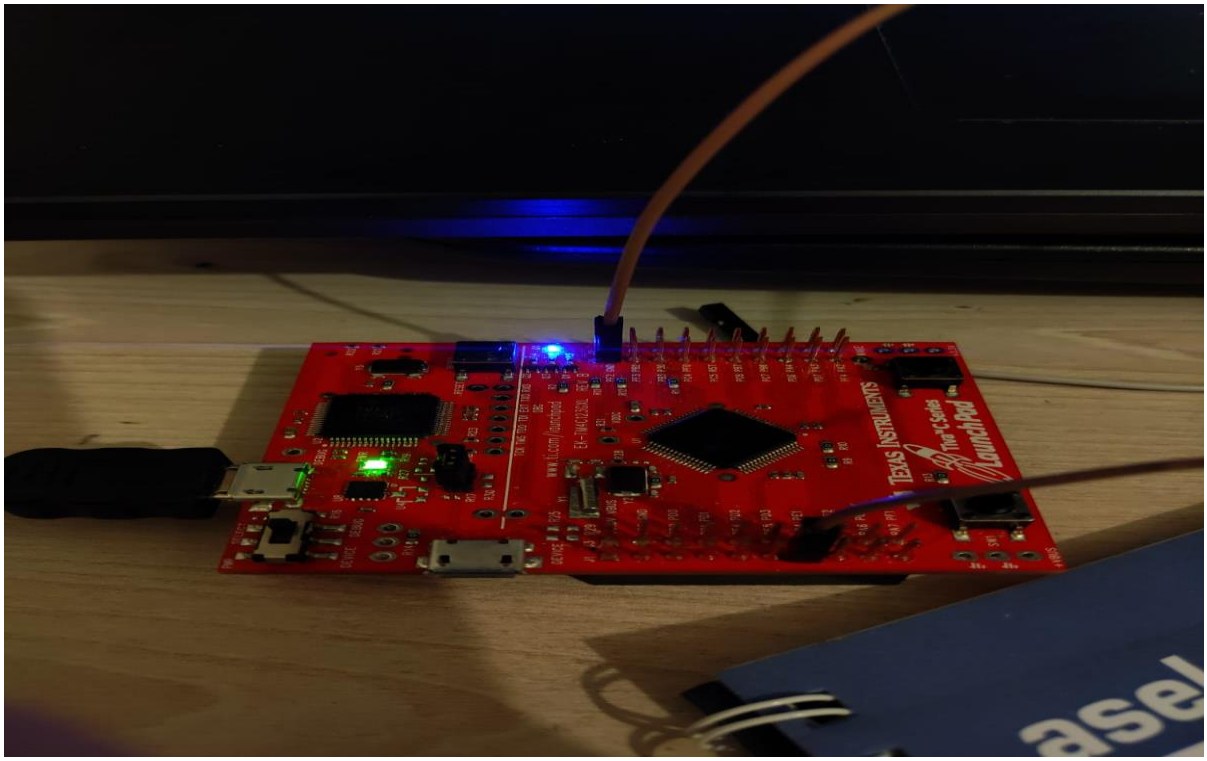
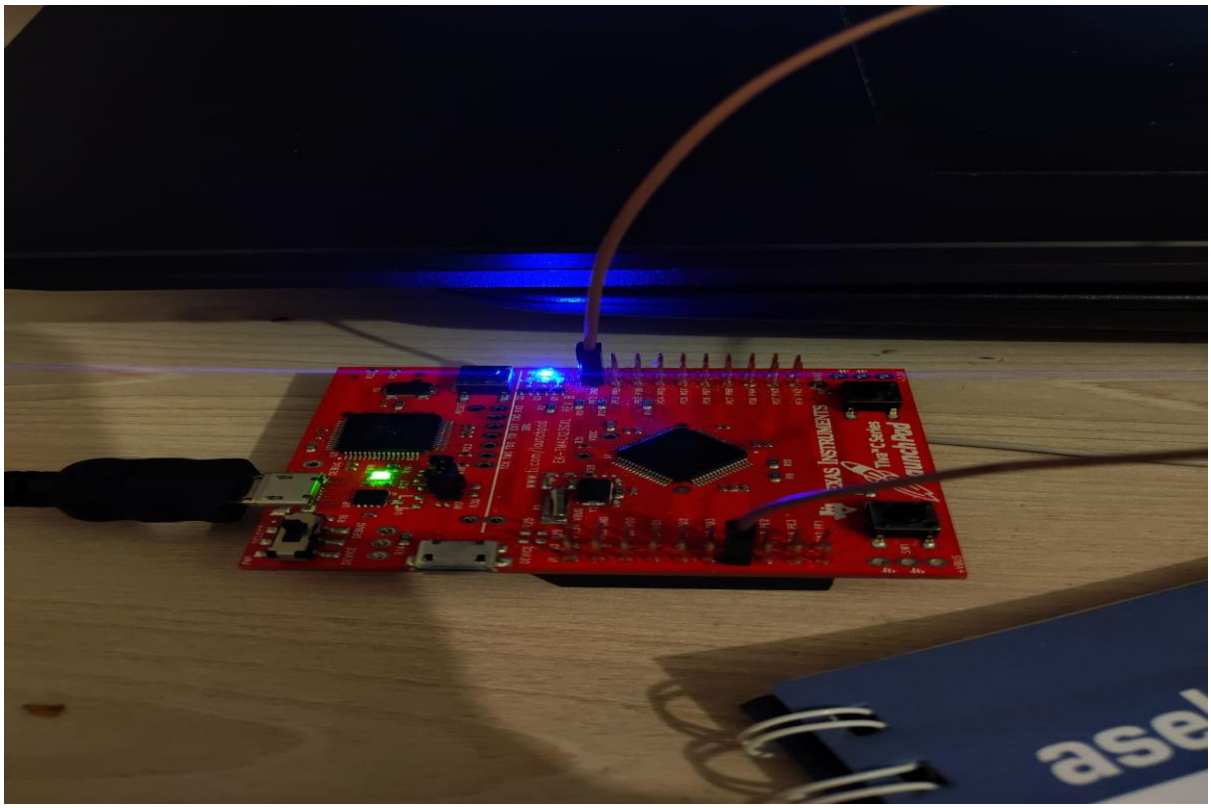
EE447 PRELIM4

EFE BERKAY YITIM 2305761

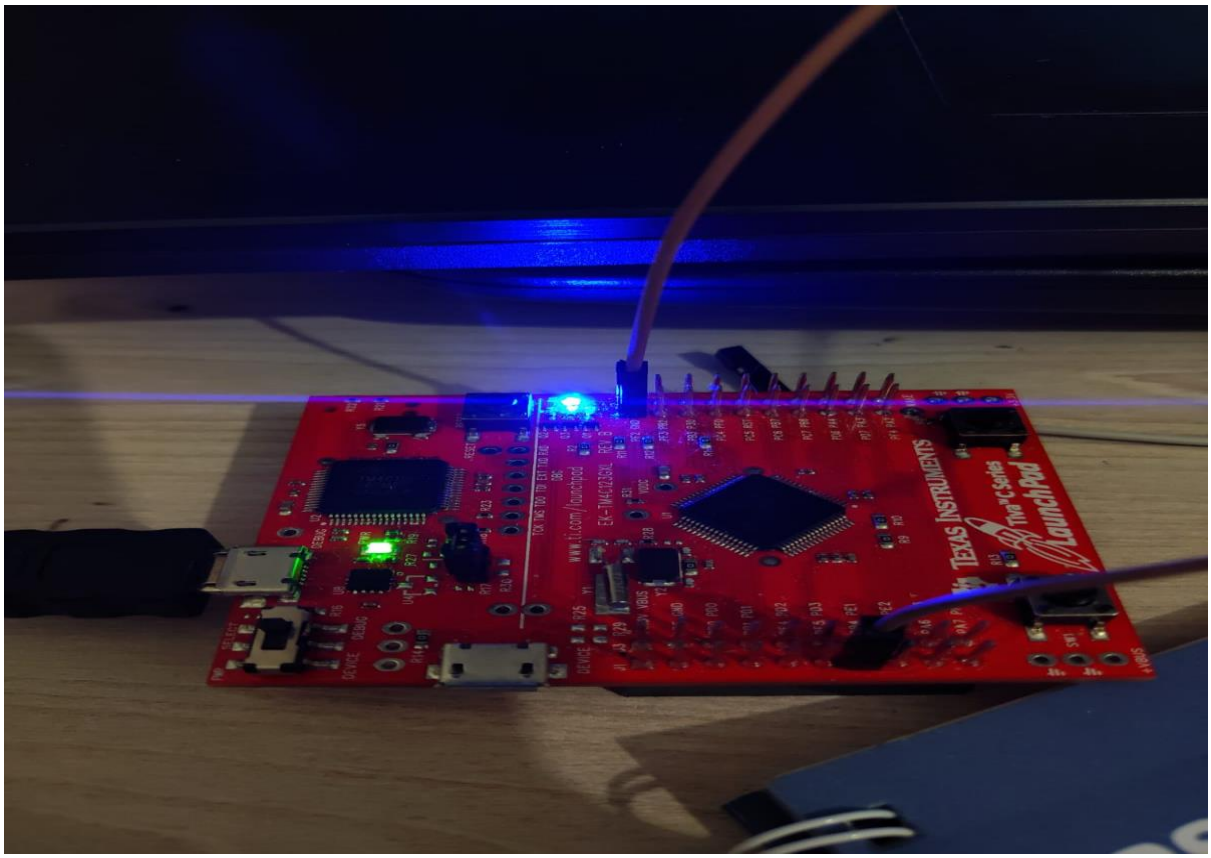
Q1) DUTY CYCLE 5.88%



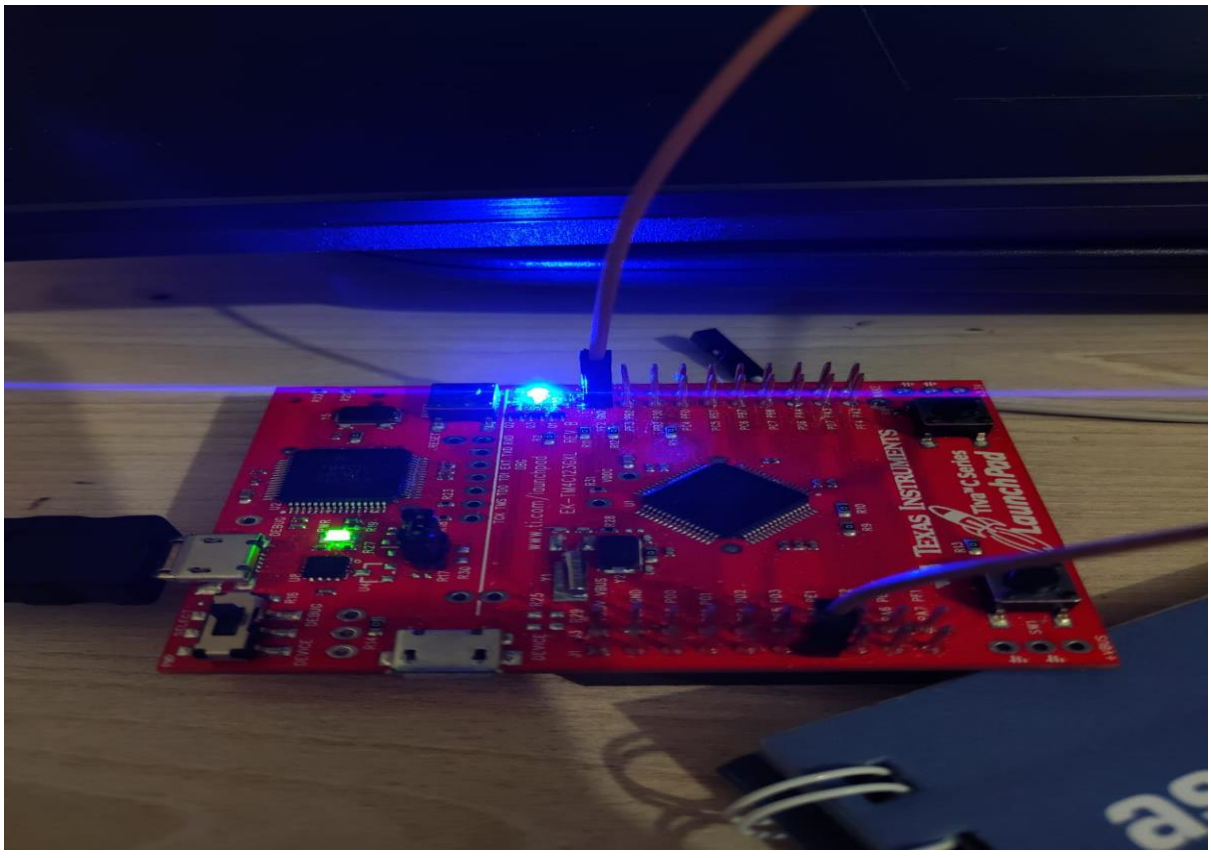
DUTY CYCLE 20%



DUTY CYCLE 50%



DUTY CYCLE 93.75%



```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                 MAIN OF THE Q1                  ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;LABEL           DIRECTIVE   VALUE               COMMENT
                 AREA main,  CODE,    READONLY,   ALIGN=2
                 THUMB

                 IMPORT      PULSE_INIT
                 EXPORT      __main


__main           PROC
                 BL      PULSE_INIT
LOOP             WFI
                 B       LOOP

                 ENDP
                 END
```

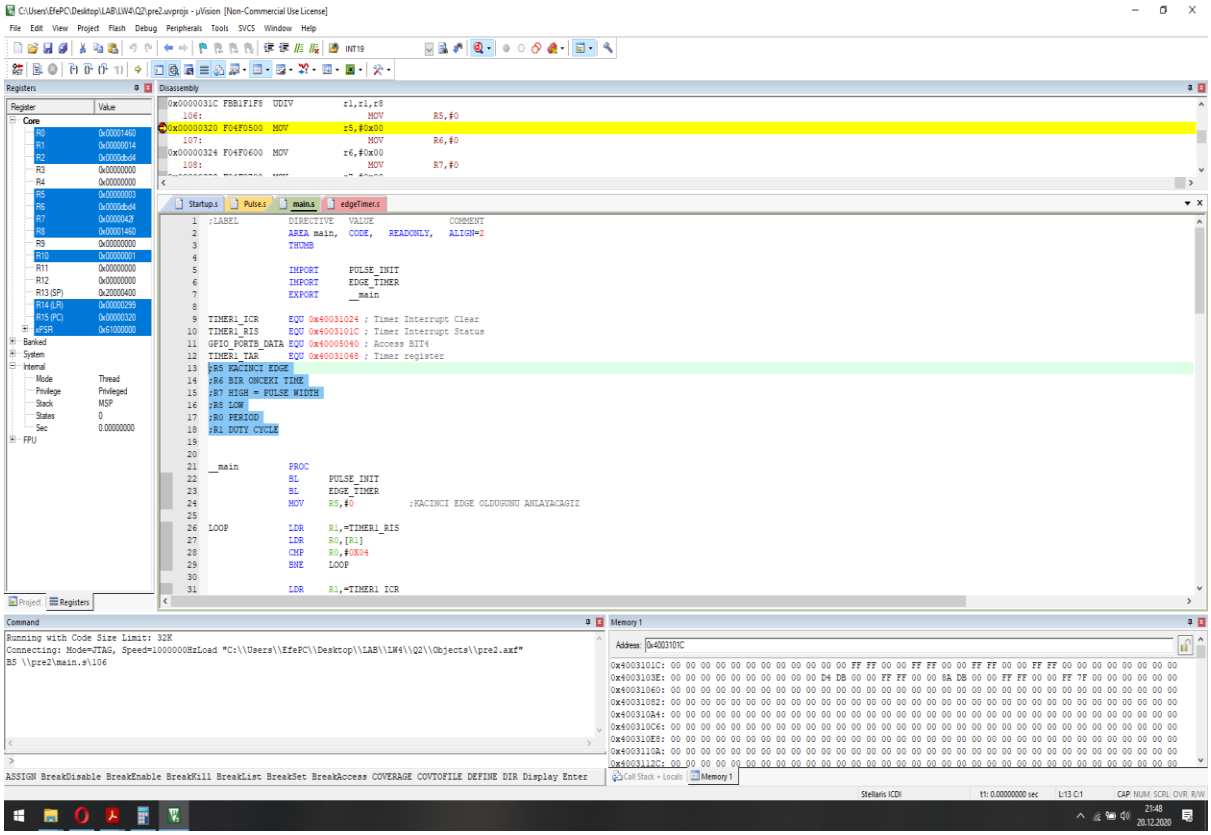
```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                  PULSE OF THE Q1                  ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; Pulse.s
; Routine for creating a pulse train using interrupts
; This uses Channel 0, and a 1MHz Timer Clock (_TAPR = 15 )
; Uses Timer1A to READ EDGES on PB4

;Nested Vector Interrupt Controller registers
NVIC_EN0_INT19      EQU 0x00080000 ; Interrupt 19 enable
NVIC_EN0            EQU 0xE000E100 ; IRQ 0 to 31 Set Enable Register
NVIC_PRI4           EQU 0xE000E410 ; IRQ 16 to 19 Priority Register

; 16/32 Timer Registers
TIMER0_CFG          EQU 0x40031000
TIMER0_TAMR         EQU 0x40031004
TIMER0_CTL          EQU 0x4003100C
TIMER0_IMR          EQU 0x40031018
TIMER0_RIS          EQU 0x4003101C ; Timer Interrupt Status
TIMER0_ICR          EQU 0x40031024 ; Timer Interrupt Clear
TIMER0_TAILR        EQU 0x40031028 ; Timer interval
TIMER0_TAPR         EQU 0x40031038
TIMER0_TAR          EQU 0x40031048 ; Timer register

;GPIO Registers
GPIO_PORTF_DATA     EQU 0x40025010 ; Access BIT2
GPIO_PORTF_DIR      EQU 0x40025400 ; Port Direction
GPIO_PORTF_AFSEL    EQU 0x40025420 ; Alt Function enable
GPIO_PORTF_DEN      EQU 0x4002551C ; Digital Enable
GPIO_PORTF_AMSEL    EQU 0x40025528 ; Analog enable
GPIO_PORTF_PCTL     EQU 0x4002552C ; Alternate Functions

;System Registers
SYSCTL_RCGCGPIO     EQU 0x400FE608 ; GPIO Gate Control
SYSCTL_RCGCTIMER    EQU 0x400FE604 ; GPTM Gate Control

;---------------------------------------------------
LOW                 EQU 0x00000100
HIGH                EQU 0x00000040
;---------------------------------------------------

            AREA    routines, CODE, READONLY
            THUMB
            EXPORT  My_Timer0A_Handler
            EXPORT  PULSE_INIT

;---------------------------------------------------
My_Timer0A_Handler  PROC
                    ;...
                    ADD     R10,#1
                    CMP     R10,#1
                    BEQ     HIGHX

LOWX                LDR R0,=GPIO_PORTF_DATA
                    LDR R1,[R0]
                    MOV R1,#0
                    STR R1,[R0]
                    LDR R1, =TIMER0_TAILR ; initialize match clocks
                    LDR R2, =LOW
                    STR R2, [R1]
                    MOV R10,#0
                    B   EXIT

HIGHX               LDR R0,=GPIO_PORTF_DATA
                    LDR R1,[R0]
                    MOV R1,#4
                    STR R1,[R0]
                    LDR R1, =TIMER0_TAILR ; initialize match clocks
                    LDR R2, =HIGH
                    STR R2, [R1]
                    B   EXIT
```

```
 73
 74  EXIT                LDR R0,=TIMER0_ICR
 75                      ORR R1,#0X01
 76                      STR R1,[R0]
 77                      BX  LR
 78                      ENDP
 79  ;-------------------------------------------------
 80
 81  PULSE_INIT  PROC
 82              LDR R1, =SYSCTL_RCGCGPIO ; start GPIO clock
 83              LDR R0, [R1]
 84              ORR R0, R0, #0x20 ; set bit 5 for port F
 85              STR R0, [R1]
 86              NOP ; allow clock to settle
 87              NOP
 88              NOP
 89              LDR R1, =GPIO_PORTF_DIR ; set direction of PF2
 90              LDR R0, [R1]
 91              ORR R0, R0, #0x04 ; set bit2 for output
 92              STR R0, [R1]
 93              LDR R1, =GPIO_PORTF_AFSEL ; regular port function
 94              LDR R0, [R1]
 95              BIC R0, R0, #0x04
 96              STR R0, [R1]
 97              LDR R1, =GPIO_PORTF_PCTL ; no alternate function
 98              LDR R0, [R1]
 99              BIC R0, R0, #0x00000F00
100              STR R0, [R1]
101              LDR R1, =GPIO_PORTF_AMSEL ; disable analog
102              MOV R0, #0
103              STR R0, [R1]
104              LDR R1, =GPIO_PORTF_DEN ; enable port digital
105              LDR R0, [R1]
106              ORR R0, R0, #0x04
107              STR R0, [R1]
108
109              LDR R1, =SYSCTL_RCGCTIMER ; Start Timer0
110              LDR R2, [R1]
111              ORR R2, R2, #0x01
112              STR R2, [R1]
113              NOP ; allow clock to settle
114              NOP
115              NOP
116              LDR R1, =TIMER0_CTL ; disable timer during setup LDR R2, [R1]
117              BIC R2, R2, #0x01
118              STR R2, [R1]
119              LDR R1, =TIMER0_CFG ; set 16 bit mode
120              MOV R2, #0x04
121              STR R2, [R1]
122              LDR R1, =TIMER0_TAMR
123              MOV R2, #0x02 ; set to periodic, count down
124              STR R2, [R1]
125              LDR R1, =TIMER0_TAILR ; initialize match clocks
126              LDR R2, =LOW
127              STR R2, [R1]
128              LDR R1, =TIMER0_TAPR
129              MOV R2, #15 ; divide clock by 16 to
130              STR R2, [R1] ; get 1us clocks
131              LDR R1, =TIMER0_IMR ; enable timeout interrupt
132              MOV R2, #0x01
133              STR R2, [R1]
134  ; Configure interrupt priorities
135  ; Timer0A is interrupt #19.
136  ; Interrupts 16-19 are handled by NVIC register PRI4.
137  ; Interrupt 19 is controlled by bits 31:29 of PRI4.
138  ; set NVIC interrupt 19 to priority 2
139              LDR R1, =NVIC_PRI4
140              LDR R2, [R1]
141              AND R2, R2, #0x00FFFFFF ; clear interrupt 19 priority
142              ORR R2, R2, #0x40000000 ; set interrupt 19 priority to 2
143              STR R2, [R1]
144  ; NVIC has to be enabled
```

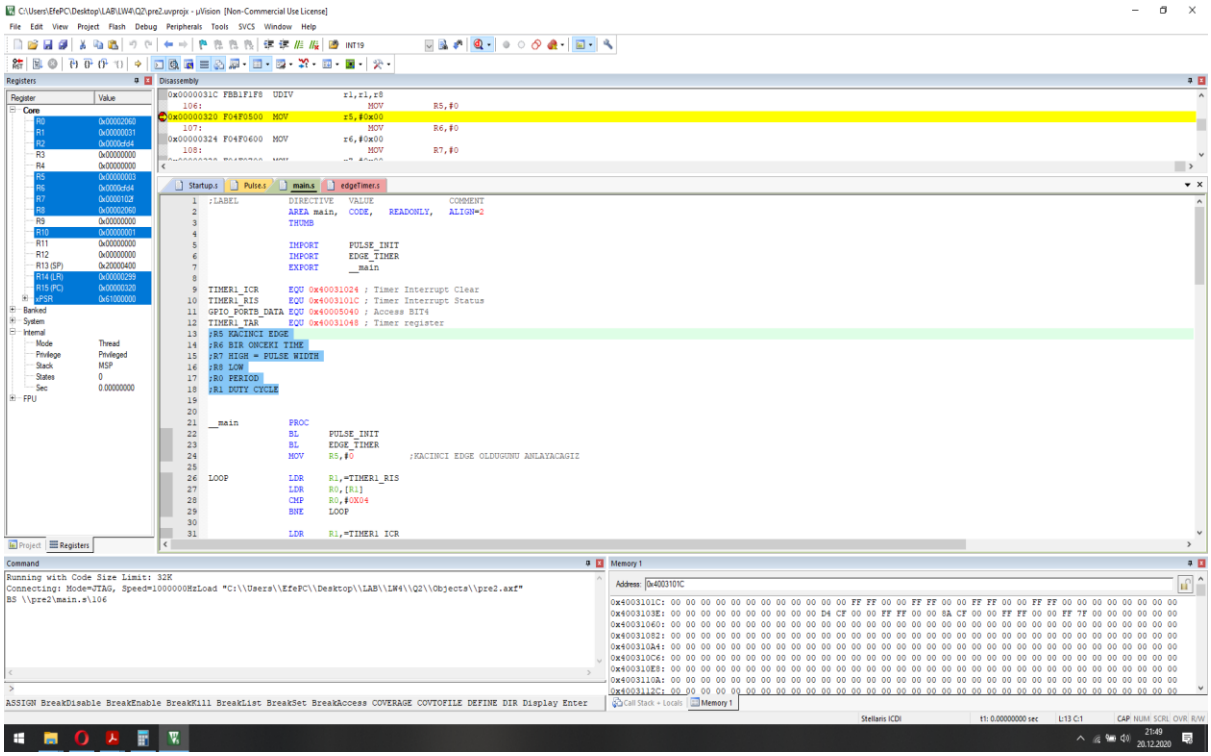
```
145   ; Interrupts 0-31 are handled by NVIC register EN0
146   ; Interrupt 19 is controlled by bit 19
147   ; enable interrupt 19 in NVIC
148               LDR R1, =NVIC_EN0
149               MOVT R2, #0x08 ; set bit 19 to enable interrupt 19
150               STR R2, [R1]
151   ; Enable timer
152               LDR R1, =TIMER0_CTL
153               LDR R2, [R1]
154               ORR R2, R2, #0x03 ; set bit0 to enable
155               STR R2, [R1] ; and bit 1 to stall on debug
156               BX LR ; return
157               ENDP
158               END
```

Q2) R0=PERIOD, R1=DUTY CYCLE, R7=PULSE WIDTH, WE SHOULD DIVIDE R0 AND R7 BY 16 TO GET THE CORRECT RESULT IN NANOSECONDS BECAUSE PULSE IS 1MHZ. (SHIFT 1 BYTE RIGHT)

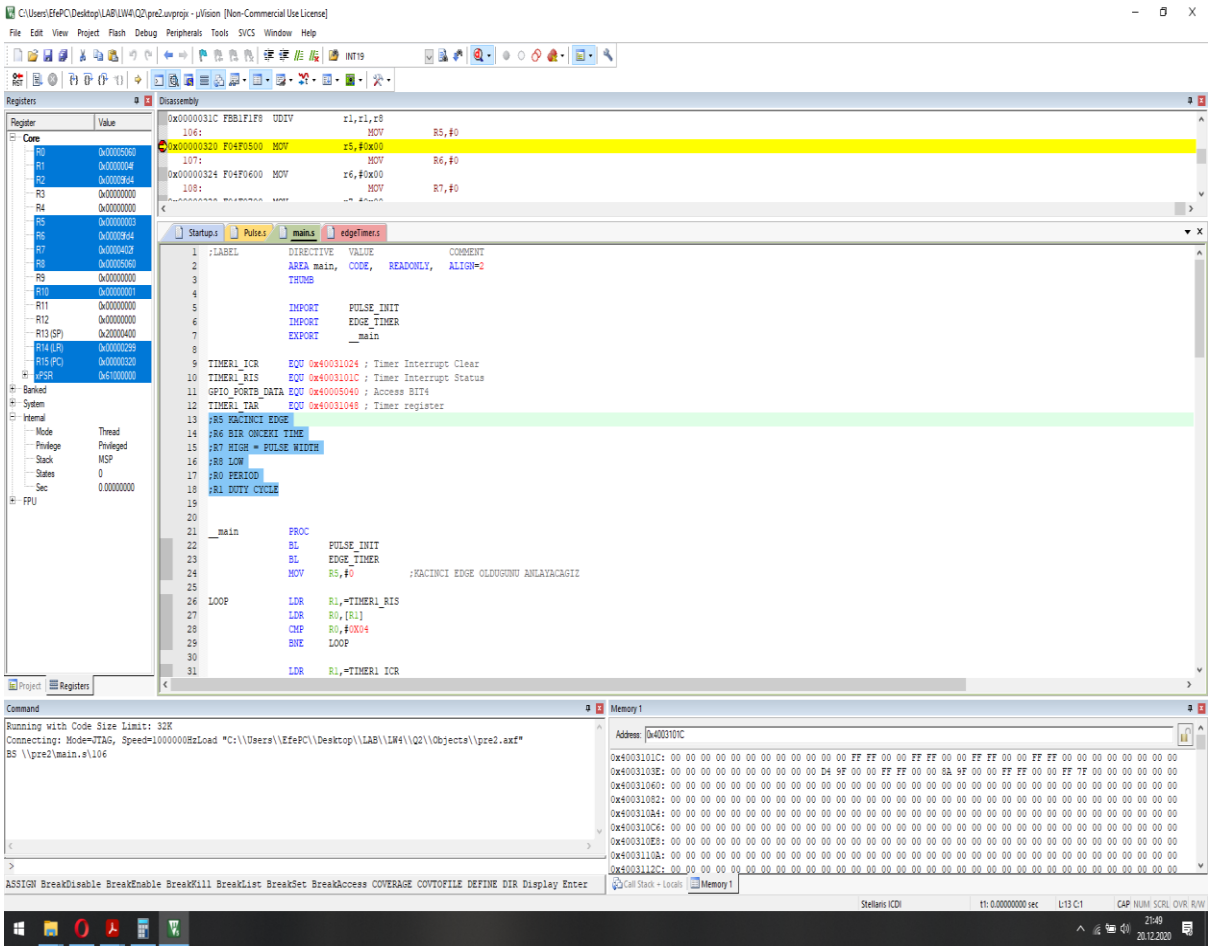
TEST1 %20 DUTY CYCLE, LOW = 0X00000100, HIGH=0X00000040



TEST2 %50 DUTY CYCLE, LOW = HIGH = 0X00000100



TEST3 %80 DUTY CYCLE, LOW = 0X00000100, HIGH = 0X00000400



```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                 MAIN OF THE Q2                  ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;LABEL          DIRECTIVE   VALUE               COMMENT
                AREA main,  CODE,    READONLY,  ALIGN=2
                THUMB

                IMPORT      PULSE_INIT
                IMPORT      EDGE_TIMER
                EXPORT      __main

TIMER1_ICR      EQU 0x40031024 ; Timer Interrupt Clear
TIMER1_RIS      EQU 0x4003101C ; Timer Interrupt Status
GPIO_PORTB_DATA EQU 0x40005040 ; Access BIT4
TIMER1_TAR      EQU 0x40031048 ; Timer register
;R5 KACINCI EDGE
;R6 BIR ONCEKI TIME
;R7 HIGH = PULSE WIDTH
;R8 LOW
;R0 PERIOD
;R1 DUTY CYCLE


__main          PROC
                BL      PULSE_INIT
                BL      EDGE_TIMER
                MOV     R5,#0           ;KACINCI EDGE OLDUGUNU ANLAYACAGIZ

LOOP            LDR     R1,=TIMER1_RIS
                LDR     R0,[R1]
                CMP     R0,#0X04
                BNE     LOOP

                LDR     R1,=TIMER1_ICR
                LDR     R0,[R1]
                ORR     R0,#0X04
                STR     R0,[R1]

                LDR     R1,=GPIO_PORTB_DATA
                LDR     R0,[R1]
                LSR     R0,#4

                ADD     R5,#1
                CMP     R5,#1
                BEQ     FIRST
                CMP     R5,#2
                BEQ     SECOND
                B       THIRD

FIRST           LDR     R1,=TIMER1_TAR
                LDR     R6,[R1]
                B       FINISH



SECOND          LDR     R1,=TIMER1_TAR
                LDR     R2,[R1]
                CMP     R0,#0
                BEQ     POSEDGE
                B       NEGEDGE



THIRD           LDR     R1,=TIMER1_TAR
                LDR     R2,[R1]
                CMP     R0,#0
                BEQ     POSEDGE
                B       NEGEDGE
```

```
 73
 74  POSEDGE         SUB     R7,R6,R2
 75                  CMP     R6,R2
 76                  CPYHI   R6,R2
 77                  BHI     EXIT
 78                  SUB     R7,R2,R6
 79                  LDR     R0,=0X10000 ;FULL CYCLE
 80                  ADD     R7,R0
 81                  CPY     R6,R2
 82                  B       EXIT
 83
 84
 85
 86  NEGEDGE         SUB     R8,R6,R2
 87                  CMP     R6,R2
 88                  CPYHI   R6,R2
 89                  BHI     EXIT
 90                  SUB     R8,R2,R6
 91                  LDR     R0,=0X10000 ;FULL CYCLE
 92                  ADD     R8,R0
 93                  CPY     R6,R2
 94                  B       EXIT
 95
 96  CALC            ADD     R0,R7,R8        ;PERIOD
 97                  ;MOV        R3,#625
 98                  ;MOV        R4,#10
 99                  ;MUL        R2,R3
100                  ;UDIV   R2,R4           ;PERIOD IN NANOSECONDS
101
102                  ;CPY        R0,R7           ;PULSE WIDTH
103                  ;MUL        R0,R3
104                  ;UDIV   R0,R4           ;PULSE WIDTH IN NANOSECONDS
105                  ;R7                     ;PULSE WIDTH
106
107                  MOV     R1,#100
108                  MUL     R1,R7
109                  ADD     R8,R7
110                  UDIV    R1,R8           ;DUTY CYCLE
111                  MOV     R5,#0
112                  MOV     R6,#0
113                  MOV     R7,#0
114                  MOV     R8,#0
115                  B       FINISH
116
117
118  EXIT            CMP     R5,#0X03
119                  BEQ     CALC
120  FINISH          LDR     R0,=TIMER1_ICR
121                  ORR     R1,#0X04            ;CLEAR BIT2, BECAUSE CAPTURE MODE
122                  STR     R1,[R0]
123
124                  B       LOOP
125
126                  ENDP
127                  END
128
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                  PULSE OF THE Q2                      ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; Pulse.s
; Routine for creating a pulse train using interrupts
; This uses Channel 0, and a 1MHz Timer Clock (_TAPR = 15 )
; Uses Timer1A to READ EDGES on PB4

;Nested Vector Interrupt Controller registers
NVIC_EN0_INT19      EQU 0x00080000 ; Interrupt 19 enable
NVIC_EN0            EQU 0xE000E100 ; IRQ 0 to 31 Set Enable Register
NVIC_PRI4           EQU 0xE000E410 ; IRQ 16 to 19 Priority Register

; 16/32 Timer Registers
TIMER0_CFG          EQU 0x40030000
TIMER0_TAMR         EQU 0x40030004
TIMER0_CTL          EQU 0x4003000C
TIMER0_IMR          EQU 0x40030018
TIMER0_RIS          EQU 0x4003001C ; Timer Interrupt Status
TIMER0_ICR          EQU 0x40030024 ; Timer Interrupt Clear
TIMER0_TAILR        EQU 0x40030028 ; Timer interval
TIMER0_TAPR         EQU 0x40030038
TIMER0_TAR          EQU 0x40030048 ; Timer register

;GPIO Registers
GPIO_PORTF_DATA     EQU 0x40025010 ; Access BIT2
GPIO_PORTF_DIR      EQU 0x40025400 ; Port Direction
GPIO_PORTF_AFSEL    EQU 0x40025420 ; Alt Function enable
GPIO_PORTF_DEN      EQU 0x4002551C ; Digital Enable
GPIO_PORTF_AMSEL    EQU 0x40025528 ; Analog enable
GPIO_PORTF_PCTL     EQU 0x4002552C ; Alternate Functions

;System Registers
SYSCTL_RCGCGPIO     EQU 0x400FE608 ; GPIO Gate Control
SYSCTL_RCGCTIMER    EQU 0x400FE604 ; GPTM Gate Control

;--------------------------------------------------
LOW                 EQU 0x00000100
HIGH                EQU 0x00000040
;--------------------------------------------------

            AREA    routines, CODE, READONLY
            THUMB
            EXPORT  My_Timer0A_Handler
            EXPORT  PULSE_INIT

;--------------------------------------------------
My_Timer0A_Handler  PROC
                    ;...
                    ADD     R10,#1
                    CMP     R10,#1
                    BEQ     HIGHX

LOWX                LDR R0,=GPIO_PORTF_DATA
                    LDR R1,[R0]
                    MOV R1,#0
                    STR R1,[R0]
                    LDR R1, =TIMER0_TAILR ; initialize match clocks
                    LDR R2, =LOW
                    STR R2, [R1]
                    MOV R10,#0
                    B   EXIT

HIGHX               LDR R0,=GPIO_PORTF_DATA
                    LDR R1,[R0]
                    MOV R1,#4
                    STR R1,[R0]
                    LDR R1, =TIMER0_TAILR ; initialize match clocks
                    LDR R2, =HIGH
                    STR R2, [R1]
                    B   EXIT
```

```
 73
 74  EXIT                LDR R0,=TIMER0_ICR
 75                      ORR R1,#0X01
 76                      STR R1,[R0]
 77                      BX  LR
 78                      ENDP
 79  ;-------------------------------------------------
 80
 81  PULSE_INIT  PROC
 82              LDR R1, =SYSCTL_RCGCGPIO ; start GPIO clock
 83              LDR R0, [R1]
 84              ORR R0, R0, #0x20 ; set bit 5 for port F
 85              STR R0, [R1]
 86              NOP ; allow clock to settle
 87              NOP
 88              NOP
 89              LDR R1, =GPIO_PORTF_DIR ; set direction of PF2
 90              LDR R0, [R1]
 91              ORR R0, R0, #0x04 ; set bit2 for output
 92              STR R0, [R1]
 93              LDR R1, =GPIO_PORTF_AFSEL ; regular port function
 94              LDR R0, [R1]
 95              BIC R0, R0, #0x04
 96              STR R0, [R1]
 97              LDR R1, =GPIO_PORTF_PCTL ; no alternate function
 98              LDR R0, [R1]
 99              BIC R0, R0, #0x00000F00
100              STR R0, [R1]
101              LDR R1, =GPIO_PORTF_AMSEL ; disable analog
102              MOV R0, #0
103              STR R0, [R1]
104              LDR R1, =GPIO_PORTF_DEN ; enable port digital
105              LDR R0, [R1]
106              ORR R0, R0, #0x04
107              STR R0, [R1]
108
109              LDR R1, =SYSCTL_RCGCTIMER ; Start Timer0
110              LDR R2, [R1]
111              ORR R2, R2, #0x01
112              STR R2, [R1]
113              NOP ; allow clock to settle
114              NOP
115              NOP
116              LDR R1, =TIMER0_CTL ; disable timer during setup LDR R2, [R1]
117              BIC R2, R2, #0x01
118              STR R2, [R1]
119              LDR R1, =TIMER0_CFG ; set 16 bit mode
120              MOV R2, #0x04
121              STR R2, [R1]
122              LDR R1, =TIMER0_TAMR
123              MOV R2, #0x02 ; set to periodic, count down
124              STR R2, [R1]
125              LDR R1, =TIMER0_TAILR ; initialize match clocks
126              LDR R2, =LOW
127              STR R2, [R1]
128              LDR R1, =TIMER0_TAPR
129              MOV R2, #15 ; divide clock by 16 to
130              STR R2, [R1] ; get 1us clocks
131              LDR R1, =TIMER0_IMR ; enable timeout interrupt
132              MOV R2, #0x01
133              STR R2, [R1]
134  ; Configure interrupt priorities
135  ; Timer0A is interrupt #19.
136  ; Interrupts 16-19 are handled by NVIC register PRI4.
137  ; Interrupt 19 is controlled by bits 31:29 of PRI4.
138  ; set NVIC interrupt 19 to priority 2
139              LDR R1, =NVIC_PRI4
140              LDR R2, [R1]
141              AND R2, R2, #0x00FFFFFF ; clear interrupt 19 priority
142              ORR R2, R2, #0x40000000 ; set interrupt 19 priority to 2
143              STR R2, [R1]
144  ; NVIC has to be enabled
```

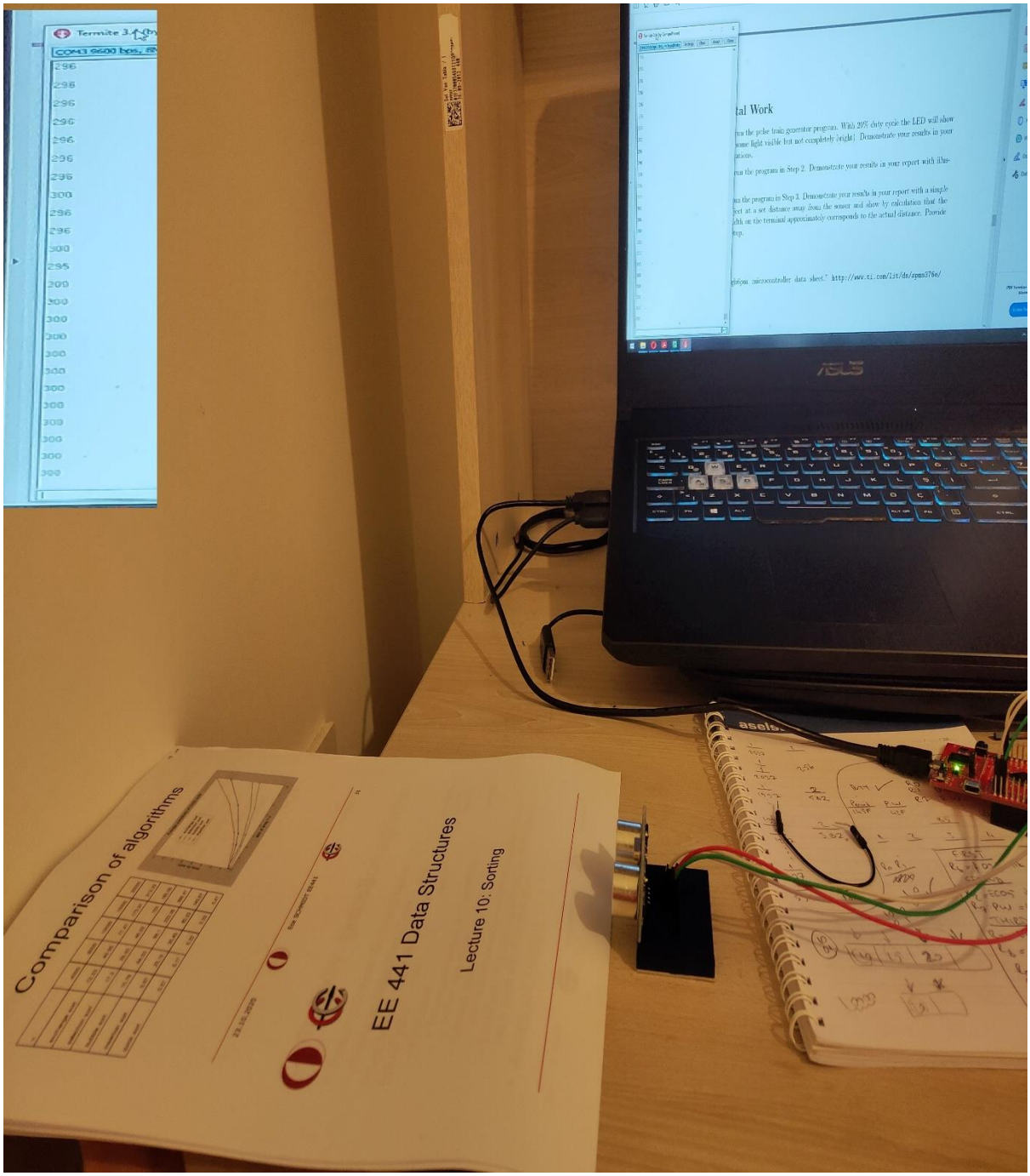
```
145   ; Interrupts 0-31 are handled by NVIC register EN0
146   ; Interrupt 19 is controlled by bit 19
147   ; enable interrupt 19 in NVIC
148               LDR R1, =NVIC_EN0
149               MOVT R2, #0x08 ; set bit 19 to enable interrupt 19
150               STR R2, [R1]
151   ; Enable timer
152               LDR R1, =TIMER0_CTL
153               LDR R2, [R1]
154               ORR R2, R2, #0x03 ; set bit0 to enable
155               STR R2, [R1] ; and bit 1 to stall on debug
156               BX LR ; return
157               ENDP
158               END
```

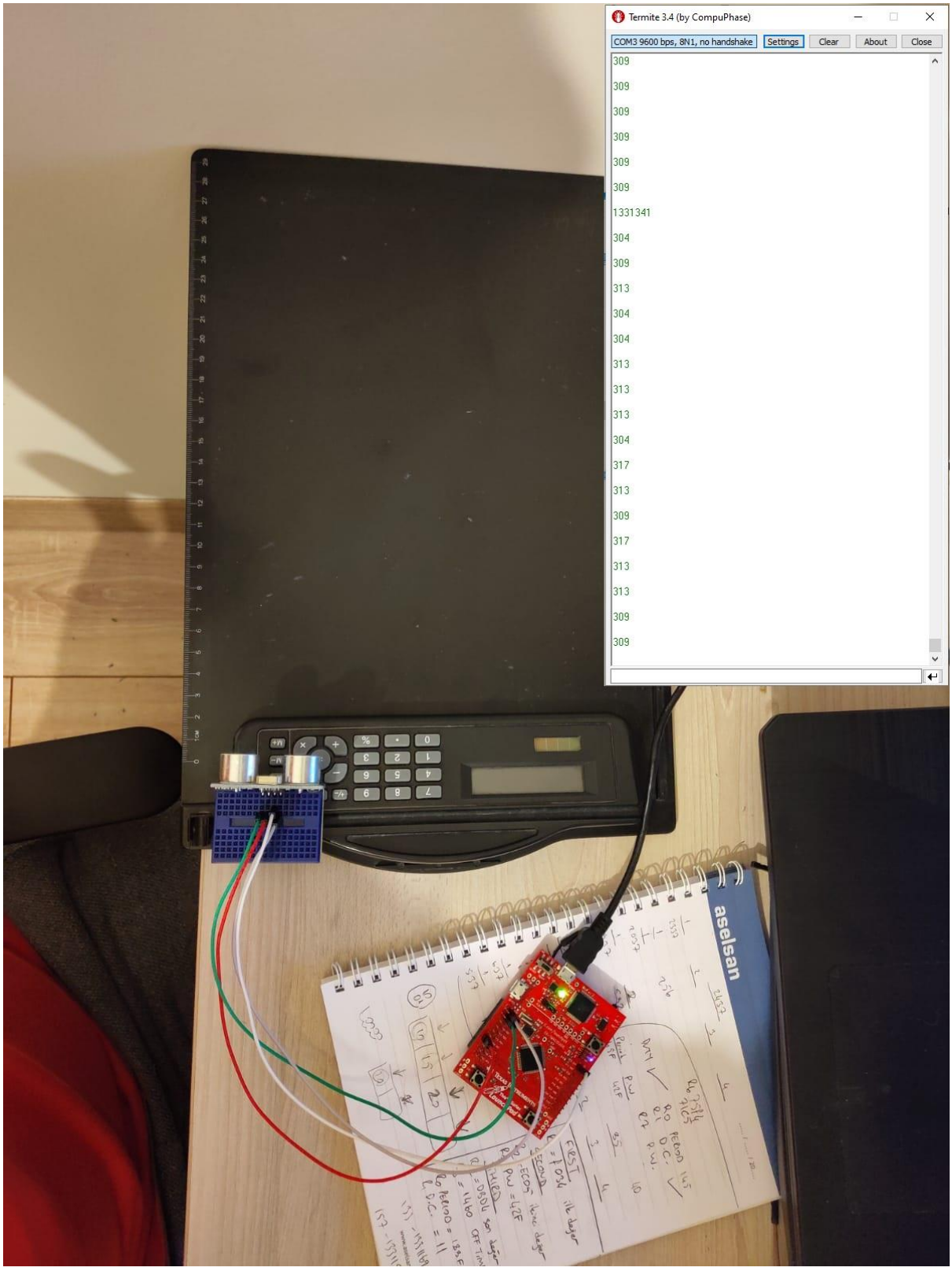
```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                  EDGE TIMER OF THE Q2        ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; edgeTimer.s
; Uses Timer1A to COUNT EDGES on PB4

;Nested Vector Interrupt Controller registers
NVIC_EN0_INT19      EQU 0x00080000 ; Interrupt 19 enable
NVIC_EN0            EQU 0xE000E100 ; IRQ 0 to 31 Set Enable Register
NVIC_PRI5           EQU 0xE000E414 ; IRQ 16 to 19 Priority Register

; 16/32 Timer Registers
TIMER1_CFG          EQU 0x40031000
TIMER1_TAMR         EQU 0x40031004
TIMER1_CTL          EQU 0x4003100C
TIMER1_IMR          EQU 0x40031018
TIMER1_RIS          EQU 0x4003101C ; Timer Interrupt Status
TIMER1_ICR          EQU 0x40031024 ; Timer Interrupt Clear
TIMER1_TAILR        EQU 0x40031028 ; Timer interval
TIMER1_TAPR         EQU 0x40031038
TIMER1_TAR          EQU 0x40031048 ; Timer register

;GPIO Registers
GPIO_PORTB_DATA     EQU 0x40005040 ; Access BIT4
GPIO_PORTB_DIR      EQU 0x40005400 ; Port Direction
GPIO_PORTB_AFSEL    EQU 0x40005420 ; Alt Function enable
GPIO_PORTB_DEN      EQU 0x4000551C ; Digital Enable
GPIO_PORTB_AMSEL    EQU 0x40005528 ; Analog enable
GPIO_PORTB_PCTL     EQU 0x4000552C ; Alternate Functions
GPIO_PORTB_PDR      EQU 0X40005514 ;PULL DOWN REGISTER


;System Registers
SYSCTL_RCGCGPIO     EQU 0x400FE608 ; GPIO Gate Control
SYSCTL_RCGCTIMER    EQU 0x400FE604 ; GPTM Gate Control

;----------------------------------------------------
LOW                 EQU 0x00000100
HIGH                EQU 0x00000040
;----------------------------------------------------

            AREA    routines, CODE, READONLY
            THUMB
            ;EXPORT     My_Timer1A_Handler
            EXPORT  EDGE_TIMER

;----------------------------------------------------
;My_Timer1A_Handler PROC






;EXIT               LDR R0,=TIMER1_ICR
;                   ORR R1,#0X04            ;CLEAR BIT2, BECAUSE CAPTURE MODE
;                   STR R1,[R0]
;                   BX  LR
;                   ENDP
;----------------------------------------------------

EDGE_TIMER  PROC
            LDR R1, =SYSCTL_RCGCGPIO ; start GPIO clock
            LDR R0, [R1]
            ORR R0, R0, #0x02 ; set bit 1 for port B
            STR R0, [R1]
            NOP ; allow clock to settle
            NOP
            NOP
            LDR R1, =GPIO_PORTB_DIR ; set direction of PB4
            LDR R0, [R1]
```

```
 73              BIC R0, R0, #0x10 ; SET BIT4 AS INPUT
 74              STR R0, [R1]
 75              LDR R1, =GPIO_PORTB_AFSEL ; ALTERNATE PB4
 76              LDR R0, [R1]
 77              ORR R0, R0, #0x10 ;PB4
 78              STR R0, [R1]
 79              LDR R1, =GPIO_PORTB_PCTL ; ALTERNATE PB4
 80              LDR R0, [R1]
 81              ORR R0, R0, #0x00070000 ;PB4
 82              STR R0, [R1]
 83              LDR R1, =GPIO_PORTB_AMSEL ; disable analog
 84              MOV R0, #0
 85              STR R0, [R1]
 86              LDR R1, =GPIO_PORTB_DEN ; enable port digital
 87              LDR R0, [R1]
 88              ORR R0, R0, #0x10
 89              STR R0, [R1]
 90              LDR R1, =GPIO_PORTB_PDR ;PULL DOWN PB4
 91              LDR R0,[R1]
 92              ORR R0,#0X10
 93              STR R0,[R1]
 94
 95              LDR R1, =SYSCTL_RCGCTIMER ; Start Timer1
 96              LDR R2, [R1]
 97              ORR R2, R2, #0x02
 98              STR R2, [R1]
 99              NOP ; allow clock to settle
100              NOP
101              NOP
102              LDR R1, =TIMER1_CTL ; disable timer during setup
103              LDR R2, [R1]
104              BIC R2, R2, #0x01
105              STR R2, [R1]
106              LDR R1, =TIMER1_CFG ; set 16 bit mode
107              MOV R2, #0x04
108              STR R2, [R1]
109              LDR R1, =TIMER1_TAMR
110              MOV R2, #0x07 ; CAPTURE, EDGE TIME, COUNT DOWN 00111
111              STR R2, [R1]
112              LDR R1, =TIMER1_TAILR ; initialize match clocks
113              LDR R2, =0XFFFF
114              STR R2, [R1]
115              ;LDR R1, =TIMER1_TAPR
116              ;MOV R2, #15 ; divide clock by 16 to
117              ;STR R2, [R1] ; get 1us clocks
118              ;LDR R1, =TIMER1_IMR ; enable timeout interrupt
119              ;MOV R2, #0x01
120              ;STR R2, [R1]
121  ; Configure interrupt priorities
122  ; Timer0A is interrupt #19.
123  ; Interrupts 16-19 are handled by NVIC register PRI4.
124  ; Interrupt 19 is controlled by bits 31:29 of PRI4.
125  ; set NVIC interrupt 19 to priority 2
126              ;LDR R1, =NVIC_PRI5
127              ;LDR R2, [R1]
128              ;AND R2, R2, #0xFFFF00FF ; clear interrupt 21 priority
129              ;ORR R2, R2, #0x00004000 ; set interrupt 21 priority to 2
130              ;STR R2, [R1]
131  ; NVIC has to be enabled
132  ; Interrupts 0-31 are handled by NVIC register EN0
133  ; Interrupt 19 is controlled by bit 19
134  ; enable interrupt 19 in NVIC
135              ;LDR R1, =NVIC_EN0
136              ;MOVT R2, #0x20 ; set bit 21 to enable interrupt 21
137              ;STR R2, [R1]
138  ; Enable timer
139              LDR R1, =TIMER1_CTL
140              LDR R2, [R1]
141              ORR R2, R2, #0x0F ; set bit0 to enable
142              STR R2, [R1] ; and bit 1 to stall on debug, SET BIT 3:2 TO DETECT BOTH EDGES
143              BX LR ; return
144              ENDP
```

```
145                 END
146
```

Q3) A4 PAPER LENGTH IS 29.7CM, THE RULER IS 30CM. I ALSO TESTED IT USING MY PHONE, IT IS 15CM AND THE RESULT WAS CORRECT BUT I COULDN'T TAKE THE PHOTO BECAUSE I USED MY PHONE AS A RULER.




Termite 3.4 (by CompuPhase)
COM3 9600 bps, 8N1, no handshake
Settings
Clear
About
Close
309
309
309
309
309
309
1331341
304
309
313
304
304
313
313
313
304
317
313
309
317
313
313
309
309
aselsan

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                  MAIN OF THE Q3                  ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;LABEL          DIRECTIVE   VALUE               COMMENT
                AREA main,  CODE,    READONLY,   ALIGN=2
                THUMB

                IMPORT      OutStr
                IMPORT      CONVRT
                IMPORT      PULSE_INIT
                IMPORT      EDGE_TIMER
                EXPORT      __main

TIMER1_ICR      EQU 0x40031024 ; Timer Interrupt Clear
TIMER1_RIS      EQU 0x4003101C ; Timer Interrupt Status
GPIO_PORTB_DATA EQU 0x40005040 ; Access BIT4
TIMER1_TAR      EQU 0x40031048 ; Timer register
TIMER0_CTL      EQU 0x4003000C

NUM             EQU 0X20000500
;R5 KACINCI EDGE
;R6 BIR ONCEKI TIME
;R7 HIGH = PULSE WIDTH
;R8 LOW
;R0 PERIOD
;R1 DUTY CYCLE

MSG         DCB         "DISTANCE IN MM: "
            ;DCB            0x0D        ; Carriage return is like new line
            DCB         0x04        ; it is like EOF or \0


__main          PROC
                BL      EDGE_TIMER
                BL      PULSE_INIT
                MOV     R5,#0           ;KACINCI EDGE OLDUGUNU ANLAYACAGIZ

LOOP            LDR     R1,=TIMER1_RIS
                LDR     R0,[R1]
                CMP     R0,#0X04
                BNE     LOOP


                LDR     R1,=TIMER1_ICR
                LDR     R0,[R1]
                ORR     R0,#0X04
                STR     R0,[R1]

                LDR     R1,=GPIO_PORTB_DATA
                LDR     R0,[R1]
                ;LSR        R0,#4

                ;ADD        R0,R0,R5
                ;CMP        R0,#1
                ;BNE        LOOP

                ADD     R5,#1
                CMP     R5,#1
                BEQ     FIRST
                B       SECOND

FIRST           LDR     R1,=TIMER1_TAR
                LDR     R6,[R1]
                LDR R1, =TIMER0_CTL ; disable timer during setup LDR R2, [R1]
                BIC R2, R2, #0x01
                STR R2, [R1]
                B       FINISH



SECOND          LDR     R1,=TIMER1_TAR
```

```
 73                  LDR     R2,[R1]
 74                  B       POSEDGE
 75
 76  POSEDGE         SUB     R7,R6,R2
 77                  ;CMP        R6,R2
 78                  ;CPYHI  R6,R2
 79                  ;BHI        EXIT
 80                  ;SUB        R7,R2,R6
 81                  ;LDR        R0,=0X10000 ;FULL CYCLE
 82                  ;ADD        R7,R0
 83                  CPY     R6,R2
 84                  B       EXIT
 85
 86
 87  CALC            LDR     R5,=34
 88                  LDR     R6,=3200
 89                  MUL     R7,R5
 90                  UDIV    R7,R6   ;MM DISTANCE
 91                  CPY     R4,R7
 92                  ;LDR        R5,=MSG
 93                  ;BL     OutStr
 94                  LDR     R5,=NUM
 95                  BL      CONVRT
 96                  LDR     R5,=NUM
 97                  BL      OutStr
 98
 99                  MOV     R5,#0
100                  MOV     R6,#0
101                  MOV     R7,#0
102                  LDR     R1, =TIMER0_CTL
103                  LDR     R2, [R1]
104                  ORR     R2, R2, #0x03 ; set bit0 to enable
105                  STR     R2, [R1] ; and bit 1 to stall on debug
106                  B       FINISH
107
108
109 EXIT             CMP     R5,#0X02
110                  BEQ     CALC
111 FINISH           LDR     R0,=TIMER1_ICR
112                  ORR     R1,#0X04            ;CLEAR BIT2, BECAUSE CAPTURE MODE
113                  STR     R1,[R0]
114
115                  B       LOOP
116
117                  ENDP
118                  END
119
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                 PULSE OF THE Q3                  ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; Pulse.s
; Routine for creating a pulse train using interrupts
; This uses Channel 0, and a 1MHz Timer Clock (_TAPR = 15 )
; Uses Timer1A to READ EDGES on PB4

;Nested Vector Interrupt Controller registers
NVIC_EN0_INT19      EQU 0x00080000 ; Interrupt 19 enable
NVIC_EN0            EQU 0xE000E100 ; IRQ 0 to 31 Set Enable Register
NVIC_PRI4           EQU 0xE000E410 ; IRQ 16 to 19 Priority Register

; 16/32 Timer Registers
TIMER0_CFG          EQU 0x40030000
TIMER0_TAMR         EQU 0x40030004
TIMER0_CTL          EQU 0x4003000C
TIMER0_IMR          EQU 0x40030018
TIMER0_RIS          EQU 0x4003001C ; Timer Interrupt Status
TIMER0_ICR          EQU 0x40030024 ; Timer Interrupt Clear
TIMER0_TAILR        EQU 0x40030028 ; Timer interval
TIMER0_TAPR         EQU 0x40030038
TIMER0_TAR          EQU 0x40030048 ; Timer register

;GPIO Registers
GPIO_PORTF_DATA     EQU 0x40025010 ; Access BIT2
GPIO_PORTF_DIR      EQU 0x40025400 ; Port Direction
GPIO_PORTF_AFSEL    EQU 0x40025420 ; Alt Function enable
GPIO_PORTF_DEN      EQU 0x4002551C ; Digital Enable
GPIO_PORTF_AMSEL    EQU 0x40025528 ; Analog enable
GPIO_PORTF_PCTL     EQU 0x4002552C ; Alternate Functions

;System Registers
SYSCTL_RCGCGPIO     EQU 0x400FE608 ; GPIO Gate Control
SYSCTL_RCGCTIMER    EQU 0x400FE604 ; GPTM Gate Control

;-------------------------------------------------
LOW                 EQU 0xFFFFFFFF
HIGH                EQU 0x0000000F
;-------------------------------------------------

            AREA    routines, CODE, READONLY
            THUMB
            IMPORT  DELAY100
            EXPORT  My_Timer0A_Handler
            EXPORT  PULSE_INIT

;-------------------------------------------------
My_Timer0A_Handler  PROC
                    ;...
                    ADD     R10,#1
                    CMP     R10,#1
                    BEQ     HIGHX


LOWX                LDR R0,=GPIO_PORTF_DATA
                    LDR R1,[R0]
                    MOV R1,#0
                    STR R1,[R0]
                    LDR R1, =TIMER0_TAILR ; initialize match clocks
                    LDR R2, =LOW
                    STR R2, [R1]
                    MOV R10,#0
                    B   EXIT

HIGHX               LDR R0,=GPIO_PORTF_DATA
                    LDR R1,[R0]
                    MOV R1,#4
                    STR R1,[R0]
                    LDR R1, =TIMER0_TAILR ; initialize match clocks
                    LDR R2, =HIGH
```

```
                        STR R2, [R1]
                        B   EXIT

EXIT                    LDR R0,=TIMER0_ICR
                        ORR R1,#0X01
                        STR R1,[R0]
                        BX  LR
                        ENDP
;-------------------------------------------------

PULSE_INIT  PROC
            LDR R1, =SYSCTL_RCGCGPIO ; start GPIO clock
            LDR R0, [R1]
            ORR R0, R0, #0x20 ; set bit 5 for port F
            STR R0, [R1]
            NOP ; allow clock to settle
            NOP
            NOP
            LDR R1, =GPIO_PORTF_DIR ; set direction of PF2
            LDR R0, [R1]
            ORR R0, R0, #0x04 ; set bit2 for output
            STR R0, [R1]
            LDR R1, =GPIO_PORTF_AFSEL ; regular port function
            LDR R0, [R1]
            BIC R0, R0, #0x04
            STR R0, [R1]
            LDR R1, =GPIO_PORTF_PCTL ; no alternate function
            LDR R0, [R1]
            BIC R0, R0, #0x00000F00
            STR R0, [R1]
            LDR R1, =GPIO_PORTF_AMSEL ; disable analog
            MOV R0, #0
            STR R0, [R1]
            LDR R1, =GPIO_PORTF_DEN ; enable port digital
            LDR R0, [R1]
            ORR R0, R0, #0x04
            STR R0, [R1]

            PUSH {LR}
            BL  DELAY100
            POP {LR}
            LDR R1, =SYSCTL_RCGCTIMER ; Start Timer0
            LDR R2, [R1]
            ORR R2, R2, #0x01
            STR R2, [R1]
            NOP ; allow clock to settle
            NOP
            NOP
            LDR R1, =TIMER0_CTL ; disable timer during setup LDR R2, [R1]
            BIC R2, R2, #0x01
            STR R2, [R1]
            LDR R1, =TIMER0_CFG ; set 16 bit mode
            MOV R2, #0x04
            STR R2, [R1]
            LDR R1, =TIMER0_TAMR
            MOV R2, #0x02 ; set to periodic, count down
            STR R2, [R1]
            LDR R1, =TIMER0_TAILR ; initialize match clocks
            LDR R2, =LOW
            STR R2, [R1]
            LDR R1, =TIMER0_TAPR
            MOV R2, #15 ; divide clock by 16 to
            STR R2, [R1] ; get 1us clocks
            LDR R1, =TIMER0_IMR ; enable timeout interrupt
            MOV R2, #0x01
            STR R2, [R1]
; Configure interrupt priorities
; Timer0A is interrupt #19.
; Interrupts 16-19 are handled by NVIC register PRI4.
; Interrupt 19 is controlled by bits 31:29 of PRI4.
; set NVIC interrupt 19 to priority 2
            LDR R1, =NVIC_PRI4
```

```
145             LDR R2, [R1]
146             AND R2, R2, #0x00FFFFFF ; clear interrupt 19 priority
147             ORR R2, R2, #0x40000000 ; set interrupt 19 priority to 2
148             STR R2, [R1]
149 ; NVIC has to be enabled
150 ; Interrupts 0-31 are handled by NVIC register EN0
151 ; Interrupt 19 is controlled by bit 19
152 ; enable interrupt 19 in NVIC
153             LDR R1, =NVIC_EN0
154             MOVT R2, #0x08 ; set bit 19 to enable interrupt 19
155             STR R2, [R1]
156 ; Enable timer
157             LDR R1, =TIMER0_CTL
158             LDR R2, [R1]
159             ORR R2, R2, #0x03 ; set bit0 to enable
160             STR R2, [R1] ; and bit 1 to stall on debug
161             BX LR ; return
162             ENDP
163             END
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                 ECHO OF THE Q3                  ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; edgeTimer.s
; Uses Timer1A to COUNT EDGES on PB4

;Nested Vector Interrupt Controller registers
NVIC_EN0_INT19      EQU 0x00080000 ; Interrupt 19 enable
NVIC_EN0            EQU 0xE000E100 ; IRQ 0 to 31 Set Enable Register
NVIC_PRI5           EQU 0xE000E414 ; IRQ 16 to 19 Priority Register

; 16/32 Timer Registers
TIMER1_CFG          EQU 0x40031000
TIMER1_TAMR         EQU 0x40031004
TIMER1_CTL          EQU 0x4003100C
TIMER1_IMR          EQU 0x40031018
TIMER1_RIS          EQU 0x4003101C ; Timer Interrupt Status
TIMER1_ICR          EQU 0x40031024 ; Timer Interrupt Clear
TIMER1_TAILR        EQU 0x40031028 ; Timer interval
TIMER1_TAPR         EQU 0x40031038
TIMER1_TAR          EQU 0x40031048 ; Timer register

;GPIO Registers
GPIO_PORTB_DATA     EQU 0x40005040 ; Access BIT4
GPIO_PORTB_DIR      EQU 0x40005400 ; Port Direction
GPIO_PORTB_AFSEL    EQU 0x40005420 ; Alt Function enable
GPIO_PORTB_DEN      EQU 0x4000551C ; Digital Enable
GPIO_PORTB_AMSEL    EQU 0x40005528 ; Analog enable
GPIO_PORTB_PCTL     EQU 0x4000552C ; Alternate Functions
GPIO_PORTB_PDR      EQU 0X40005514 ;PULL DOWN REGISTER


;System Registers
SYSCTL_RCGCGPIO     EQU 0x400FE608 ; GPIO Gate Control
SYSCTL_RCGCTIMER    EQU 0x400FE604 ; GPTM Gate Control

;---------------------------------------------------
LOW                 EQU 0x00000100
HIGH                EQU 0x00000040
;---------------------------------------------------

            AREA    routines, CODE, READONLY
            THUMB
            ;EXPORT     My_Timer1A_Handler
            EXPORT  EDGE_TIMER

;---------------------------------------------------
;My_Timer1A_Handler PROC






;EXIT               LDR R0,=TIMER1_ICR
;                   ORR R1,#0X04            ;CLEAR BIT2, BECAUSE CAPTURE MODE
;                   STR R1,[R0]
;                   BX  LR
;                   ENDP
;---------------------------------------------------

EDGE_TIMER  PROC
            LDR R1, =SYSCTL_RCGCGPIO ; start GPIO clock
            LDR R0, [R1]
            ORR R0, R0, #0x02 ; set bit 1 for port B
            STR R0, [R1]
            NOP ; allow clock to settle
            NOP
            NOP
            LDR R1, =GPIO_PORTB_DIR ; set direction of PB4
            LDR R0, [R1]
```

```
 73              BIC R0, R0, #0x10 ; SET BIT4 AS INPUT
 74              STR R0, [R1]
 75              LDR R1, =GPIO_PORTB_AFSEL ; ALTERNATE PB4
 76              LDR R0, [R1]
 77              ORR R0, R0, #0x10 ;PB4
 78              STR R0, [R1]
 79              LDR R1, =GPIO_PORTB_PCTL ; ALTERNATE PB4
 80              LDR R0, [R1]
 81              ORR R0, R0, #0x00070000 ;PB4
 82              STR R0, [R1]
 83              LDR R1, =GPIO_PORTB_AMSEL ; disable analog
 84              MOV R0, #0
 85              STR R0, [R1]
 86              LDR R1, =GPIO_PORTB_DEN ; enable port digital
 87              LDR R0, [R1]
 88              ORR R0, R0, #0x10
 89              STR R0, [R1]
 90              LDR R1, =GPIO_PORTB_PDR ;PULL DOWN PB4
 91              LDR R0,[R1]
 92              ORR R0,#0X10
 93              STR R0,[R1]
 94
 95              LDR R1, =SYSCTL_RCGCTIMER ; Start Timer1
 96              LDR R2, [R1]
 97              ORR R2, R2, #0x02
 98              STR R2, [R1]
 99              NOP ; allow clock to settle
100              NOP
101              NOP
102              LDR R1, =TIMER1_CTL ; disable timer during setup
103              LDR R2, [R1]
104              BIC R2, R2, #0x01
105              STR R2, [R1]
106              LDR R1, =TIMER1_CFG ; set 16 bit mode
107              MOV R2, #0x04
108              STR R2, [R1]
109              LDR R1, =TIMER1_TAMR
110              MOV R2, #0x07 ; CAPTURE, EDGE TIME, COUNT DOWN 00111
111              STR R2, [R1]
112              LDR R1, =TIMER1_TAILR ; initialize match clocks
113              LDR R2, =0XFFFF
114              STR R2, [R1]
115              LDR R1, =TIMER1_TAPR
116              MOV R2, #15 ; divide clock by 16 to
117              STR R2, [R1] ; get 1us clocks
118              ;LDR R1, =TIMER1_IMR ; enable timeout interrupt
119              ;MOV R2, #0x01
120              ;STR R2, [R1]
121  ; Configure interrupt priorities
122  ; Timer0A is interrupt #19.
123  ; Interrupts 16-19 are handled by NVIC register PRI4.
124  ; Interrupt 19 is controlled by bits 31:29 of PRI4.
125  ; set NVIC interrupt 19 to priority 2
126              ;LDR R1, =NVIC_PRI5
127              ;LDR R2, [R1]
128              ;AND R2, R2, #0xFFFF00FF ; clear interrupt 21 priority
129              ;ORR R2, R2, #0x00004000 ; set interrupt 21 priority to 2
130              ;STR R2, [R1]
131  ; NVIC has to be enabled
132  ; Interrupts 0-31 are handled by NVIC register EN0
133  ; Interrupt 19 is controlled by bit 19
134  ; enable interrupt 19 in NVIC
135              ;LDR R1, =NVIC_EN0
136              ;MOVT R2, #0x20 ; set bit 21 to enable interrupt 21
137              ;STR R2, [R1]
138  ; Enable timer
139              LDR R1, =TIMER1_CTL
140              LDR R2, [R1]
141              ORR R2, R2, #0x0F ; set bit0 to enable
142              STR R2, [R1] ; and bit 1 to stall on debug, SET BIT 3:2 TO DETECT BOTH EDGES
143              BX LR ; return
144              ENDP
```

```
145                 END
146
```