EE 337: Microprocessors Laboratory (Spring 2024)

Indian Institute of Technology Bombay

MIDSEM: 25 points (Duration: 2.5 hours) Date: February 10, 2024

Instructions:

- You are allowed to use **only your** codes from previous labs. Sharing material or old codes is strictly not permitted.
- Use of internet during the course of this examination will be considered as copying and strict action will be taken.
  - **You must put your laptops in airplane mode.**
  - **No internet browser should be opened during the exam**
  - **Install a non-browser pdf reader to read pdf files during the exam.** Examples are Acrobat Reader, Nitro PDF Reader, Sumatra PDF Reader.
- Any discussion during exam is not permitted.
- At the completion of **every question**, **show your solution to your evaluating TA** for evaluation.
- For Question 2, it is advised to add features one by one as mentioned in the rubrics section to maximise your chances of getting partial credits if you are not able to finish within time.

1. [6 points] In this part, you need to generate a square waveform of a specific time period and duty cycle and show it on Keil Logic Analyzer.

   o The time period and duty cycle are derived from the last 2 digits of your roll-number. For example, if your roll-number if 22B01**69**, then the duty cycle is 60% and the time period is 9 *ms*. Refer to 1 for illustration.

   o If there is a zero in either of the last 2 digits, then derive both duty cycle and period using the non-zero digit. For example, if your roll-number if 22B05**4**0, then the duty cycle is 40% and the time period is 4 *ms*. Similarly if your roll-number if 22B050**8**, then the duty cycle is 80% and the time period is 8 *ms*.

   o If both your last 2 digits are zero, then derive both duty cycle and period using the previous non-zero digit. For example, if your roll-number if 22B1**2**00, then the duty cycle is 20% and the time period is 2 *ms*.

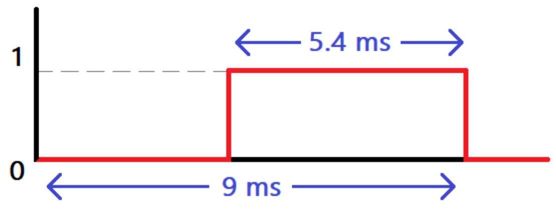

Figure 1: Square Wave of 9 *ms* period and 60% duty cycle

Use Port P1.0 to generate the square waveform. Observe it using Keil Logic Analyzer. Measure the obtained time period and on period, and calculate the obtained duty cycle from the waveform in the Logic Analyzer. They should match with the values derived from your roll-number.

2. [19 points] In this part, you need to compute the inverse of a 3×3 matrix.
If A is a non-singular square matrix of shape 3×3, the inverse of A is given by,

$$A^{-1} = \frac{\text{Adj}(A)}{|A|}$$

Illustration of the computation:

* Let the elements of the matrix A be depicted as follows- $A = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix}$

* The Determinant is given by the following formula.

$$|A| = a_{00}.(a_{11}.a_{22} - a_{12}.a_{21}) - a_{01}.(a_{10}.a_{22} - a_{12}.a_{20}) + a_{02}.(a_{10}.a_{21} - a_{11}.a_{20})$$

* The Adjoint is the transpose of the co-factor Matrix of A.

* The co-factors are given by the following equations.
$c_{00} = a_{11}.a_{22} - a_{12}.a_{21}$; $c_{01} = a_{12}.a_{20} - a_{10}.a_{22}$; $c_{02} = a_{10}.a_{21} - a_{11}.a_{20}$
$c_{10} = a_{02}.a_{21} - a_{01}.a_{22}$; $c_{11} = a_{00}.a_{22} - a_{02}.a_{20}$; $c_{12} = a_{01}.a_{20} - a_{00}.a_{21}$
$c_{20} = a_{01}.a_{12} - a_{02}.a_{11}$; $c_{21} = a_{10}.a_{02} - a_{12}.a_{00}$; $c_{22} = a_{00}.a_{11} - a_{01}.a_{10}$

* $\implies \text{Adj}(A) = C^T = \begin{pmatrix} c_{00} & c_{10} & c_{20} \\ c_{01} & c_{11} & c_{21} \\ c_{02} & c_{12} & c_{22} \end{pmatrix}$

Assume that all intermediate and final results are 8-bit numbers ONLY, which means 16-bit or 24-bit arithmetic need NOT be implemented. Here are some more guidelines to perform the task.

o The elements of the matrix A are to be stored in memory locations `60h` to `68h`.

o Create a sub-routine/function to compute $y = a.b - c.d$, since this computation is being used several times.

o Using the above sub-routine, compute the determinant of A. If $|A| = 0$, the matrix is singular (not invertible), so you need to terminate the program (i.e skip the adjoint matrix computations and jump to the end). To indicate that the given matrix is singular, store `FFh` in locations `70h` to `78h`.

o If $|A| \neq 0$, then the matrix is invertible, so you can continue with the next step of calculating co-factors. Store the co-factor matrix in locations `70h` to `78h`.

o Then write a sub-routine/function to perform matrix transpose to obtain $\text{Adj}(A)$ and then divide each element by $|A|$ to obtain A^{-1}. The elements of the matrix A^{-1} are to be stored in the same memory locations (i.e. `70h` to `78h`).

RUBRICS

- Part 1:
  - 3 MARKs for correct time period duration.
  - 3 MARKs for correct duty cycle percentage.
- Part 2:
  - 2 MARKs for correct working of $y = a.b - c.d$ (i.e determinant of 2×2 matrix) sub-routine.
  - 3 MARKS for correct working of determinant of 3×3 matrix sub-routine.
  - 2 MARKS for checking if matrix is invertible or not and terminating the program if it is not invertible.
  - 3 MARKS for correct calculation of co-factor matrix elements.
  - 3 MARKS for correct working of transpose sub-routine.
  - 6 MARKS for correct final result of A^{-1} for the given test cases.
    (2 test cases will be provided, both need to be verified. 3 MARKS for successful verification of each test case)