# LAB 2 REPORT

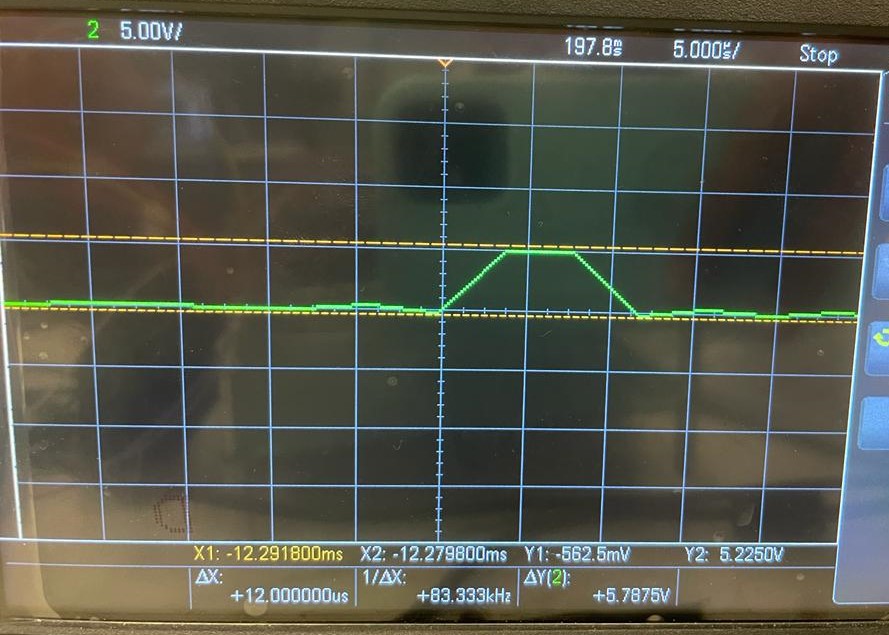
## PART 1

**Using the instruction set summary tables (available in the programmer's guide or instruction set documents), calculate how long the ISR takes to execute once, assuming a clock frequency of 11.0592 MHz. You will likely have some conditional jumps in your ISR code, so make sure to calculate both the longest and shortest time it takes the ISR to execute.**

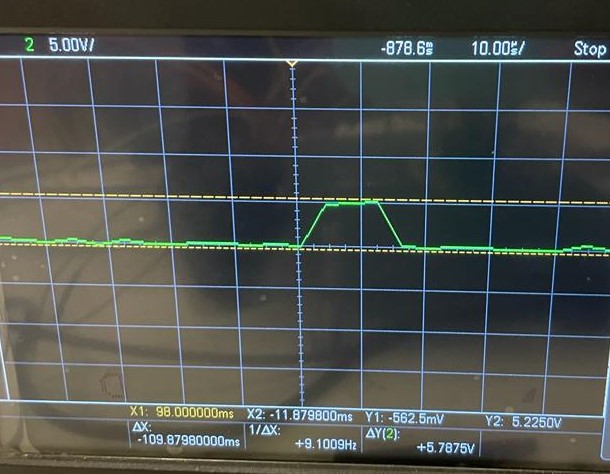
The shortest time it takes the ISR in my program to execute: 10.85us

The longest time it takes the ISR in my program to execute: 13us

**Compare the calculated ISR time to the time measured with the second port pin, which toggles at the beginning and at the end of each ISR execution. Do the calculated and measured times match? State your assumptions and explain any differences you see.**

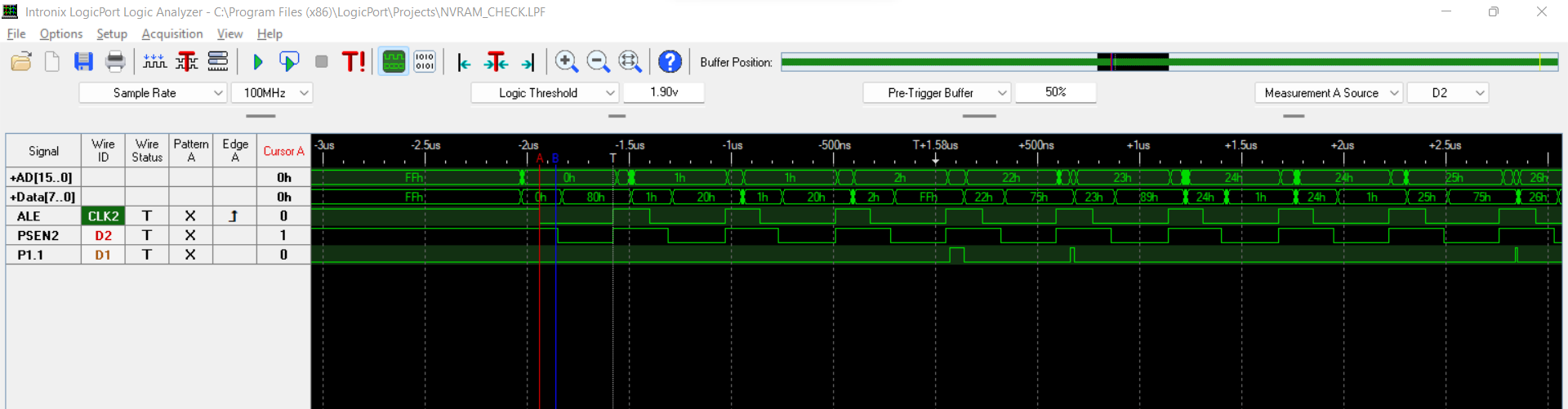


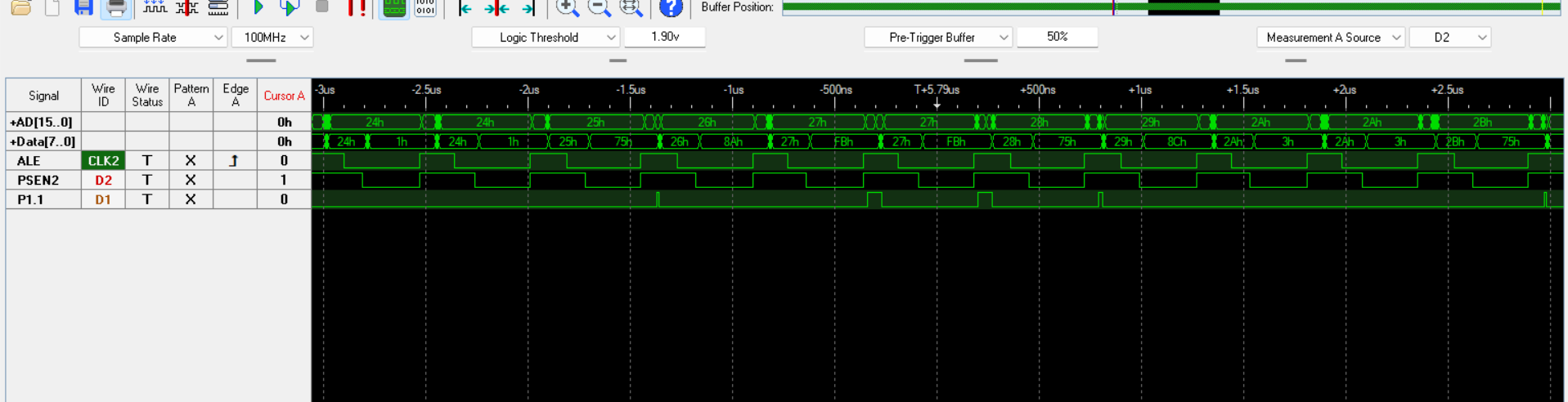
The measured time for the shortest ISR loop is around 12 us

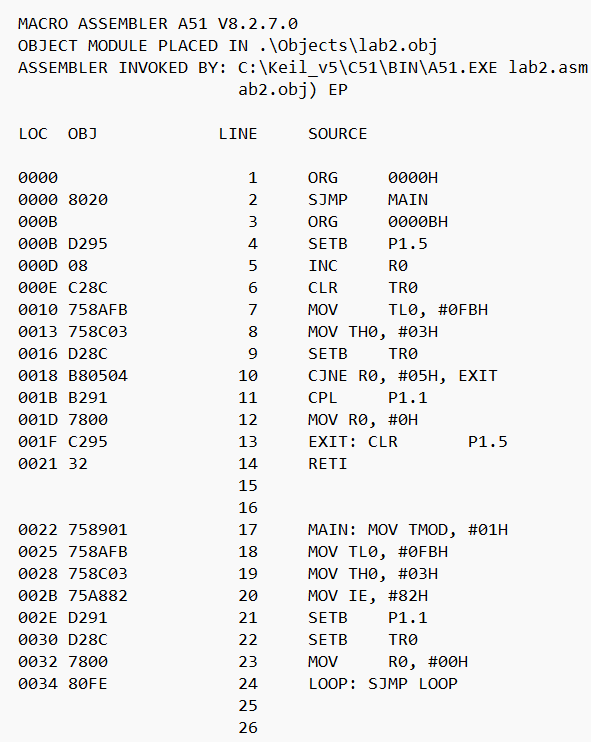


The measured time for the longest ISR loop is around 15 us.

**Using the state mode, capture a sequence of instructions and compare the sequence to the listing file for the code being executed. For the state clock, you can investigate using /PSEN, /READ or ALE.**





From the program, the memory location 0h holds the data 80h.

After reset is pressed, the logic analyser shows the data 80h at memory location 0h.

Similarly, when the main function is executed, the location 22h holds the data 75h.

The logic analyser shows the data 75h at memory location 75h.



The port pin P1.1 is toggling at frequency of 1.428Hz ie. The on-time and off-time are each 350ms.

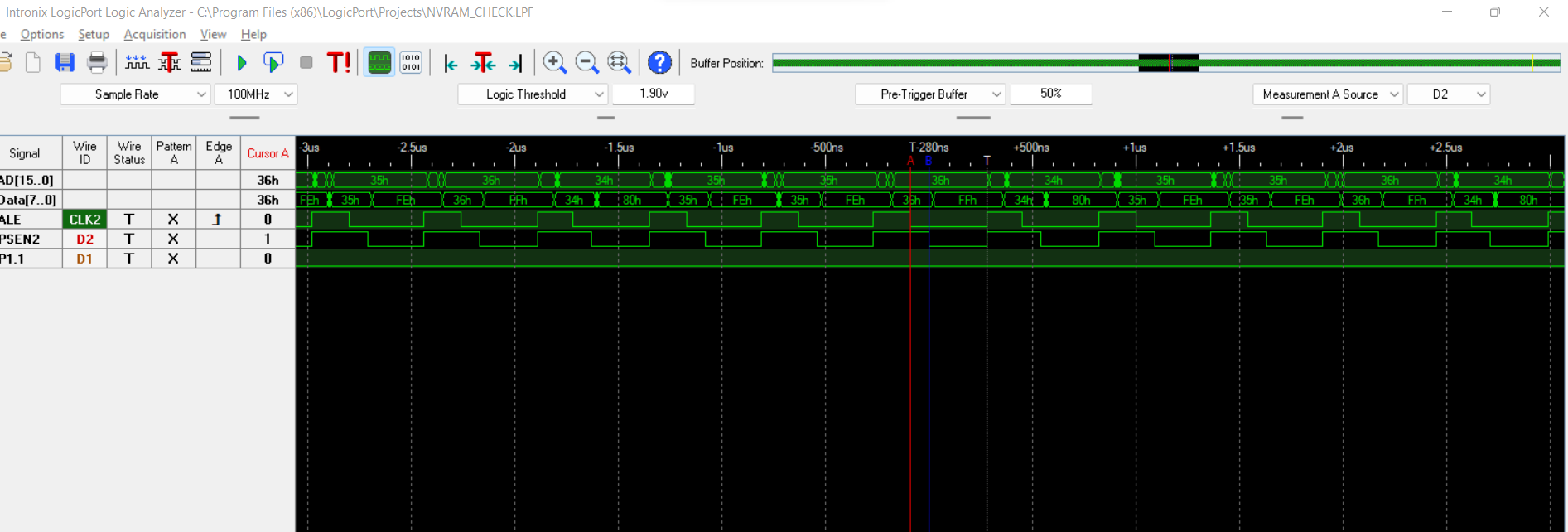
**Using the timing mode, measure the time which elapses from when the 74LS373 latches the**

**address supplied by the 8051 to when the PSEN signal is activated during an instruction fetch.**

**Before your demo to the TA, annotate a screen capture to show the measurement of tLLPL in**

**timing mode and prove that your measured time meets the processor data sheet specification**

**for tLLPL.**



The minimum amount of time between when the processor drives ALE low to when it drives the /PSEN low is Tllpl.

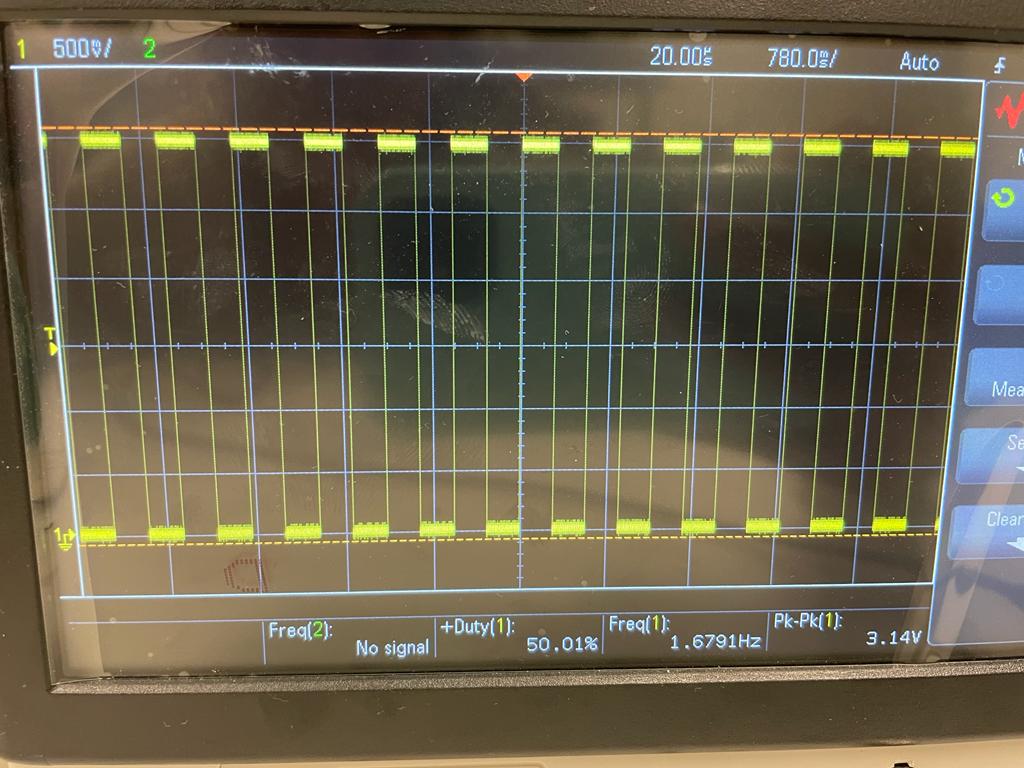
From the logic analyser the measured value=90ns.

The calculated value= [(1/f)-15] ns= 75ns.

## PART 2:

STM32

**18. b) Write a program to toggle the on-board red LED using a timer interrupt on the ARM processor.**

 The LED pin is toggling at a frequency of 1.679 Hz ie. The on-time and the off-time are each 300ms.

21. **[Part 2 Required Element1**] As part of your submission, provide answers to the following:

a) What operating system (including revision) did you use for your 8051 code development?

Windows 11- 64-bit operating system, x64-based processor

b) What assembler(s) (including revision) did you use?

The A51 Assembler is used as macro assembler for this lab.

c) What ARM development tools did you use?

I used the STM32CubeIDE by ST

d) Did you install and use any other software tools to complete your lab assignment?

I used Keil uvision to debug the assembly programs.

e) Did you experience any problems or challenges with this lab assignment or any of the software tools? If so, describe the issues.

None, whatsoever.

f) If you have any suggestions.

None.