Fire Bird V P89V51RD2

Interrupts

ERTS LAB
CSE IIT BOMBAY

By Pawankumar Suryawanshi

Interrupts

- As the name implies, an interrupt is some event which interrupts normal program execution.
- Various interrupts like:
  - Timer Overflow Interrupt
  - Timer Compare Interrupt
  - Serial interrupt
  - Wire & Wireless Interrupt
  - External hardware Interrupt

External hardware Interrupt

- Position encoder on the left motor is connected to INT1
- 30 pulse per revolution
- Diameter of wheel = 52mm
- Resolution of position encoder= πd/30= 5.44
- Pulse count = distance/ 5.44

4 pulse=4 Interrupts

Interrupts initialization

- To start interrupt service routine, we need to initialize some registers.
  - IEN0 – Interrupt Enable Register 0
  - TCON – Timer/Counter Resister

IEN0 – Interrupt Enable Register 0

Used to enable all interrupts servicing and enable external interrupt.

Bit	Symbol	Description	Bit Value
7	EA	Interrupt Enable Bit, EA=1; Interrupt Service Enable	1
6	EC	PCA Interrupt Enable Bit, EC=0; PCA interrupt disabled	0
5	ET2	Timer 2 Interrupt Enable Bit, ET2=0; Disable Timer 2 interrupt	0
4	ES	Serial Port Interrupt Enable Bit, ES=0; Disable Serial Port interrupt	0
3	ET1	Timer 1 overflow Interrupt Enable Bit, ET1=0; Disable Timer 1 overflow interrupt	0
2	EX1	External Interrupt 1 Enable; EX1=1; Enable external interrupt 1	1
1	ET0	Timer 0 overflow Interrupt Enable Bit, ET0=0; Disable Timer 0 overflow interrupt	0
0	EX0	External Interrupt 0 Enable; EX0=0; Disable external interrupt 0	0

IEN0=0X84;

TCON – Timer/Counter Resister

Used to set Interrupt 1 Trigger type to falling trigger.

Bit	Symbol	Description	Bit Value
7	TF1	Timer 1 overflow flag. Set by the hardware. Not used.	0
6	TR1	Timer 1 run control bit. TR1=0, Timer 1 not used.	0
5	TF0	Timer 1 overflow flag. Set by the hardware. Not used.	0
4	TR0	Timer 0 run control bit. TR0=0, Timer 0 not used.	0
3	IE1	Timer 1 overflow Interrupt Enable Bit, ET1=0; Disable Timer 1 overflow interrupt	0
2	IT1	Interrupt 1 Type control bit, IT1=1; Interrupt 1 falling edge trigger selected.	1
1	IE0	Interrupt 0 Edge flag. Set by the hardware. Not used.	0
0	IT0	Interrupt 1 Type control bit, IT1=1; IT0=0 or 1; Don't care.	0

TCON=0X04;

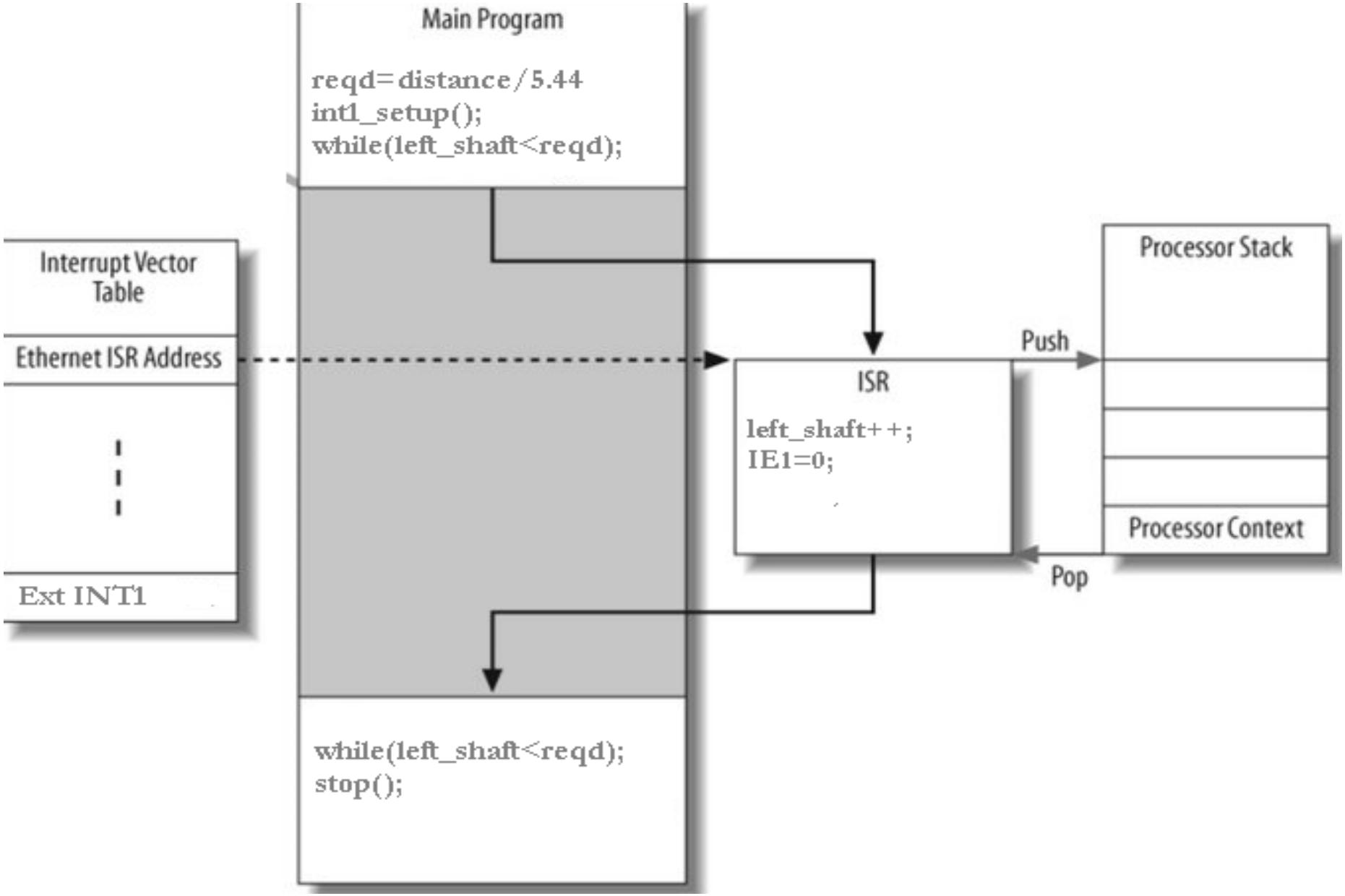
Format for writing Interrupt Service Routine

- **void** int1_isr(void)**interrupt** 2

Interrupt	Integer Constant
EXTERNAL_INT 0	0
TIMER/COUNTER0	1
EXTERNAL_INT1	2
TIMER/COUNTER 1	3
SERIAL PORT	4

TCON=0X04;

Interrupt Flow



Interrupt 1 Setup & Service Routine

```
void int1_setup()
{
TCON = 0x04; // set int 1 trigger type
IEN0 = 0x84; // enable all ISR servicing and enable int 1
P3 = 0x08; // set P3.3 (INT 1) as input port
}

void int1_isr(void)interrupt 2
{
left_shaft_count ++; //Increment Encoder count
IE1=0; //Reset Interrupt 1 flag
}
```

Main Program

```
reqd_shaft_count_int =(unsigned int)(distance *100 / 544);
//initialize external Interrupt 1

int1_setup();
//set motor enable pins to 1

left_velocity =1;
right_velocity =1;
forward();//go forward

while(left_shaft_count<reqd_shaft_count_int);
//wait till required distance moved
stop();
while(1);
```