Name: Ta Quang Tung

Class: Computer Systems (Thursday afternoon)

Title: Lab 8

6 – Error message: Immediate value cannot be encoded.

7 – 104222 = 0x1971E

8.1 – MOV only works with numbers with at least 24 bits set to 0 because it combines both the operation and the operand into a 32-bit word, the former occupying a total of 24 bits for the opcode and the ROR.

8.2 – For numbers that do not satisfy this condition, MOV and be used in conjunction with ORR to construct the full number part by part.

8.3 – Code to load 0x1971E into register:

mov r2,$19000

orr r2,$00710

orr r2,$0000E

15 – Approach: Modify the existing infinite loop so that it will loop three times instead of looping forever. After the loop has finished, run a 3 second timer. Wrap all of this in a new infinite loop.

16, 17, 18 – Code:

format binary as ‘img’

BASE=$3F000000

GPIO\_OFFSET=$200000

TIMER\_OFFSET=$3000

mov r0,BASE

orr r0,GPIO\_OFFSET

mov r3,BASE

orr r3,TIMER\_OFFSET

mov r1,#1

lsl r1,#24

str r1,[r0,#4]

mov r1,#1

lsl r1,#18

loop1:

mov r2,#3

loop2:

str r1,[r0,#28]

mov r4,$70000

orr r4,$0A100

orr r4,$00020

ldrd r6,r7,[r3,#4]

mov r5,r6

timerloop:

ldrd r6,r7,[r3,#4]

sub r8,r6,r5

cmp r8,r4

bls timerloop

str r1,[r0,#40]

ldrd r6,r7,[r3,#4]

mov r5,r6

timerloop2:

ldrd r6,r7,[r3,#4]

sub r8,r6,r5

cmp r8,r4

bls timerloop2

sub r2,#1

cmp r2,#0

bne loop2

mov r4,$2D0000

orr r4,$00C600

orr r4,$0000C0

ldrd r6,r7,[r3,#4]

mov r5,r6

timerloop3:

ldrd r6,r7,[r3,#4]

sub r8,r6,r5

cmp r8,r4

bls timerloop3

b loop1