Timers

6.

Graphical user interface, text, application

Description automatically generated

7. student id to hex: 103809048 -> 0x6300018

8.

8.1 because MOV need 24 bits to work

8.2 because need to add 2 integers at a time using orr

8.3

Mov r2,$100000

Orr r2,$003800

Orr r2,$000009

Ok4.asm

Code of ok4.asm

BASE = $3F000000 ; Use $3F000000 for 2B, 3B, 3B+

GPIO\_OFFSET = $200000

mov r0,BASE

orr r0,GPIO\_OFFSET ;Base address of GPIO

mov r1,#1

lsl r1,#24; GPIO18

str r1,[r0,#4] ;enable output

mov r1,#1

lsl r1,#18

loop$:

mov r2,#3

timerloop4:

str r1,[r0,#28] ;Turn on LED

;new timer

TIMER\_OFFSET = $3000

;TIMER\_MICROSECONDS = 524288 ; $0080000 ;0.524288 s

mov r3,BASE

orr r3,TIMER\_OFFSET ;store base address of timer (r3)

mov r4,$70000

orr r4,$0A100

orr r4,$00020 ;TIMER\_MICROSECONDS = 500,000

;store delay (r4)

ldrd r6,r7,[r3,#4]

mov r5,r6 ;store starttime (r5)(=currenttime (r6))

timerloop:

ldrd r6,r7,[r3,#4] ;read currenttime (r6)

sub r8,r6,r5 ;remainingtime (8)= currenttime (r6) - starttime (r5)

cmp r8,r4 ;compare remainingtime (r8), delay (r4)

bls timerloop ;loop if LE (reaminingtime <= delay)

str r1,[r0,#40] ;turn off LED

;re-use timer

ldrd r6,r7,[r3,#4]

mov r5,r6 ;store starttime (r5)(=currenttime (r6))

timerloop2:

ldrd r6,r7,[r3,#4] ;read currenttime (r6)

sub r8,r6,r5 ;remainingtime (8)= currenttime (r6) - starttime (r5)

cmp r8,r4 ;compare remainingtime (r8), delay (r4)

bls timerloop2 ;loop if LE (reaminingtime <= delay)

sub r2,#1

cmp r2,#0

bne timerloop4

;NEW TIMER

str r1,[r0,#40]

mov r9,$2D0000

orr r9,$00C600

orr r9,$0000C0

ldrd r6,r7,[r3,#4]

mov r5,r6

timerloop3:

ldrd r6,r7,[r3,#4]

sub r8,r6,r5

cmp r8,r9

bls timerloop3

b loop$