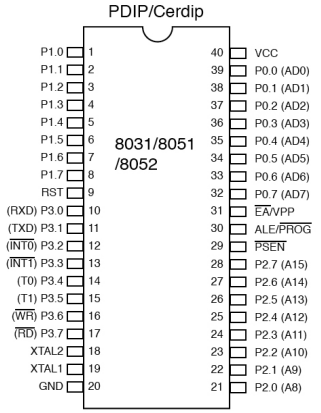
8051 Microcontroller:

Interrupts



Virendra Singh

Computer Architecture and Dependable Systems Lab

Department of Electrical Engineering

Indian Institute of Technology Bombay

http://www.ee.iitb.ac.in/~viren/

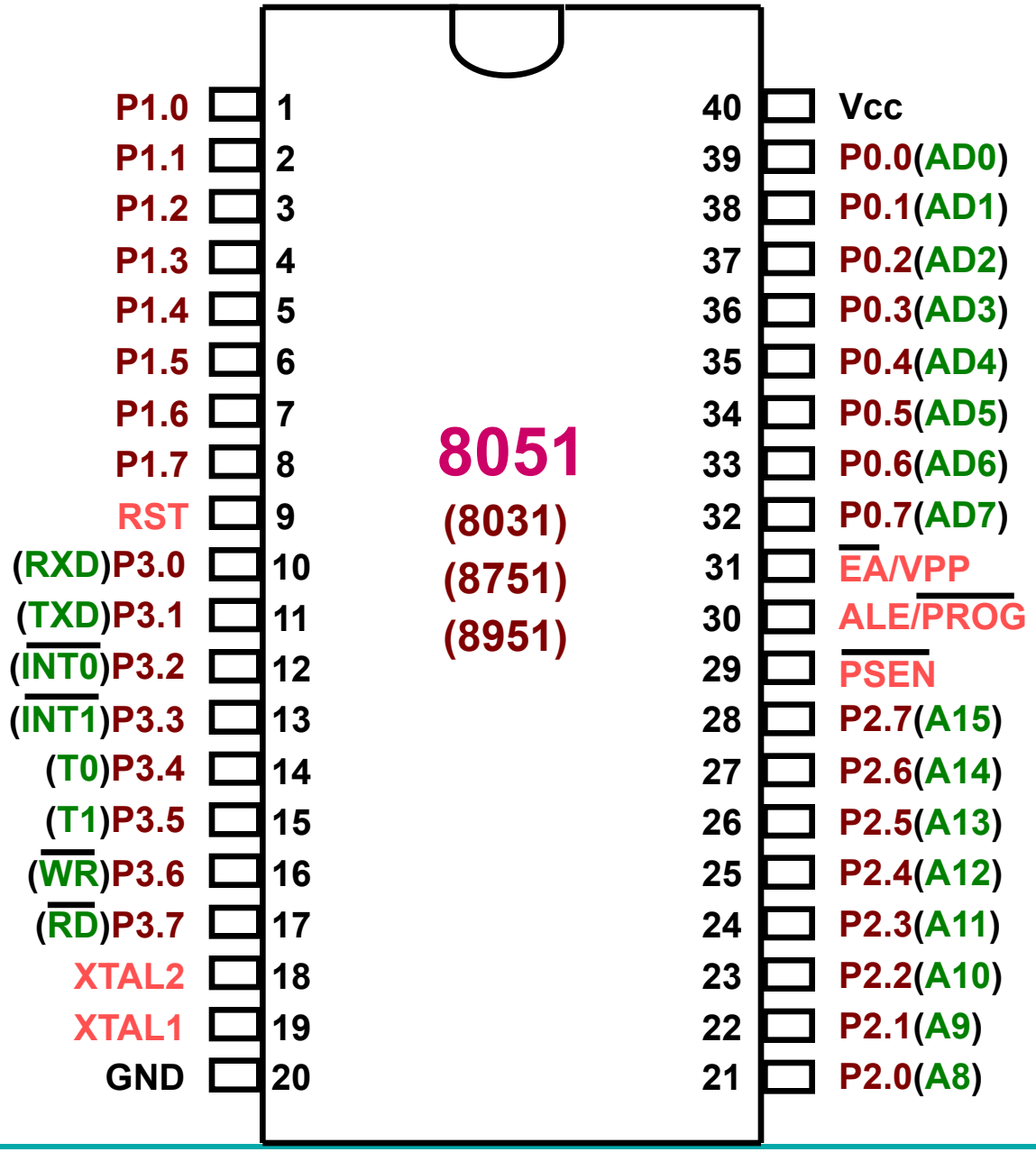
E-mail: viren@ee.iitb.ac.in

EE-309: Microprocessors

Lecture 10 (10 Aug 2015)

CADSL

8051 Pin Diagram



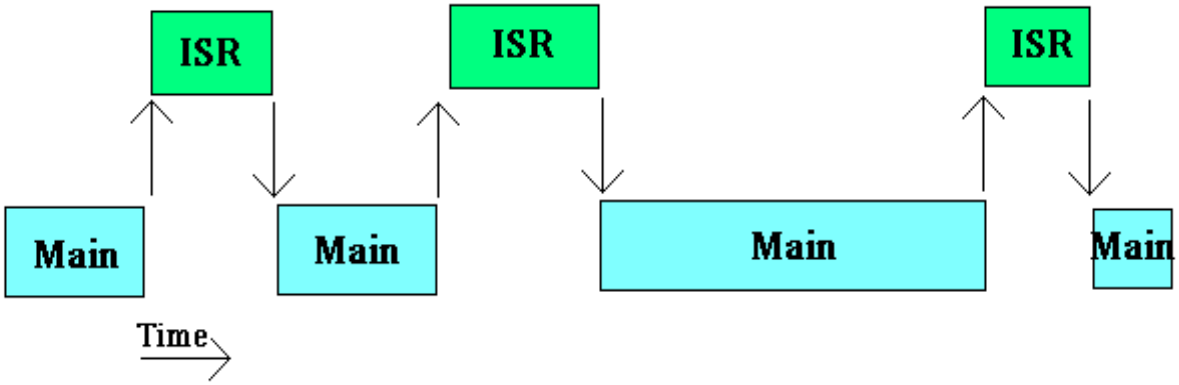
Interrupt

- An interrupt is the occurrence of a condition--an event -- that cause a temporary suspension of a program while the event is serviced by another program (Interrupt Service Routine ISR or Interrupt Handler).
- Interrupt is an asynchronous event
- Interrupt-Driven System-- gives the illusion of doing many things simultaneously, quick response to events, suitable for real-time control application.

Interrupt Execution

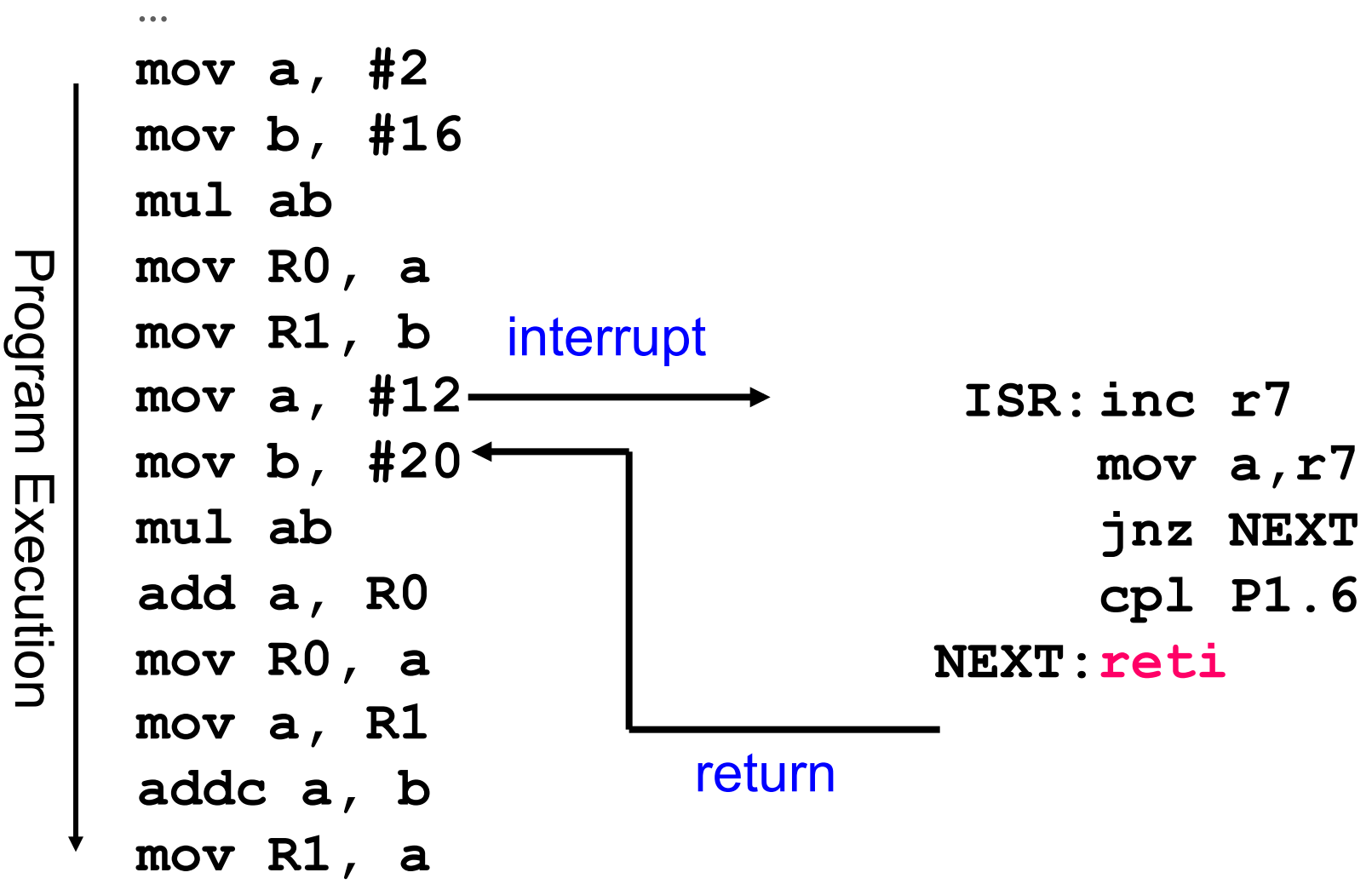
Program execution without intrrupts :

Program execution with intrrupts :



ISR : Intrrupt Service Routin

Interrupt Execution



Interrupt

- Microcontroller/ microprocessor finishes the execution of the current instruction
- Saves the address of the next instruction
- Also saves the current status of all interrupts (Not in status) – priority level FF
- For fast context switching it jumps to some fixed location

Interrupt Process

If interrupt event occurs AND interrupt flag for that event is enabled, AND interrupts are enabled, then:

1. Current PC is pushed on stack.
2. Program execution continues at the interrupt vector address for that interrupt.
3. When a RETI instruction is encountered, the PC is popped from the stack and program execution resumes where it left off.

Interrupt Sources

- Original 8051 has 5 sources of interrupts
  - Timer 0 overflow
  - Timer 1 overflow
  - External Interrupt 0
  - External Interrupt 1
  - Serial Port events (buffer full, buffer empty, etc)

- Enhanced version has 22 sources
  - More timers, programmable counter array, ADC, more external interrupts, another serial port (UART)

8051 Interrupt Organization

- 5 interrupt sources: 2 external, 2 timer, a serial port
- 2 programmable interrupt priority levels
- fixed interrupt polling sequence
- can be enabled or disabled
- IE (A8H), IP (B8H) for controlling interrupts

Interrupt Vectors

Each interrupt has a specific place in code memory where program execution (interrupt service routine) begins.

```
External Interrupt 0:     0003h
Timer 0 overflow:         000Bh
External Interrupt 1:     0013h
Timer 1 overflow:         001Bh
Serial :                  0023h
```

Note: that there are only 8 memory locations between vectors.

Interrupt Vectors

To avoid overlapping Interrupt Service routines, it is common to put JUMP instructions at the vector address. This is similar to the reset vector.

```
        org   0013H      ; at EX1 vector
        ljmp EX1ISR
        org 0100         ; at Main program
Main:    ...             ; Main program
         ...
EX7ISR:...               ; Interrupt service routine
        ...              ; Can go after main program
        reti             ; and subroutines.
```

Interrupt Execution

- Each Interrupt can be individually enabled or disabled

EA	X	X	ES	ET1	EX1	ET0	EX0

- Each source of interrupt can be programmed to one of two priority levels

X	X	X	PS	PT1	PX1	PT0	PX0

- Priority of an interrupt under execution should be stored

Enabling and Disabling Interrupts
IE (Interrupt Enable Register A8H)

Bit	Symbol	Bit Address	Description (1=enable, 0=disable)
IE.7	EA	AFH	Global enable/disable
IE.6	-	AEH	Undefined
IE.5	ET2	ADH	Enable timer 2 interrupt (8052)
IE.4	ES	ACH	Enable serial port interrupt
IE.3	ET1	ABH	Enable timer 1 interrupt
IE.2	EX1	AAH	Enable external 1 interrupt
IE.1	ET0	A9H	Enable timer 0 interrupt
IE.0	EX0	A8H	Enable external 0 interrupt

Two bits must be set to enable any interrupt: the individual enable bit and global enable bit

```
SETB ET1
SETB EA
MOV  IE,#10001000B
```

Interrupt Priorities

- What if two interrupt sources interrupt at the same time?
- The interrupt with the highest PRIORITY gets serviced first.
- All interrupts have a default priority order.
- Priority can also be set to “high” or “low”.

Interrupt Priority (IP, B8H)

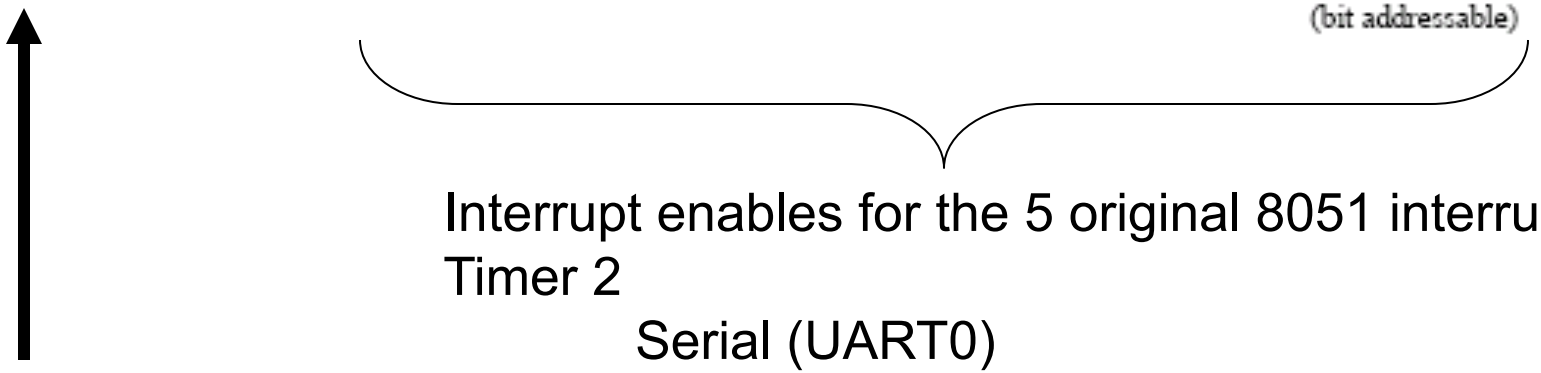
Bit	Symbol	Bit Address	Description (1=high, 0=low priority)
IP.7	-	-	Undefined
IP.6	-	-	Undefined
IP.5	PT2	BDH	Priority for timer 2 interrupt (8052)
IP.4	PS	BCH	Priority for serial port interrupt
IP.3	PT1	BBH	Priority for timer 1 interrupt
IP.2	PX1	BAH	Priority for external 1 interrupt
IP.1	PT0	B9H	Priority for timer 0 interrupt
IP.0	PX0	B8H	Priority for external 0 interrupt

- 0= lower priority, 1= higher priority, reset IP=00H
- Lower priority ISR can be interrupted by a high priority interrupt.
- A high priority ISR can not be interrupted.

Interrupt SFRs

Figure 12.9. IE: Interrupt Enable

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
EA	IEGF0	ET2	ES0	ET1	EX1	ET0	EX0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xA8



Interrupt enables for the 5 original 8051 interrupts:
Timer 2
Serial (UART0)
Timer 1
External 1
Timer 0
External 0

Global Interrupt Enable – must be set to 1 for any interrupt to be enabled

1 = Enable
0 = Disable

Thank You