8051

i) 8bit CHMOS CPU   4KB ROM

ii) ~~256~~ 128 bytes on chip RAM

Registers

1) Accumulator
2) PSW
3) register B
4) 8 bit Stack pointer
5) 16 bit data pointer
6) P.C 16 bit
7) Program address register
8) 16 bit timer registers
9) inst. registers
10) control registers

4 banks, 8 registers each.
Physically these occupy first 32 bits of RAM.
Only one bank made active at a time.
bit 3 & bit 4 decide which ~~a~~ bank to make active.

Data Pointer: DPH → higher   DPL → lower

P0, P1, P2, P3 : Ports 0, 1, 2, 3.

Serial Data buffer: Transmit buffer register,
Receive buffer register.

Timer Registers: (TH0, TL0), (TH1, TL1), (TH2, TL2)
These are register pairs (16 bit) for Timer/Counter 0, 1, 2.

Capture registers: RCAP2H & RCAP2L are capture registers for Timer 2 capture mode.

PSW → Program Status Word

7	6	5	4	3	2	1	0
CY	AC	F0	RS1	RS0	OV	–	P

P → Parity
OV → overflow
RS0 & RS1 → Selection of register bank.
F0 → Used for general purpose
AC → Auxilliary carry
CY → Carry.

pins of 8051

RST→ Reseting the device

XTAL2→ It is the output of an inverting amplifier which is a part of on chip oscillator.

XTAL1→ input of an inverting amplifier which is a part of on chip oscillator.

$\overline{PSEN}$ → Program Store enable. It is read strobe ~~for~~ to ext mem.

ALE→ Address latch enable. Control signal.

$\overline{EA}$ → External access. Controls access of program memory.

When an ext mem. is attached to the 8051, PORT0 & ~~P0~~ PORT2 acts as multiplexed add./data bus.

2 lines of Port 3 → $\overline{RD}$ & $\overline{WR}$ generate

- - - - → for TxD & RxD for serial data transmission

- - - - - → ext. interrupts

- - - → ext. input for timers Timer 0 & Timer 1.

Interrupts

5 interrupts from → serial port, 2 ~~timer~~ from timers, 2 from ext. interrupts ($\overline{INT0}$ & $\overline{INT1}$)

Interrupt	Flag	Vector Addr.	Priority
Ext interrupt $\overline{INT0}$	IE0	0003	Highest
Timer/Counter0 Interrupt	TF0	000B	
Ext. interrupt $\overline{INT1}$	IE1	0013	
Timer/Counter1	TF1	001B	
Serial Port	RI or TI	0023	
Timer/Counter 2	TF2	002B	Lowest.

IE Register

	6	5	4	3	2	1	0
EA	X	ET2	ES	ET1	EX1	ET0	EX0

→ 1 → enable
0 → disable.

↳ 0 → ~~enables~~ disables all the interrupts
1 → Normal mode

Interrupt Priority:

7	6	5	4	3	2	1	0
X	X	PT2	PS	PT1	PX1	PT0	PX0

PX0 → 1 → set high priority
PX0 → 0 → set low priority.

Timers / Counters:

→ Counter / Timer selector

7	6	5	4	3	2	1	0
Gate	C/T	M1	M0	Gate	C/T	M1	M0
Timer/counter 1				Timer/counter 0			

M1	M0	Mode	
0	0	0	13 bit timer/counter
0	1	1	16 bit - - -
1	0	2	8 bit - - -
1	1	3	Timer/counter 0 operates as 2 separate timer/counters. Timer/Counter 1 cannot work in Mode 3. However can work in Mode 0,1,2.

Internal Data Memory Space

00-07H : Bank→0 : R0-R7 (working registers)
08-0FH : Bank→1 : R0-R7 (working registers)
10-17H : Bank→2 : - - - -
18-1FH : Bank→3 : - - - -

20H – 2FH : Bit addressable RAM locations.
(128 location for 128 bits)

30H – 7FH : General byte addressable.

Addressing Modes:

i) Register Addressing: 8 working registers of selected register bank are addressed by reg addr. Last 3 bits of opcode.

ii) Direct Addressing: Access SFR & lower 128 bits of RAM.

iii) Register indirect: The content of either R0 or R1 is used as pointer to access mem. locations in 256 bytes block. execution of PUSH, POP operations.

iv) Immediate → It is used for program memory.

v) Base + Index register Indirect Addressing →

eg: @ DPTR + A

or @ PC + A.

# → immediate addr   @ → content of.

MOVC A, @ A + DPTR

~~MOVC~~ → loads accumulator with a code byte or constant from the memory.

~~MOVX →~~

MOVX A, @Ri → Moves data from ext. mem to accumulator or vice versa.

XCH → exchange

XCHD → exchanges lower order nibble.

eg: XCHD A, @Ri ~~to~~ → $[A_{3-0}] \rightleftarrows [R_{3-0}]$

ADDC → Add with carry

SUBB → Subtract with borrow

ANL → Bitwise AND (logical)

ORL → " OR

XRL → " ~~X~~XOR

RL → Rotate left

RLC →

ACALL → Absolute call

LCALL → long subroutine call

RET → Returns from subroutine

RET → ″ ″ intrroupt sorvicedsubroutine.

AJMP → Absolute Jump

LJMP → Unconditional long jump.

SJMP → - - - short ″

JMP @A+DPTR → Jump indirect relative to DPTR.

CJNE → Compare & jump if not equal.

~~DJ~~ ~~DN~~ DJNZ direct, rel → Decrement the contene of specified direct address & jumps to rel if not zero.

JBC → Jump ~~in~~ if content of specified bit addr is set to 1.

JNB → - - - - - - - - - . not set.

SETB C → set carry flag.

SETB bit → set content of specified bit addr.