Laboratory 3 tasks

Alexander Lundqvist

**For compiling and installing on chipKit**

make clean

make

make install TTYDEV=/dev/tty.usbserial-A503WFGA

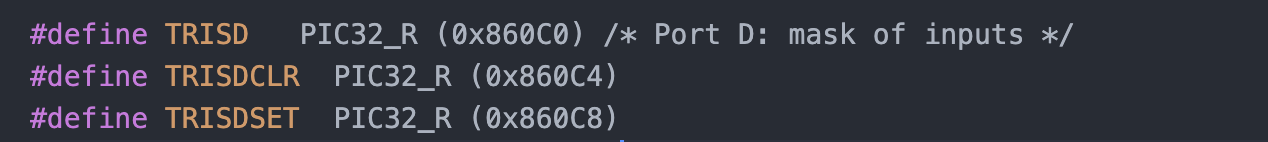
**Task 1**

*• Test pressing BTN3 and BTN2 at the same time. What happens? Why?*

Both digit 3 and two gets updated value on screen. This is because the program actually executes all the if statements. We have bruteforce statements that says, for instance, that if we press 3&2 (011), 2 of the if-statements have this comparison.

*• Three device-registers for input/output control are TRISE, TRISESET, and TRISECLR.*

*Their functions are related. How? What are the differences?*



**12.2.8Set, Clear, and Invert I/O Port Registers**

TRISE:

TRISESET:

TRISECLR:

*• In the generated assembly code, in which MIPS register will the return values from*

*functions getbtns and getsw be placed in. You should be able to answer this question*

*without debugging the generated assembly code.*

*• In this exercise, we explained which bits that should be used in Port D and Port E. How can*

*you find this information in the PIC32 and ChipKIT manuals? Be prepared to demonstrate*

*how to find this information in the manuals.*

*Advice: check the lecture slides from lecture 5 for ideas.*

**Task 2**

*• When the time-out event-flag is a "1", how does your code reset it to "0"?*

*• What would happen if the time-out event-flag was not reset to "0" by your code? Why?*

*• Which device-register (or registers) must be written to define the time between time-out*

*events? Describe the function of that register (or of those registers).*

*• If you press BTN3 quickly, does the time update reliably? Why, or why not? If not, would*

*that be easy to change? If so, how?*

**Task 3**

*• When the time-out event-flag is a "1", how does your code reset it to "0"?*

*• What would happen if the time-out event-flag was not reset to "0" by your code? Why?*

*• From which part of the code is the function user\_isr called? Why is it called from there?*

*• Why are registers saved before the call to user\_isr? Why are only some registers saved?*

asd

*• Which device-register (or registers), and which processor-register (or registers) must be*

*written to enable interrupts from the timer? Describe the functions of the relevant registers.*

ASD