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

Name: Bryan Lu We Zhern Lab Group: SE4 Date: 26/8/2022

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 & 0b00011000) >> 3;

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?

So that the compiler do not skip, remove or modify any of the related code during compilation due to code optimization of the compiler.

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?

It does not matter if we use SDIV or UDIV as both yield the same results given the same input in the input range.

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 Program Counter (PC) registers’s value, which is the subsequent instruction’s address before branching, is stored in the LR register when calling routine Convert, then the BX LR instruction load the original instruction address back to the PC.

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-R3

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

R1 is loaded with the memory address that would be used to store the value of R0 later.

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.

0x3F804 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.

The codes are allocated to .text section.

system\_msp432p401r.obj takes the largest file size of 0x32C bytes.

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, Port2\_Init() starts from 0x0000043B, while from the disassembly window, it starts from 0x0000043A.

They are not the same due to Port2\_Init() being a separate function in the C file, so the “entry point” of Port2\_Init() in the map file and in the disassembly window might differ.