CE2107 Lab1 Assignment Sheet (to be submitted to NTULearn before next lab)

Name: Dayna Chia Lab Group: SE1 Date: 10/09/22

1. Section 7.4. Write one C statement to set bit 7 and 6 of P1SEL0 register, keeping the rest of the bits in the register unchanged.

P1->SEL0 |= 0xC0

1. Section 7.4. Write C statement(s) to extract bit 4 and 3 of variable ‘x’ and right align these two bits. Masked off all other bits in variable ‘x’. e.g. if ‘x’ has a value 1101 0111b initially, it should have a value of 0000 0010b after executing the C statement.

x = x>>3

x &= 0x18

1. Section 7.4. Why do we need to declare the P1IN register, which is of the registers that contain the status of the processor input pin logic as a ‘volatile’ keyword qualifier?

Status of the P1IN logic may be altered by external events. Declaring it as ‘volatile’ ensures that changes to the register value outside of program context are recorded and not discarded when code optimization is enabled.

1. Section 7.5. Why do we use SDIV instead of UDIV when calculating the Distance D? Or does it really matter whether SDIV or UDIV is used for this case?

SDIV performs signed integer division, while UDIV performs unsigned integer division.

It does not matter in this case since distance is always positive.

1. Section 7.5. What is saved into the LR register when the calling routine calls “BL Convert”? What command is used to return from the sub-routine to the calling routine?

The address of the instruction right after “BL Convert” is stored into the LR register.

“BX LR” is used to return from the sub-routine to the calling routine.

1. Section 7.5. If a function has 4 input parameters, which registers does the calling routine used to pass these parameters to the function according to AAPCS?

R0, R1, R2 and R3

1. Section 7.6. What is loaded into R1 by the instruction “ldr r1, [pc, #0x2e4]”?

0x40004C0A

It is the content stored at the address stored in the PC with an offset of #0x2e4.

1. Section 7.7. The Memory Section “MAIN” correspond to the On-Chip Flash Memory in MSP432. How much on-chip flash memory is available for future code development? Hint: Check the map file.

0x0003f804 bytes

1. Section 7.7. Which software section are code allocated to by default? Which file consumes the largest code size in this project? Hint: check the map file.

Code is allocated to .text by default.

The “system\_msp432p401r.obj” file consumes the largest code size.

1. Section 7.7. From the map file, what is the starting address of Port2\_Init()? Compare with the address you see in the Disassembly Window, are they the same? If not, why?

From the map file, the starting address is 0x43b. From the address in the Disassembly Window, the starting address is 0x43a.

They are not the same.

Port2\_Init() is a Thumb function that runs in Thumb state, so any pointer to this function must have the least significant bit set (0x43b = 0100 0011 1011b vs 0x43a = 0100 0011 1010b).