] analyzer

Communication systems :

Cellular phone, cable TV terminals, FAX

peripheral controllers in computer

Keyboard controller, DMA controller

<u>Instruments</u>

Industrial process controller, electronic smart [illegible] display system

[illegible]

<u>Embedded system applications</u>

(i) Home appliances : Microwave oven, refrigerators

(ii) Automobiles : engine control, climate control

(iii) Environmental control : green house, factory

(iv) Aerospace

3) Parallel Ports

→ 8 bit on chip ports
→ Programmable as input / output
→ Individual bits accessible on modified

→ Port with internal pull up
→ Open drain Port (Require external pull up)
→ Quasi bi directional (external pull up not required)

4) EEPROM & FLASH

→ EEPROM
  * Byte by byte erase/cycle
  * Row of byte erase/cycle
  * Complete chip erase/cycle
→ FLASH
  * Sector erase/cycle
  * Row of byte erase/cycle
  * Complete chip erase/cycle
  * Sector erase/cycle except OTP sector for boot

5) PWM (Pulse width Modulated Output)

→ Used for DAC
→ DAC is used in DC motor control & heater current control
→ width percentage proportional to modulation parameter P with which is related to analog voltage
→ Integrator is used to convert PWM to analog output
→ PWM device with a control register used as DAC

* Power up reset (PFFH)
* Reset instruction or timeout from watchdog timer (WDT)
* Push button reset or power reset
* Reset for two clock cycles.
* Reset address can be programmable.

2) WATCHDOG TIMER
→ Resets system after predefined timeout
→ On chip WDT
→ Timeout can be programmed
→ Implemented using a free running counter & o/p compare register.

9) BIT WISE MANIPULATIONS CAPABILITY
→ logical operation
→ set / reset / compare
→ Transfer
→ Bit o/p

6) ON CHIP ADC
- → Sample & Hold circuit
- → 2 reference analog inputs $V_{REF}(+)$ & $V_{REF}(-)$
- → Digital o/p is mapped to $V_{REF}(+) - V_{REF}(-)$
- → $V_{REF}(+)$ & $V_{REF}(-)$ can be programmable
- → if $V_a \geq V_{REF}(+)$ all o/ps are 1 & if $V_a \leq V_{REF}(-)$ all o/ps are 0s.

7) RESET CIRCUIT
- → Start up address
  - * 0000 in 8051
  - * FFFFH
- → 2 startup addresses (as defined in fixed memory location).
  - * Power up reset (FFFEH)
  - * Reset instruction or timeout from watch dog timer (FFFEH)
  - * Push button reset or power reset
  - * Reset for four clock cycles.
  - * Start address can be programmable.

8) WATCH DOG TIMER
- → Resets system after predefined timeout
- → On chip WDT
- → Timeout can be programmed
- → Implemented using a free running counter & o/p compare register.

9) BIT WISE MANIPULATION CAPABILITY
- → logical operation
- → set / reset / compare
- → Transfer
- → etc of one

10) TIMERS
- → Timer / counter

11) Serial Communication Port
- → Synchronous
- → Asynchronous

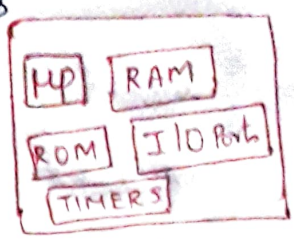
µp - Whenever programmer involved we use µp

MODULE - 5

Comparison b/w µp and µctrls

µp

µctrls



	µp	µctrls
1)	It contains ALU, GPRS Stack pointer, Program counter, clock timing circuit and interrupt circuit	It contains the circuitry of µp and in addition It has built in RAM, ROM, timers, I/O device etc.
2)	It has many instruction to move data between memory and CPU	It has 1 or 2 instruct to move data b/w m/m and CPU
3)	It has one or 2 bit handling instruction	It has many bit handling instructions.
4)	Access time for m/m and I/O devices are more.	Less acces time for built in m/m and I/O device
5)	µp based system require more hardware	µC based s/m requires less h/w reducing PCB size and increasing reliability.
6)	µp based system is more flexible in design point of view	less flexible in design point of view

7)	It has single m/m map for data and code	It has separate m/m map for data and code
8)	Less no. of pins are multifunctional	More no of pin are multifunctional.

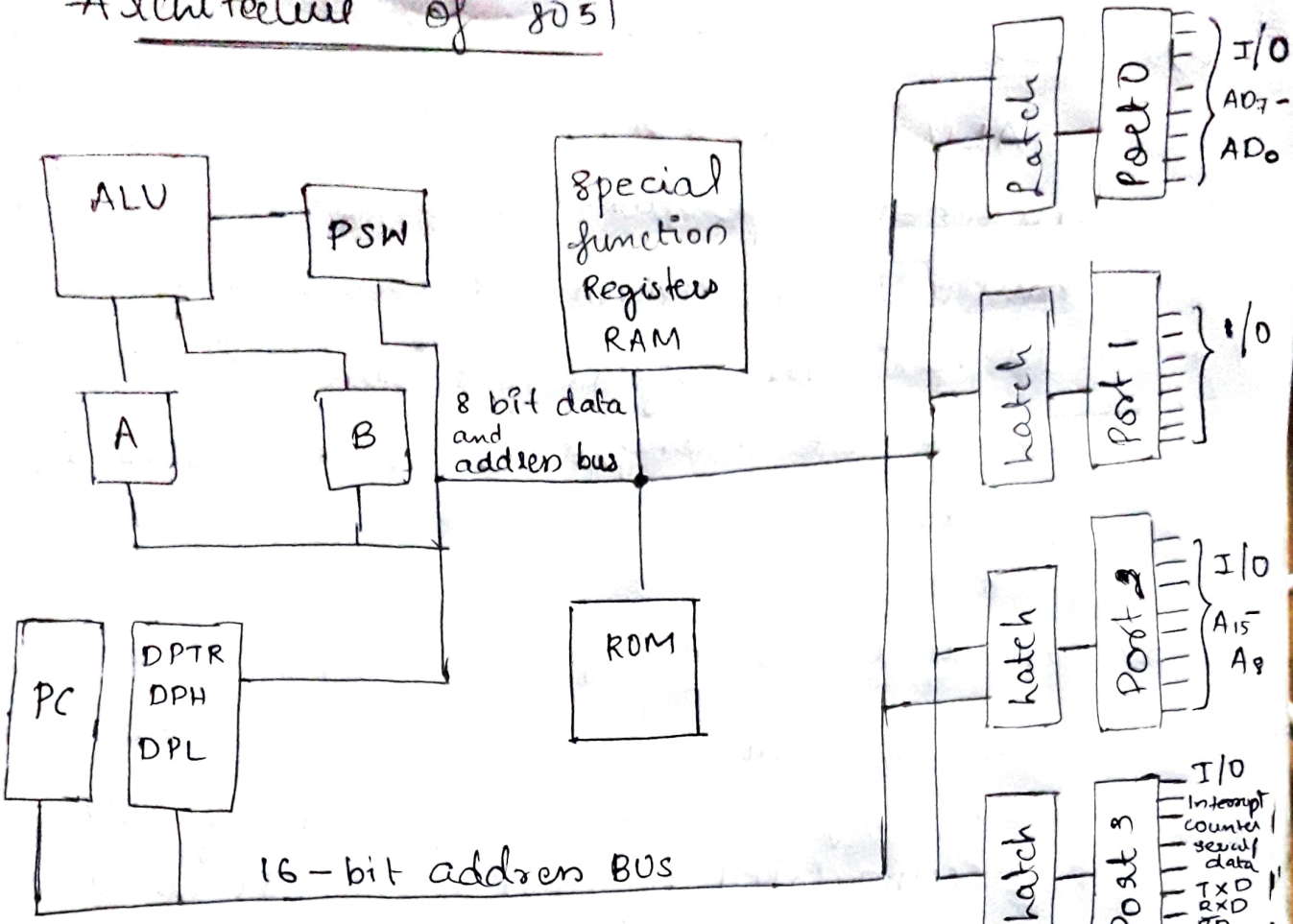
Architecture of 8051

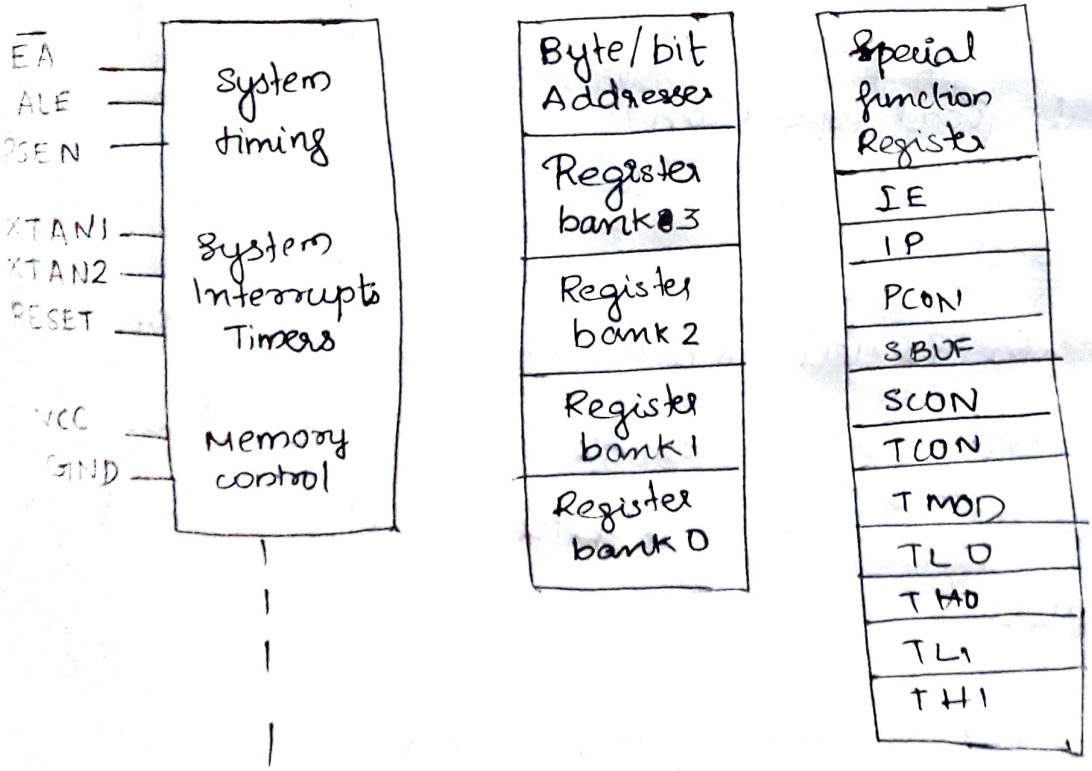
8051 is a 8bit microcontroller with 4kb of ROM and 128 bytes of RAM storage, 2 16 bit timers. It consist of 4 parallel 8bit ports. An on-chip crystal oscillator is integrated in the µC having crystal frequency of 12 MHz. It has 34 general purpose registers - A, B and 32 are bank of registers.

1) Oscillator and clock generator

All operations in a microcontroller are synchronized by the help of an oscillator clock. The oscillator clock generates the clk pulses by which the internal operations are synchronized. A resonant network is connected through pin XTAL1 and XTAL2 for CLK generation.

Architecture of 8051





Internal RAM structure

2) ALU

ALU is a 8 bit unit. It peeforms arithmetic operation as addition, subtraction, multiplication, division, increment, decrement etc. It peeforms logical operation like AND, OR, EXOR. It manipulates 8 bit and 16 bit data.

3) Accumulator (A register)

It is 8 bit register, result of arithmetic and logic operation peeformed by ALU is accumulated by this register. It is used to store 8 bit data and to hold one of the operand of ALU unit during arithmetical and logical operation

4) B-register

It is special 8 bit math register. It is bit and byte accesible. It is used in conjunction with A register as I/p operand for ALU.

Smoke.

5) PSW (Program status word)

CY	AC	F0	RS1	RS0	OV	-	P

1) P- parity flag

The bit will be set if ACC has odd number of 1's after an operation. If not, bit will remain cleared.

2) OV – overflow flag

It in signed arithmetic operations result exceeds more than 7 bit then OV flag is set. else resets. It is used in signed arithmetic operation only.

-ve number result

3) RS1 RS0

These 2 bits are used for RAM bank selection

RS1	RS0	
0	0	– Reg bank 0 is selected
0	1	– " 1 "
1	0	– " 2
1	1	– " 3 "

4) F0 :

It is user defined flag. The user defines the function of this flag. The user can set or reset this flag through s/w

AC - Auxillary carry

If during addtn and subtn any carry or borrow is generated from lower 4 bit to higher 4 bit then AC sets else it resets It is used in BCD arithmetic operation.

6) carry flag (CY)

During addition and subtractn any carry or borrow is generated then carry flag is set otherwise carry flag resets.

⑥ PC - Program Counter

The PC is a 2 byte address which tells the 8051 where the next instruction to execute is found in m/m. It is used to hold 16 bit address of internal RAM, external RAM or external ROM location.

When 8051 is initialized PC always starts at 0000H.

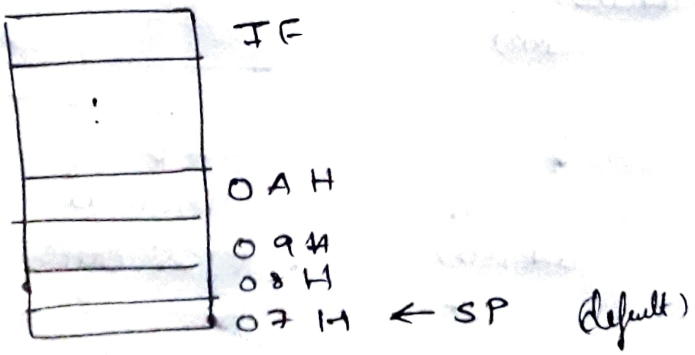
7) Data pointer register (DPTR)

It is a 16 bit register used to hold address of external or internal RAM where data is stored or result is to be stored.

It is divided into 2 8 bit registers.
DPH and DPL.

8) STACK Pointer (SP)

Stack is a section of RAM. The register used to access the stack is called the stack pointer (SP) register. The stack pointer in 8051 is 8 bit wide. When 8051 is powered up SP ~~code~~ contains the value 07H. This means RAM location ~~00~~ 08H is the first location used by stack by the 8051. PUSH instruction is used to store values to stack As data is pushed SP is incremented by 1
POP instruction is used to take value from the stack. As data is poped SP in decremented by 1.



9) Internal RAM

Internal RAM has min 128-byte. Internal RAM is organized into three distinct areas.

10) Internal ROM

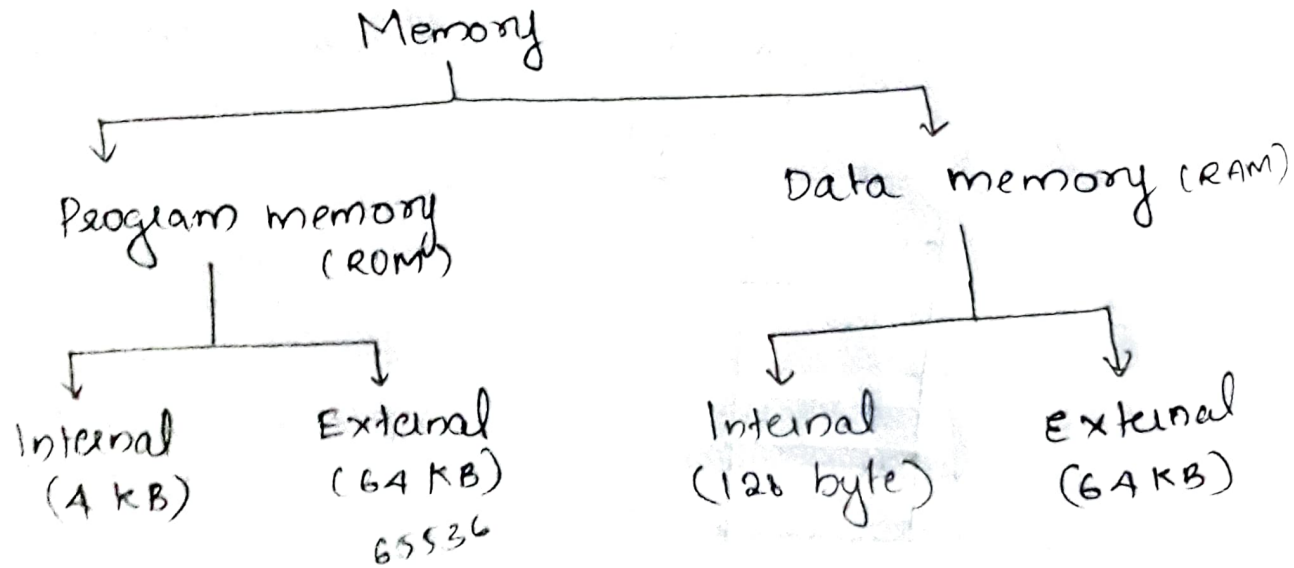
8051 has internal 4KB of internal ROM with address ranging from 0000H to 0FFFH

11) SFR

A program may inspect/change the Operating mode by manipulating the values of 8051 is SFR.

SFR are aliased as if they were normal Internal RAM address in the range 80H to FFH.

Memory Organisation



1 KB = 1024 B

4 KB x 8 = size

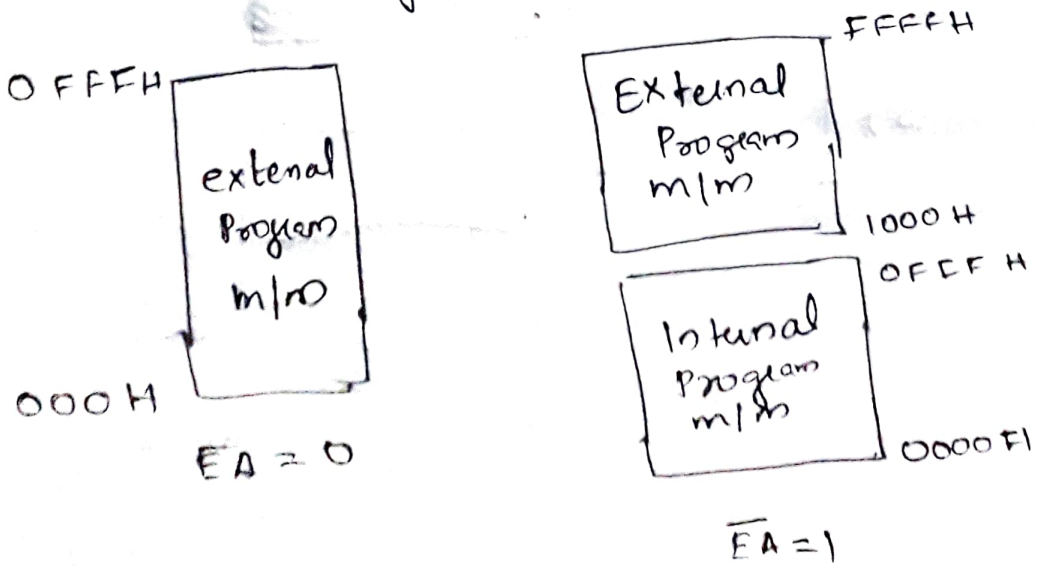
Program m/m

→ 8051 architecture provides lowest 4KB of program m/m as on chip m/m

→ switching b/w the internal program m/m and external program m/m is accomplished by changing the logic level of the Pin $\overline{EA}$ (external access)

→ connecting $\overline{EA}$ pin to logic 1, configures the chip to execute instruction from program m/m upto 4KB (0000H – 0FFFH) from internal m/m and 4KB from 1000H onwards from external m/m

→ connecting EA pin to logic 0 (GND) configures the chip to external program execution mode. when the entire code m/m is executed from the external m/m.



* Data Memory | Reg bank | Internal RAM

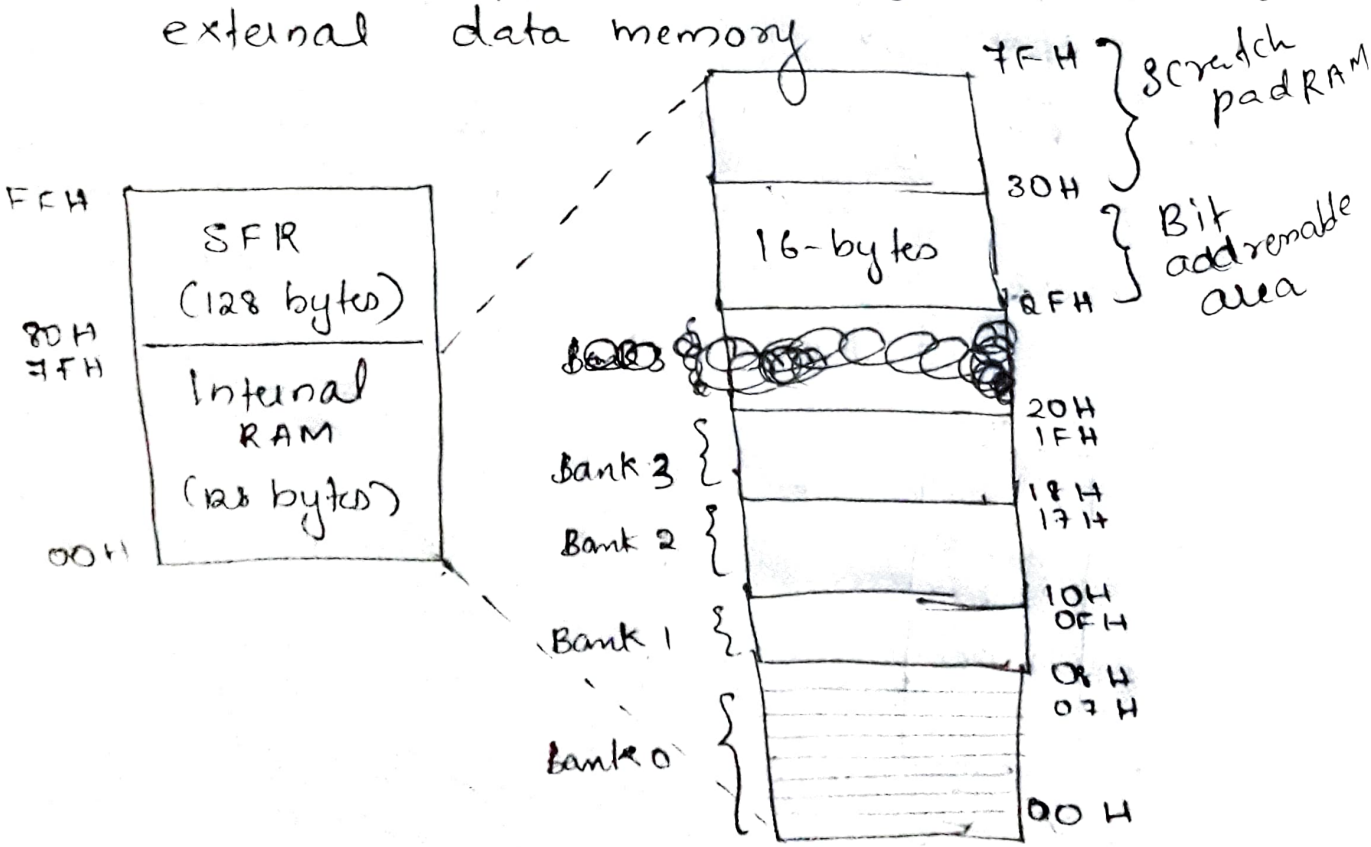
→ 8051 supports 128 bytes of internal data m/m and 128 bytes of special function registers m/m.

→ SFR m/m is n't available for general data m/m application

→ The address range for internal user data m/m is 00H to FFH

→ SFR are residing at memory area 80H to FFH

→ 8051 can support interfacing of 64 KB of external data memory



Lower 128 bytes internal data m/m organisat°

→ RAM area is volatile and the lowest 32 bytes of RAM 00H to 1FH are grouped into 4 banks of 8 registers each

→ There registers are known as R_0 to R_7 which are temporary data storage registers during program execution.

→ The next 16 bytes of RAM 20H to 2FH is bit addresable m/m, which accomodates 128 bits (16 byte x8) which can be accesed by direct bit addresing

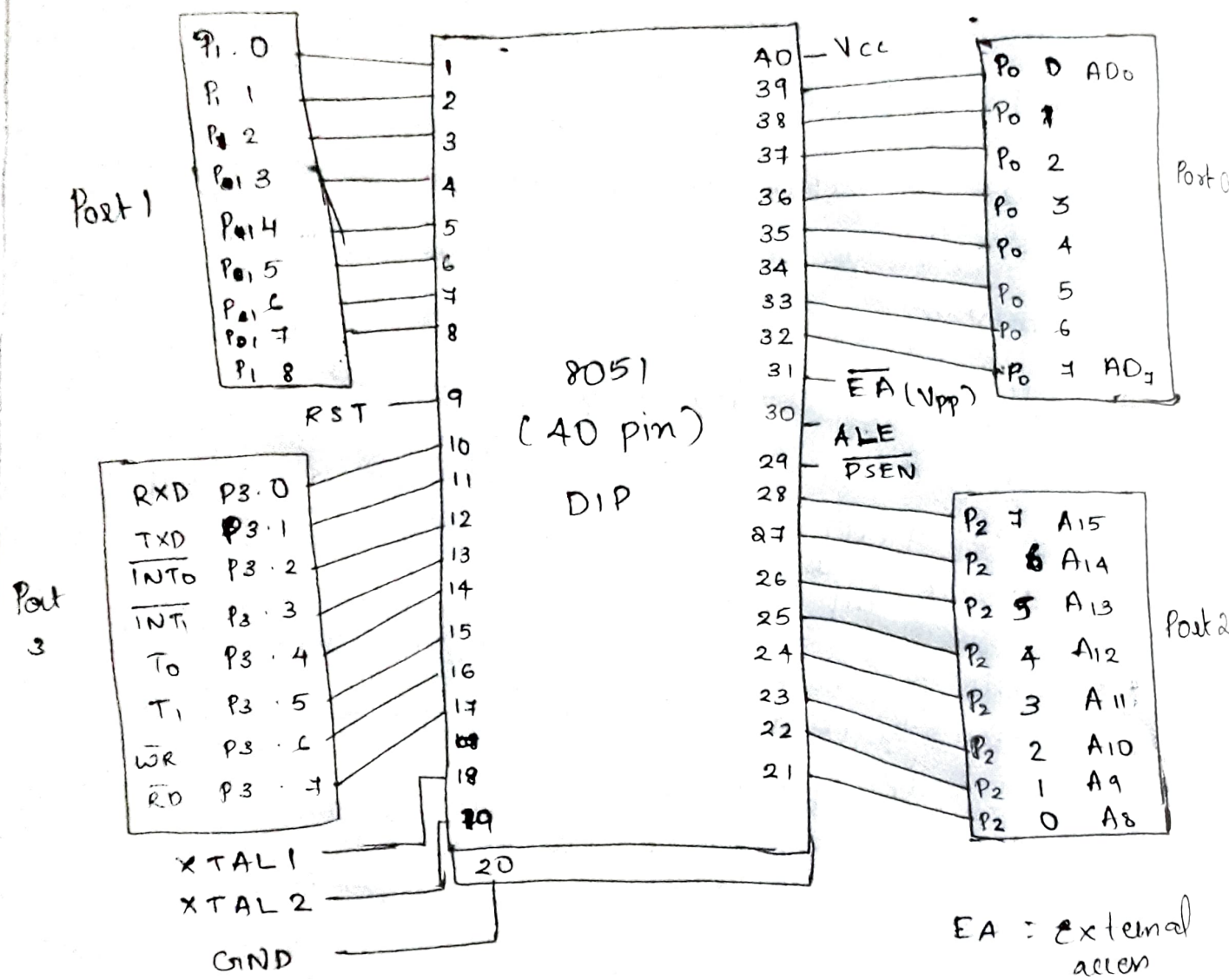
→ The next 80 bytes of RAM with address 30H to 7FH is used as general purpose scratch pad RAM

Upper 128 bytes RAM (SFR's)

→ The upper 128 bytes of RAM contains the SFRS

→ SFRS include port latches, status and control bits timers and value registers, CPU registers are 1 byte level accenible and some are both bit and byte level accenible

PIN CONFIGURATION AND FUNCTIONS



Port 0

→ Port 0 pins can be used as I/O pins

→ Port 0 also the low order byte of the external m/m address, time multiplexed with the data being written or read. So port 0 can be used as multiplexed address/data line

→ port 0 can be used as i/p port if logic 1 is written to the corresponding port SFR. and ports can be configured as output when we write logic 0 to port SFR

I/P port

```
MOV A, # FFH
MOV P0, A
```

Output port

```
MOV A, # 00H
MOV P0, A
```

Port 1

Port 1 pins can be used only as I/O pins port 1 can be configured as input port by writing logical 1 to 'pi' port register and writing logical 0 will configure the port as output port.

I/P Port

```
MOV A, # FFH
MOV P1, A
```

O/P port

```
MOV A, # 00H
MOV P1, A
```

Port 2

The port 2 are used to access external m/m. Port 2 outputs highers order bytes of the external m/m address, when the address is 16 bits otherwise. Port 2 is used as I/O ports

I/P Port

```
MOV A, #FFH
MOV P2, A
```

O/P port

```
MOV A, #00H
MOV P2, A
```

Port 3

All pin in port 3 are multifunctional. They have special function, which are described below. Apart from special functions, the port can be used as I/O ports.

I/P ports

```
MOV A, #FFH
MOV P3, A
```

O/P Port

```
MOV A, #00H
MOV P3, A
```

$P_{3.7}$	$\overline{RD}$	Read data control output
$P_{3.6}$	$\overline{WR}$	write data control output
$P_{3.5}$	T1	Timer/counter 1 external input
$P_{3.4}$	T0	Timer/counter 0 "
$P_{3.3}$	$\overline{INT_1}$	Interrupt 1 I/P pin
$P_{3.2}$	$\overline{INT_0}$	" 0 I/P pin
$P_{3.1}$	TXD	Transmit data pin for serial port
$P_{3.0}$	RXD	Receive data pin for serial port

Power supply pin Vcc

8051 operates on dc power supply of +5V with respect to ground.

Oscillator pins XTAL2 and XTAL1

For generating an internal clock signal the external oscillator is connected at these 2 pins.

ALE (Address batch enable)

→ $AD_0 - AD_7$ lines are multiplexed
→ ALE is used to demultiplex these lines, and to obtain lower half of an address, an external latch and ALE pin of 8051 is used.

RST (Reset)

→ To reset 8051

→ For proper reset operation, the reset sig must be held high at least for 2 machine cycles, while oscillator is running.

$\overline{PSEN}$ (Program store enable)

→ It is an active low o/p control sig used to activate the enable sig of the external ROM/EPROM

$\overline{EA}$ (external access)

When $\overline{EA}$ is high ie connected to Vcc, program fetches to address 0000H - 0FFFH which are directed to internal ROM and program fetches to address 1000H to FFFFH which are directed to external ROM/EPROM

When $\overline{EA}$ is low all address (0000 - FFFF) fetched by program are directed to the external ROM/EPROM.

Sources Of Interrupts in 8051

1) TIMER FLAG INTERRUPTS

Timer flag interrupt is generated when any one of the 2-timers T0 or T1 in 8051 overflows. when timers overflows, corresponding flags TF_0 or TF_1 is set to 1.

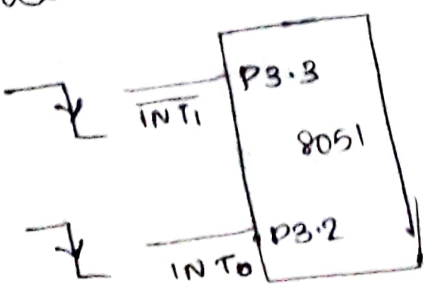
2) SERIAL PORT INTERRUPTS

If a data byte is received an interrupt bit RI is set to 1 in the SCON register.

when a data byte has been transmitted and interrupt bit TI is set in SCON register. These are ORed together to provide a single interrupt to the processor.

3) EXTERNAL INTERRUPTS

Pins $\overline{INT0}$ and $\overline{INT1}$ are used by external circuitry. Input on these pin can set the interrupt flags. IE_0 and IE_1 in the TCON register is set to 1 when any one of the external interrupt occurs



4) RESET

It is considered to be the ultimate interrup
bcos the pgm may not block the action
of the voltage on the RST pin. This type of
interrupt is often called non-maskable
interrupt.

Refer

SPECIAL FUNCTION REGISTER TO CONTROL INTERRUPTS

1) IE (interrupt enable special function Register)

7	6	5	4	3	2	1	0
EA	-	EI_2	ES	ET_1	EX_1	ET_0	EX_0

EA : Bit 7 : 0 - disable all interrupts
1 - enable all "

ET_2 : Bit 5 : Reserved for future use

ES : Bit 4 : 0 - disable serial port interrupt
1 - enable serial "

ET_1 : Bit 3 : 0 - disable timer 1 overflow interrup
1 - enable "

EX1
[redacted] : Bit 2 : 0 - disable [redacted] external interrupt 1

1 - enable "

ETo : Bit 1 : 0 - disable timer 0 overflow interrupt

1 - enable "

EXo : Bit 0 : 0 - disable external interrupt 0

1 - enable "

2) IP (interrupt priority special function register)

7	6	5	4	3	2	1	0
-	-	PT_2	PS	PT_1	PX_1	PT_0	PX_0

PT_2 - future use

PS - priority of serial/port interrupt

PT_1 - priority of Timer / overflow "

PX_1 - " external interrupt 1

PT_0 - " timer 0 overflow interrupt

PX_0 - " external interrupt 0

This register is used to alter the default priority scheme of interrupts. Storing '1' makes the corresponding interrupts to get high priority.

EX1

[redacted] : Bit 2 : 0 - disable [redacted] external interrupt 1

1 - enable "

ET0 : Bit 1 : 0 - disable timer 0 overflow interrupt

1 - enable "

EX0 : Bit 0 : 0 - disable external interrupt 0

1 - enable "

2) IP (interrupt priority special function register)

7	6	5	4	3	2	1	0
-	-	PT_2	PS	PT_1	PX_1	PT_0	PX_0

PT_2 - future use

PS - priority of serial/port interrupt

PT_1 - priority of Timer / overflow "

PX_1 - " external interrupt 1

PT_0 - " timer 0 overflow interrupt

PX_0 - " external interrupt 0

This register is used to alter the default priority scheme of interrupts. Storing '1' makes the corresponding interrupts to get high priority.