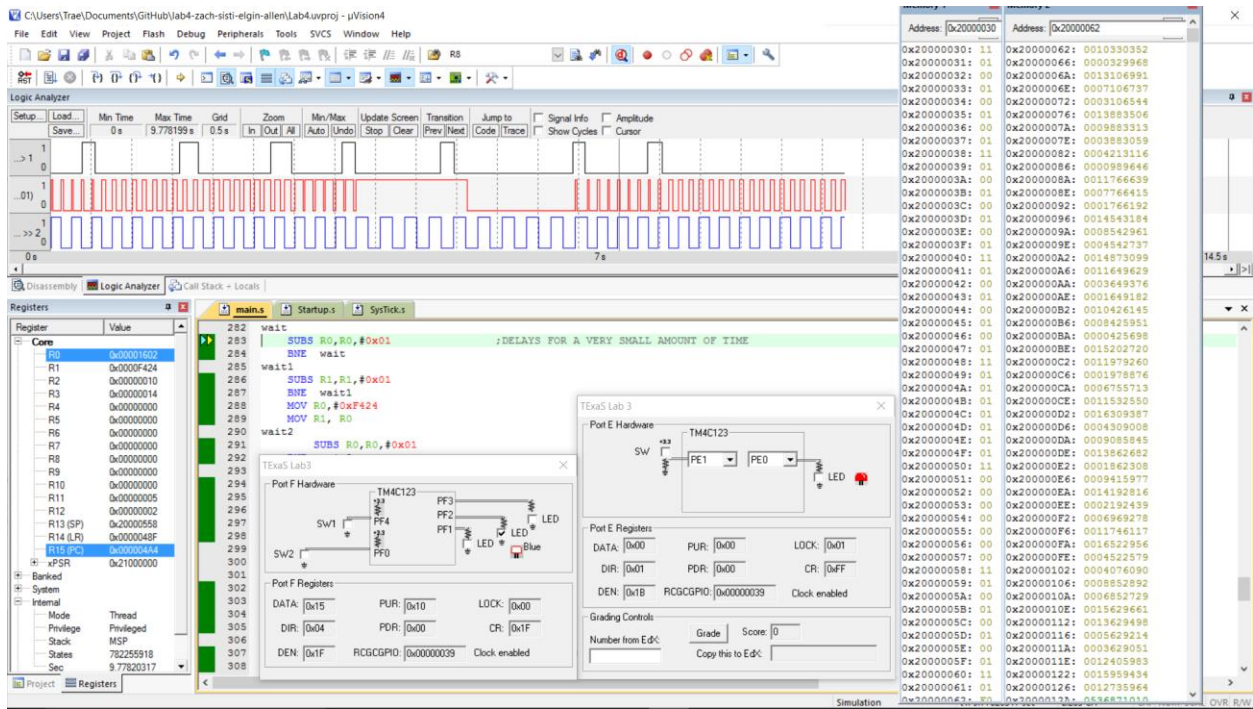


Zach Sisti / Elgin Allen

## 1. Screenshot of system



## 2. Assembly Code

```
GPIO_PORTE_DATA_R EQU 0x400243FC
GPIO_PORTE_DIR_R EQU 0x40024400
GPIO_PORTE_AFSEL_R EQU 0x40024420
GPIO_PORTE_DEN_R EQU 0x4002451C

GPIO_PORTF_DATA_R EQU 0x400253FC
GPIO_PORTF_DIR_R EQU 0x40025400
GPIO_PORTF_AFSEL_R EQU 0x40025420
GPIO_PORTF_PUR_R EQU 0x40025510
GPIO_PORTF_DEN_R EQU 0x4002551C
GPIO_PORTF_CR_R EQU 0x40025524
GPIO_PORTF_PCTL_R EQU 0x4002552C
GPIO_PORTF_LOCK_R EQU 0x40025520
SYSCCTL_RCGCGPIO_R EQU 0x400FE608

NVIC_ST_CTRL_R EQU 0xE000E010
NVIC_ST_RELOAD_R EQU 0xE000E014
NVIC_ST_CURRENT_R EQU 0xE000E018
; RAM Area
    AREA DATA, ALIGN=2
; -UUU-Declare and allocate space for your Buffers
; and any variables (like pointers and counters) here
DataBuffer SPACE 50 ;each entry is 1 byte/8B
TimeBuffer SPACE 200 ; each entry is 4 bytes/32B
DataPt SPACE 4 ; Addresses are 32B
TimePt SPACE 4 ;Addresses are 32B
NEntries FILL 1,0,1
; ROM Area
    IMPORT TExaS_Init
    IMPORT SysTick_Init
; -UUU-Import routine(s) from other assembly files (like SysTick.s) here
    AREA |.text|, CODE, READONLY, ALIGN=2
    THUMB
    EXPORT Start

Start
; TExaS_Init sets bus clock at 80 MHz
BL TExaS_Init ; voltmeter, scope on PD3
CPSIE I ; TExaS voltmeter, scope runs on interrupts
    LDR R1, =SYSCCTL_RCGCGPIO_R ;1) activate clock for Port F and E
    LDR R0, [R1]
    ORR R0, R0, #0x30 ;set bit 4,5 to turn on clock
    STR R0, [R1]
    NOP
    NOP ;allow time for clock to finish
    LDR R1, =GPIO_PORTF_LOCK_R
    LDR R0, =0x4C4F434B
    STR R0, [R1]
    LDR R1, =GPIO_PORTF_LOCK_R
    LDR R0, [R1]
    LDR R1, =GPIO_PORTF_PCTL_R
    STR R0, [R1]
    LDR R1, =GPIO_PORTF_CR_R
    MOV R0, #0x7FFF
    STR R0, [R1]
    LDR R1, =GPIO_PORTF_DIR_R ;5) set direction register
    LDR R0, [R1]
    MOV R0, #0x04 ;set all pins as inputs
    STR R0, [R1]
    LDR R1, =GPIO_PORTF_DEN_R ;7) enable Port F digital port
    LDR R0, [R1]
    MOV R0, #0x1F ;1 means enable digital I/O
    STR R0, [R1]
    LDR R1, =GPIO_PORTF_AFSEL_R
    LDR R1, [R1]
```

```

AND R0, R0, #0x00
STR R0, [R1]
LDR R1, =GPIO_PORTF_PUR_R ;pull-up resistors for PF4
MOV R0, #0x10 ;enable weak pull-up on PF4
STR R0, [R1]

LDR R1, =GPIO_PORTE_DIR_R ;5) set direction register
LDR R0, [R1]
MOV R0, #0x1 ;set pins 3-1 input, pin 0 as output
STR R0, [R1]
LDR R1, =GPIO_PORTE_DEN_R ;7) enable Port E digital port
LDR R0, [R1]
MOV R0, #0x1B ;1 means enable digital I/O
STR R0, [R1]
LDR R1, =GPIO_PORTE_AFSEL_R
LDR R1, [R1]
AND R0, R0, #0x00
STR R0, [R1]
MOV R10, #0
MOV R11, #5 ;PERIOD
MOV R12, #1 ;DUTY CYCLE
BL Debug_Init

loop
LDR R1, =GPIO_PORTF_DATA_R
LDR R2, [R1]
AND R2, R2, #0x04
EOR R2, R2, #0X04
STR R2, [R1]

LDR R1, =GPIO_PORTF_DATA_R
LDR R2, [R1]
AND R2, R2, #0x10
AND R0, #0 ;CLEAR R0
CMP R2, R0
BEQ breathe
CMP R9, R0
BGT noduty
BL toggleLightOn

noduty
ADD R0, R12, R0 ;R0=DUTY
BL delay
BL delay
SUB R0, R11, R12 ;R0=PERIOD-DUTY
AND R1, R1, #0
CMP R0, R1
BEQ fullcycle
BL toggleLightOff

fullcycle
BL delay
BL delay
LDR R1, =GPIO_PORTE_DATA_R ;CHECKING SWITCH
LDR R0, [R1]
AND R2, R0, #0x02
MOV R0, #0 ;CREATING CONSTANT FOR COMPARING
CMP R2, R0 ;CHECKING SWITCH STATE
BEQ notPressed
AND R9, R9, #0
AND R6, R6, #0 ;CLEAR R6
ADD R6, R6, #1 ;INDICATES SWITCH HAS BEEN PRESSED
ADD R8, R8, #1
;CMP R8, #0
;BEQ DontCap
;BL Debug_Capture
;SUB R8, R8, #1

;DontCap
B loop
notPressed

```

```

        AND R8, R8, #0
        AND R5, R5, #0                                ;CLEAR R5
        CMP R6, R5                                    ;CHECK IF SWITCH HAS BEEN PRESSED
        BEQ loop
        MOV R10, #7
        CMP R11, R12                                ;CHECKS IF DUTY=PERIOD
        BEQ reset
        ADD R12, R12, #1                            ;IF NOT INCREASE DUTY (20%)
        AND R6, R6, #0
        B loop
reset
        AND R12, R12, #0                            ;RESETS DUTY TO 0%
        AND R6, R6, #0
        AND R9, R9, #0
        ADD R9, R9, #1
        B loop
breathe
        PUSH {R12, R7, R0, R11}
        MOV R11, #10
        MOV R7, #0                                    ;SET COUNTER TO 0
        MOV R10, #0                                    ;UP/DOWN INDICATOR (0 UP, 1 DOWN)
        MOV R1, #0                                    ;0 CONSTANT FOR COMPARING
breathe1
        BL toggleLightOn
        ADD R0, R12, R0                                ;R0=DUTY
        BL delay
        BL toggleLightOff
        SUB R0, R11, R12    ;R0=PERIOD-DUTY
        BL delay
        CMP R10, R1                                    ;increment or decrement (up/down indicator)
        BEQ increment                                ;if R10==0
        ADD R12, R12, #-1    ;decrement
        B skip
increment
        ADD R12, R12, #1
skip
        CMP R12, R11                                ;are we at top?
        BEQ top
        CMP R12, R1                                ;are we at bottom?
        BEQ bottom
        B fin
top
        MOV R10, #1                                    ;set up/down indicator to down direction (1)
        B fin
bottom
        MOV R10, #0                                    ;set up/down indicator to up direction (0)
        B fin
fin
        LDR R1, =GPIO_PORTF_DATA_R                    ;check if button is still pressed
        LDR R2, [R1]
        AND R2, R2, #0x10
        AND R0, #0                                    ;CLEAR R0
        CMP R2, R0
        BEQ breathe1
        POP {R12, R7, R0, R11}
        B loop

toggleLightOn
        PUSH {R0, R1, R2, LR}
        LDR R1, =GPIO_PORTE_DATA_R
        LDR R0, [R1]
        ORR R0, R0, #0x01                                ;SETTING PE0 TO 1
        STR R0, [R1]                                    ;STORES NEW DATAREG
        CMP R8, #1
        BNE NoCNo
        BL Debug_Capture
        MOV R8, #-250
NoCNo
        CMP R10, #0

```

```

        BEQ nocap
        BL Debug_Capture
        SUB R10, R10, #1
nocap
        POP {R0, R1, R2, LR}
        BX LR

toggleLightOff
        PUSH {R0, R1, R2, LR}

        LDR R1, =GPIO_PORTC_DATA_R
        LDR R0, [R1]
        AND R0, R0, #0xFE
        STR R0, [R1]
        CMP R8, #0
        BNE nocap1
        CMP R10, #0
        BEQ nocap1
        BL Debug_Capture
        SUB R10, R10, #1
nocap1
        POP {R0, R1, R2, LR}
        BX LR

delay
        PUSH {LR, R1, R0, R2}
        ;MOV R2, #40
        ;MUL R0, R0, R2

while
        MOV R1, #0
        BL sub1
        ADD R0, #-1
        CMP R0, R1
        BGT while
        POP {LR, R1, R0, R2}
        BX LR
        ;CLEAR R1
        ;DECREMENTENTS WHILE LOOP INDICATOR
        ;REPEATS EACH 1/5 OF THE DUTY CYCLE

sub1
        PUSH {R0,R1}
        MOV R0,#0xF424
        FOUND WITH TRIAL AND ERROR
        MOV R1, R0
        wait
        SUBS R0,R0,#0x01
        BNE wait
        wait1
        SUBS R1,R1,#0x01
        BNE wait1
        MOV R0,#0xF424
        MOV R1, R0
        wait2
        SUBS R0,R0,#0x01
        BNE wait2
        wait3
        SUBS R1,R1,#0x01
        BNE wait3
        MOV R1, R0
        POP {R0, R1}
        BX LR
        ;DELAYS FOR 1/80th OF PERIOD
        ;SETS R0 TO ARBIRTRARY CONSTANT WHICH WE

Debug_Init
        PUSH {R0, R1, R2, R3, R8, R9, LR, R10}
        LDR R0, =DataBuffer
        LDR R1, =TimeBuffer
        MOV R3, #50
        ClearLoop
        ;SETS R1 TO R0
        ;DELAYS FOR A VERY SMALL AMOUNT OF TIME
        ;Loop counter

```

```

        CMP R3, #0                                ;Check if at end of array
        BEQ ExitClear

        MOV R8, #0xFF                            ;set current Data array value to FF
        STR R8, [R0]

        MOV R9, #0xFFFFFFFF                      ;set current Time array value to 8 Fs
        STR R9, [R1]

        ADD R0, R0, #1                            ;Move to next value in Data array
        ADD R1, R1, #4                            ;Move to next value in Time Array
        SUB R3, R3, #1
        BL ClearLoop
ExitClear

        LDR R0, =DataBuffer
        LDR R1, =DataPt
        STR R0, [R1]                            ;Set pointer at start of array

        LDR R0, =TimeBuffer
        LDR R1, =TimePt
        STR R0, [R1]                            ;Set pointer at start of array

        BL SysTick_Init
        POP {R0, R1, R2, R3, R8, R9, LR, R10}
        BX LR

Debug_Capture
        PUSH {R0, R1, R2, R3, R4, R5}
        LDR R0, =NEntries
        LDR R1, [R0]
        CMP R1, #50
        BEQ FIN

        LDR R0, =GPIO_PORTE_DATA_R
        LDR R1, [R0]
        AND R1, R1, #0x03

        LDR R0, =NVIC_ST_CURRENT_R
        LDR R4, [R0]

        LSL R2, R1, #3
        AND R2, R2, #0x10
        AND R1, R1, #0x01    ;prep data to be dumped
        ADD R1, R1, R2
        LDR R0, =DataPt
        LDR R3, [R0]        ;store data
        STRB R1, [R3]

        ADD R3, R3, #1        ;increment DataPt
        STR R3, [R0]

        LDR R0, =TimePt
        LDR R3, [R0]        ;store Time
        STR R4, [R3]

        ADD R3, R3, #4        ;increment TimePt
        STR R3, [R0]

        LDR R0, =NEntries
        LDR R1, [R0]
        ADD R1, R1, #1
        STR R1, [R0]

FIN
        POP {R0, R1, R2, R3, R4, R5}
        BX LR

ALIGN    ; make sure the end of this section is aligned
END      ; end of file

```

### 3. Estimation of Execution time

Execution time: 975 ns

Average time between executions: 0.062503 seconds

Intrusiveness: .0015599251%

Paste from the saved File (50 entries)

count: 50

										12.5 <- Time per tick		
										Differences	Time(ms)	
:020000042000DA	F0A09D00	F0080500	2FFFC700	Adjust-endian Data		10330352						
:0E006200F0A09D00F00805002FFFC700B17050	B1706C00	F0662F00	72D8D300	B1CE9600	009DA0F0	10330352						
:100070006C00F0662F0072D8D300B1CE96003340EA	33403B00	7C494000	CE190F00	6F88B300	000508F0	329968	10000384	125.0048	<-time from press to release			
:100080003B007C494000CE190F006F88B3008F817D	8F817600	30F31A00	50E9DD00	F15A8200	00C7FF2F	13106991	4000193	50.0024125	<- first 6 time differences			
:10009000760030F31A0050E9DD00F15A8200115168	11514500	0BF2E200	5DC2B100	60AF3700	006C70B1	7106737	6000254	75.003175				
:1000A0004500BF2E2005DC2B10060AF37001E2ACE	1E2A1900	21179F00	DF918000	E27E0600	002F66F0	3106544	4000193	50.0024125				
:1000B000190021179F00DF918000E27E0600A0F961	A0F9E700	FCC9B600	FC311E00	81156700	00D3D872	13883506	6000254	75.003175				
:1000C000E700FCC9B600FC311E008115670006F987	06F9AF00	8BDCF800	10C04100	95A38A00	0096CEB1	9883313	4000193	50.0024125				
:1000D000AF008BDCF80010C0410095A38A001A879E	1A87D300	A46A1C00	29AD8F00	B090D800	003B4033	3883059	6000254	75.003175				
:1000E000D300A46A1C0029AD8F00B090D8003774EB	37742100	BE576A00	453BB300	CC1EFC00	0040497C	4213116	16447159					
:1000F0002100BE576A00453BB300CC1EFC005302F2	53024500	3A323E00	9C158700	79906800	000F19CE	989646	3223470	40.293375	<-time from press to release			
:1001000045003A323E009C158700799068005D7D7D	5D7DEE00	3AF8CF00	1EE55500	FB5F3700	00B38B6F	11766639	6000223	75.0027875	<- next 6 time differences			
:10011000EE003AF8CF001EE55500FB5F3700DF4CDC	DF4CBD00	8A85F300	DC55C200		0076818F	7766415	4000224	50.0028				
:0A012000BD008A85F300DC55C20023					001AF330	1766192	6000223	75.0027875				
:00000001FF					00DDE950	14543184	4000224	50.0028				
					00825AF1	8542961	6000223	75.0027875				
					00455111	4542737	4000224	50.0028				
					00E2F20B	14873099	6446854					
					00B1C25D	11649629	3223470	40.293375	<-time from press to release			
					0037AF60	3649376	8000253	100.0031625	<- next 6 time differences			
					00192A1E	1649182	2000194	25.002425				
					009F1721	10426145	8000253	100.0031625				
					008091DF	8425951	2000194	25.002425				
					00067EE2	425698	8000253	100.0031625				
					00E7F9A0	15202720	2000194	25.002425				
					00B6C9FC	11979260	3223460					
					001E31FC	1978876	10000384	125.0048	<-time from press to release			
					00671581	6755713	12000379	150.0047375	<- next 6 time differences			
					00AFF906	11532550	12000379	150.0047375				
					00F8DC8B	16309387	12000379	150.0047375				
					0041C010	4309008	12000379	150.0047375				
					008AA395	9085845	12000379	150.0047375				
					00D3871A	13862682	12000379	150.0047375				
					001C6AA4	1862308	12000374					
					008FAD29	9415977	9223547	115.2943375	<-time from press to release			
					00D890B0	14192816	12000377	150.0047125	<- next 6 time differences			
					00217437	2192439	12000377	150.0047125				
					006A57BE	6969278	12000377	150.0047125				
					00B33B45	11746117	12000377	150.0047125				
					00FC1ECC	16522956	12000377	150.0047125				
					00450253	4522579	12000377	150.0047125				
					003E323A	4076090	446489					
					0087159C	8852892	12000414	150.005175	<-time from press to release			
					00689079	6852729	2000163	25.0020375	<- next 6 time differences			
					00EE7D5D	15629661	8000284	100.00355				
					00CFF83A	13629498	2000163	25.0020375				
					0055E51E	5629214	8000284	100.00355				
					00375FFB	3629051	2000163	25.0020375				
					00BD4CDF	12405983	8000284	100.00355				
					00F3858A	15959434	13223765					
					00C255DC	12735964	3223470	40.293375				