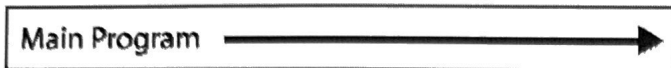
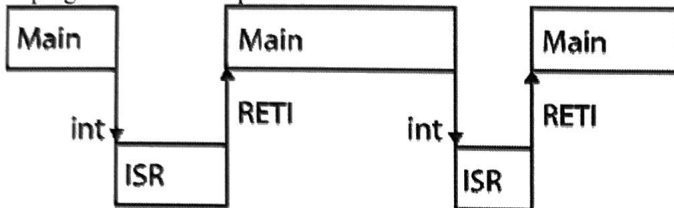


A program without interrupts:



A program with interrupts:



Interrupt Vectors

INTERRUPT	FLAG	VECTOR ADDR.
RESET	RST	0000h
External 0	IE0, TCON.1	0003h
Timer 0	TF0, TCON.5	000Bh
External 1	IE1, TCON.3	0013h
Timer 1	TF1, TCON.7	001Bh
Serial Port	R1 (SCON.0) or T1 (SCON.1)	0023h
Timer 2 (8052)	TF@ or EXF2	002Bh

1

A brief introduction to interrupts – another way to a 10kHz square wave:

```
.org 0
ljmp MAIN
org 000Bh
T0ISR:
    cpl P1.0
    reti
.ORG 0030h
MAIN:
    mov TMOD,#02h
    mov TH0, #0Ceh ;50 counts
    setb TR0
    mov IE,#82h
    loop: sjmp loop
```

Two Square Waves! 7kHz on P1.7 and 500Hz on P1.6 (approx).

```
.ORG 0
LJMP MAIN
ORG 0Bh
LJMP T0ISR
ORG 1Bh
LJMP T1ISR
.ORG 030h
MAIN:
    MOV TMOD, #12h
    MOV TH0, #0B9h
    SETB TR0
    SETB TF1
    MOV IE, #8Ah
    LOOP:
        SJMP LOOP
```

```
.ORG 200h
T0ISR:
    CPL P1.7
    RETI
T1ISR:
    CLR TR1
    MOV TH1, #0FCh
    MOV TL1, #18h
    SETB TR1
    CPL P1.6
    RETI
```

(FC18 is hex for 1000, i.e., the
T1ISR counts out 1ms.)
ASSUME 12MHz crystal!!!

2

Transmit the graphic (non-control) ascii characters:

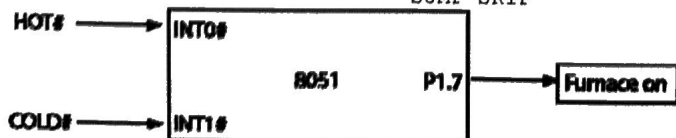
```
.ORG 0 ; 12 Mhz!!
LJMP MAIN
ORG 023h
LJMP SPISR
.ORG 030h
MAIN:
    MOV TMOD, #20h
    MOV TH1, #0E6h
    SETB TR1
    MOV SCON, #42h
    MOV A, #20h
    MOV IE, #90h
    LOOP:
        SJMP LOOP
```

```
.ORG 200h
SPISR:
    CJNE A, #7Fh, skip
    MOV A, #20h
    SKIP:
    MOV SBUF, A
    INC A
    CLR TI
    RETI
```

Again, assuming a 12Mhz crystal,
and a setup for 1200 baud serial
transmission.)

Controller for a furnace:

```
.ORG 0
LJMP MAIN
EX0ISR:
    CLR P1.7
    RETI
.ORG 013H
EX1ISR:
    SETB P1.7
    RETI
.ORG 30h
MAIN:
    MOV IE, #85h
    SETB IT0
    SETB IT1
    SETB P1.7
    JB P3.2, skip
    CLR P1.7
    SKIP:
    SJMP SKIP
```



3

Door buzzer: door opens, sounds alarm 400Hz tone for 1 sec.

```
.ORG 0
LJMP MAIN
ORG 00Bh;t0, 1sec interval
LJMP T0ISR
.ORG 01Bh;t0,400hz interval
LJMP T1ISR
.ORG 0030h
MAIN:
    SETB IT0
    MOV TMOD #11h
    MOV IE,#81h
    LOOP:
        SJMP LOOP
```

```
EX0ISR:
    MOV R7,#20
    SETB TF0
    SETB TF1
    SETB ET0
    SETB ET1
    RETI
```

```
T0ISR:
    CLR TR0
    DJNZ R7, SKIP
    CLR ET0
    CLR ET1
    LJMP EXIT
SKIP:
    ;3cb0h counts 0.05s
    MOV TH0,#3Ch
    MOV TLO,#0B0h
    SETB TR0
EXIT:
    RETI
T1ISR:
    CLR TR1
    MOV TH1, #0FBh
    MOV TL1, #01eh
    CPL P1.7
    SETB TR1
    RETI
```

4