**LABORATORY #2**

**Signal Waveforms and Parallel Port Programming**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | Student #1 | Student #2 | T.A. Initials | Date |
| Student Name |  |  |  |  |
| Student Number |  |  |  |  |
| Prelab/ Attendance |  |  |  |  |
| Lab #1 Completion |  |  |  |  |

# HOME PREPARATION:

1. Review the 8086 Hardware Reference Manual to familiarize yourself with the CPU bus signalling waveforms. Note that the 8086 or the SDK board is set up in Minimum Mode. Review the waveform diagram provided in Table 2.

**Review:** Appendix A, logic analyzer operation

 Appendix B, in particular the 8255A and DAC extensions.

1. Calculate the number of clock cycles expected in the program of PART A.
2. Write the program required in PART B. Ensure that it is well documented.
3. As in LAB #1. and for all following labs, bring in a 3.5" diskette.

# LAB #2: INSTRUCTIONS:

Fill in the blanks on these question sheets, (ONE per group). These sheets constitute your lab report. The T.A. will ask you questions about your work. *You will have to demo your work to the T.A.*

# PART A: BUS SIGNALS AND WAVEFORMS

In this part of the lab you will use a logic analyzer to monitor the bus signals on the CPU as it executes a small program. You must understand the relationship between the program and what appears on the bus. See APPENDIX A for settings and use of the logic analyzer.

1. Find the 50 pin male connector on the SDK board labelled 'DATA BUS' and plug the
logic analyzer pod into it. **NOTE:** *The logic analyzer should he powered down while
interring and removing the pod into its front panel.*
2. Assemble, link and convert to executable code the LAB2A.ASM file provided in the 361 (or 461) directory, then download this and run it on the SDK board (follow the downloading instructions provided in LAB #1, PART C).
3. Complete the timing diagram in Table 2 by copying the waveforms as observed on the logic analyzer.
4. Determine what the processor is doing at each clock cycle by looking at the address and data values on your timing diagram. The bus has either a 20 bit address or 16-bit data on it at one time, not both. Determine which one is on the bus and write down the events which are occurring and what each one means in Table I below. The first entry is given for you.

**TABLE I. Timing Diagram Event Description**

|  |  |  |  |
| --- | --- | --- | --- |
| **Clock****Cycle** | **Event (Read/****Write/Latch)** | **Address or Data****Value** | **Explanation** |
| 1-3 | Write data | A500 | Write A5H from AL into location MEM |
| 4 |  |  |  |
| 5-7 |  |  |  |
| 8-13 |  |  |  |
| 14 |  |  |  |
| 15-17 |  |  |  |
| 18 |  |  |  |
| 19-21 |  |  |  |
| 22 |  |  |  |
| 23-25 |  |  |  |
| 26-27 |  |  |  |
| 28 |  |  |  |
| 29-30 |  |  |  |

1. Explain why some of the address/data lines rise or fall in cycle 10?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Over which clock cycles is the MOV MEM, AL instruction executed (not fetched)''?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Look at the program LAB2A.ASM and refer to the 8086/8088 User’s Manual. How many clock cycles should this take? (Hint: Examine the relationship between the JMP instruction and the prefetch queue).

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Why is the number of clock cycles observed different from the number calculated above?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**TABLE 2. Timing Diagram**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Clock****Cycle** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1****0** | **1****1** | **1****2** | **1****3** | **1****4** | **1****5** | **1****6** | **1****7** | **1****8** | **1****9** | **2****0** | **2****1** | **2****2** | **2****3** | **2****4** | **2****5** | **2****6** | **2****7** | **2****8** | **2****9** | **3****0** |
| **\_\_\_\_****PCLK** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **ALE** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **\_\_\_****WR** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **\_\_****RD** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD0** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD1** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD2** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD3** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD4** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD5** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD6** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD7** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD8** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD9** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD10** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD11** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD12** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD13** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD14** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AD15** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A16** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A17** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A18** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **A19** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **ADD or DATA****MSB LSB** |  **A500** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Program fragment from LAB2A.ASM

(Loop occurs at 00106, AL=A5H)

 LOOP: MOV MSM, AL

 JMP LOOP

 MEM DS OOH, OFOH, 77H

# PART B: PARALLEL PORT PROGRAMMING AND THE D/ACONVERTER

In this part of the lab you will program the 8255A parallel I/O port to generate analog voltages via the wire-wrapped DAC0800 D/A converter expansion on the SDK board. *You will have to write a program to generate a sawtooth waveform.*

The D/Aconverter obtains its data from the P2B port of the 8255A chip (See APPENDIX B). This data is then converted into the corresponding voltage at the analog output between -4V and +4V. You can monitor this voltage with the oscilloscope. To use the P2B port for this purpose, the 8255A must first be programmed for this function via the P2 Control Register.

1. Using the keypad monitor on the SDK board, program the 8255A port P2B for output by outputting the control byte B5H to the P2 Control Register which is set up as address FFFEH on the SDK board. This is done by using the OB (output byte) command. According to the Peripherals Handbook (See page 3-104), what does this program the 8255A to do? *Please show all eight bits with an explanation of what each bit does.*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |

1. Now output a data value to the D/A converter by outputting to P2B at address FFFAH on the SDK board. If you wish, you can see these values appear on pins 18 to 25 on the second (lower) 8255A chip. This is port P2B. Monitor the voltage appearing on the D/A output pins, using the ground provided.
2. What output data produces -4V? \_\_\_\_\_\_\_\_\_ OV? \_\_\_\_\_\_\_\_\_ +4V? \_\_\_\_\_\_\_\_\_ +2V?\_\_\_\_\_\_\_\_\_
3. Write a program to generate a sawtooth waveform as shown below on the D/Aoutput. *The program should allow the period of the cycle (i.e. the rising time of the slope) and the amplitude of the sawtooth to be specified and independently controlled.* This can be done by either reassembling and redownloading the code, or by altering the program via the keypad monitor. The D/A output is viewed by connecting the oscilloscope between the D/A output and the D/A ground (both are marked on the left hand side of the SDK board).

*This program must be demonstrated to the TA. at the end of the lab. Remember to document it well!*

 Amplitude

 Period