

```

Include 'M64DEF.INC'          INT0_ISR : IN R18, PINB
.ORG 0x0000                    IN R26, PINB
JMP main                       IN R27, PINB
.ORG 0x0002                    CPI R18, 0x4F
JMP INT0_ISR                   BRNE next1
.ORG 0x0050                    LD R19, X
main : LDI R16, low(RAMEND)     ADD R19, R17
OUT SPL, R16                   ST X, R19
LDI R16, high(RAMEND)         RETI
OUT SPH, R16                   next1 : CPI R18, 0x6C
LDI R16, 0x00                 BRNE next2
STS EICRA, R16                LD R19, X
LDI R16, 0x01                 OUT PORTA, R19
OUT EIMSK, R16                RETI
LDI R16, 0x0FF                next2 : CPI R18, 0x0D8
OUT DDRA, R16                 BRNE next3
LDI R16, 0x00                 IN R19, PINC
OUT DDRB, R16                 ST X, R19
OUT DDRC, R16                 next3 : RETI
LDI R17, 0x03
SEI

```

loop : JMP Loop

$$f_t = \frac{8/192\text{MHz}}{1.24} = 8\text{KHz} \rightarrow T_t = \frac{1}{8} \text{m sec.} \rightarrow \frac{20\text{m sec.}}{T_t} = 160 \rightarrow 256 - 160 = 96 \quad -2$$

بعد از ۱۶۰ شمارش ، شمارنده وقفه تولید می کند که معادل 20 msec است. یعنی در هر 20 msec یک وقفه تولید می شود. برای داشتن ۳۰ ثانیه و ۲۰ ثانیه و ۱۰ ثانیه بایستی به ترتیب به تعداد ۱۵۰۰ و ۱۰۰۰ و ۵۰۰ وقفه را شمرد.

```

Include 'M64DEF.INC'
.ORG 0x0000
JMP main
.ORG 0x0020
JMP Timer0_ISR

.ORG 0x0050
main : LDI R16,low(RAMEND)
      OUT SPL,R16
      LDI R16,high(RAMEND)
      OUT SPH,R16
      LDI R16,0x07
      OUT TCCR0,R16
      LDI R16,96
      OUT TCNT0,R16
      LDI R16,0x01
      OUT TIMSK,R16
      LDI R16,0x38
      OUT DDRB,R16
      CBI PORTB,3
      SBI PORTB,4
      SBI PORTB,5
      CLR R17
      LDI R18,03
      CLR R19
Loop : SBIC PORTB,0
      JMP next
      LDI R16,0x00
      OUT TCCR0,R16
      JMP Loop
next : LDI R16,0x07
      OUT TCCR0,R16
      JMP Loop

Timer0_ISR : LDI R16,96
            OUT TCNT0,R16
            CALL EV_20ms
            RETI
EV_20ms : INC R17
        CPI R17,250
        BRNE EV_20m_end
        CLR R17
        CPI R18,03
        BRNE next1
        CALL EV_30s
        JMP EV_20m_end
next1 : CPI R18,02
        BRNE next2
        CALL EV_20s
        JMP EV_20m_end
next2 : CALL EV_10s
EV_20m_end : RET
EV_30s : INC R19
        CPI R19,06
        BRNE EV_30s_end
        CLR R19
        LDI R18,02
        SBI PORTB,3
        CBI PORTB,5
EV_30s_end : RET
EV_20s : INC R19
        CPI R19,04
        BRNE EV_20s_end
        CLR R19
        LDI R18,01
        SBI PORTB,5
        CBI PORTB,4
EV_20s_end : RET
EV_10s : INC R19
        CPI R19,02
        BRNE EV_10s_end
        CLR R19
        LDI R18,03
        SBI PORTB,4
        CBI PORTB,3
EV_10s_end : RET

```

$$f_t = \frac{4/0.96\text{MHz}}{1.24} = 4\text{KHz} \rightarrow T_t = \frac{1}{4} \text{msec.} \rightarrow \frac{1\text{m sec.}}{T_t} = 4 \rightarrow 256 - 4 = 252 \quad -3$$

هر یک میلی ثانیه تایمر یک وقفه تولید می کند که می توان با شمارش آن به تعداد $5000 = 50 \times 100$ تا، زمان 5 ثانیه را اندازه گیری کرد. هر 1msec. باید پورت B، یک رقم افزایش یابد که در این صورت خروجی DAC نیز 2 mV افزایش می یابد. تا خروجی DAC به 20 mV برسد که به مدت 6 msec. ثابت می ماند و سپس مجدداً صفر می شود.

```

Include 'M64DEF.INC'
.ORG 0x0000
JMP main
.ORG 0x0020
JMP Timer0_ISR
.ORG 0x0050
main : LDI R16, low(RAMEND)
      OUT SPL, R16
      LDI R16, high(RAMEND)
      OUT SPH, R16
      LDI R16, 0x07
      OUT TCCR0, R16
      LDI R16, 252
      OUT TCNT0, R16
      LDI R16, 0x01
      OUT TIMSK, R16
      LDI R16, 0x00
      OUT DDRA, R16
      LDI R16, 0xFF
      OUT DDRB, R16
      CLR R16
      CLR R17
      CLR R18
      CLR R20
      CLR R19
      CLR R21
      LDI R22, 10
      SEI
Loop : JMP Loop

Timer0_ISR : LDI R16, 252
            OUT TCNT0, R16
            CPI R20, 0
            BREQ next1
            CPI R19, 0
            BREQ next2
            INC R21
            CPI R21, 10
            BRLO next3
            CPI R21, 16
            BREQ next4
            OUT PORTB, R22
            JMP next1
next3 : OUT PORTB, R21
      JMP next1
next4 : OUT PORTB, R21
      CLR R21
      OUT PORTB, R21
      DEC R19
      BRNE next1
      CLR R20
      JMP next1
next2 : OUT PORTB, R19
next1 : CALL EV_100ms
      RETI
EV_100ms : INC R17
          CPI R17, 100
          BRNE EV_100ms_end
          CLR R17
          CALL EV_5s
EV_100ms_end : RET
EV_5s : INC R18
       CPI R18, 50
       BRNE EV_5s_end
       CLR R18
       IN R19, PINA
       LDI R20, 01
EV_5s_end : RET

```