NATIONAL UNIVERSITY OF SINGAPORE

Department of Electrical and Computer Engineering

Department of Electrical and Computer Engineering

EE2020 Laboratory

D1 Manual I

Session: Semester I, 2014/15

I. **D1 Intro Details** are applicable for D1 Intro session where you will gain familiarity with implementing TTL circuits on breadboard.
II. **D1 Project Details** give you more details regarding the D1 project, which you will be implementing progressively through D1A, D1B and D1C sessions.

I. D1 INTRO DETAILS

A. Objectives of the D1 Intro Session

1) To learn implementing TTL[1] digital circuits on breadboard.
2) To implement
    a. 5V regulator using 7805
    b. 1Hz clock using 555 timer
    c. A single pulser circuit using 74175 (Quad D-FFs) and 7408 (Quad AND gates), *taking up a minimum area* of the breadboard, and f*ollowing good wiring practices*.

B. Significance of D1 Intro Session

Though D1 Intro session itself is not graded, it is important as

- Familiarity with implementing digital circuits using TTL ICs is crucial, as we will be using them for all D1 sessions. D1 component weighs **30%** towards the final grade.
- You might be using the single pulser circuit throughout your D1 projects.
- The regulator and timer will allow you to debug your circuit at home in case you couldn't get it to work during the designated lab session(s).

C. Wiring Practices and Practical Considerations

1. Assemble the timer and the power supply at a corner of the breadboard, and use minimum number of breadboard slots for their implementation. This will leave enough space for building your circuit incrementally (you will have to re-wire portions of your circuit if you are running out of space).
2. Ensure that ALL unused input pins of your TTL chips are connected to appropriate voltages (they should not be left floating). An unconnected input will be taken as a 'high' in TTL; but to ensure trouble free operation, it should be connected physically to +5V.
3. Color code wires for easier debugging of your bread boarded circuit.
4. Wires should be just long enough to connect any two points on the bread board. The stripped part of the wire should go in about 0.7cm into the breadboard slot to make a proper connection.
5. Wires should not cross each other or go over any chip.
6. The leads of resistors, capacitors and LEDS should be just long enough to connect any two points on the bread board.
7. Mind the polarity of LEDs and electrolytic capacitors.
8. Allocate a row on your breadboard for Vcc (+5V) and another for GND (0V).
9. Have a plan on the placement of ICs on the breadboard. Make judicious use of slots available on the breadboard, and avoid the necessity for long wires. (for example, if 2 ICs have a number of interconnections, place them closer).
10. Check each unit independently for proper functionality before interconnecting them.
11. Do not resort to re-wiring your circuit each time it refuses to work. Use your logic to debug the problem.
12. Those who wish to test their circuits at home can use the 7805 fixed voltage regulator to get an accurate 5-volt supply from a 9 Volt battery.

[1]Transistor-transistor logic, a common IC logic family, and the one we use in our labs.

II.Tasks for D1 Introduction 1

A. 5V regulator using 7805

Most TTL digital circuits require a well-regulated +5 volt DC power supply. The 7805 fixed voltage regulator, as illustrated in Fig. 1 provides an economical method to obtain such a voltage provided the input voltage is in the range +7 - 15V.

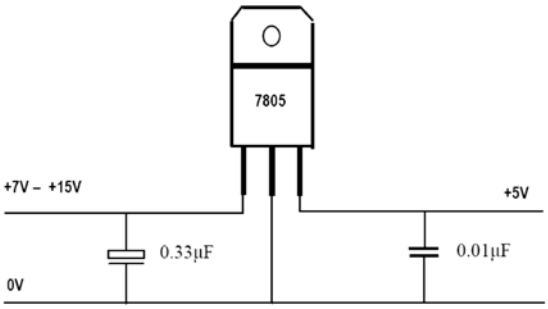

Figure 1: +5V supply using 7805 fixed voltage regulator.

Construct the +5V power supply using the schematic diagram in Fig. 1. Connect the input to a bench power supply unit and set input voltage to approximately 9V. Measure the output voltage (from the 7805 voltage regulator) and ensure that it is +5V.

You should be using this circuit to power up your ICs and other devices throughout EE2020 labs.

B. Free running clock using 555 timer

D1 project will require a clock input. A schematic diagram of a free-running clock-pulse generator, using a 555 timer IC is illustrated in Fig. 2. Frequency of the generated clock pulse is determined by resistors R1, R2 and the capacitor C according to the following equation:

$$f_{CK\_OUT} = \frac{1.44}{(R1 + 2 \times R2) \times C}$$

where R1 and R2 are specified in Ohms (Ω) and C is specified in Farads (F). Additionally, the time (t_{high}) that clock output (CK_OUT) remains at HIGH (+5V) and the time (t_{low}) that it stays at LOW (0V) during a single clock pulse are determined by the following equations:

$$t_{high} = 0.69 \times (R1 + R2) \times C \quad ; \quad t_{low} = 0.69 \times R2 \times C$$

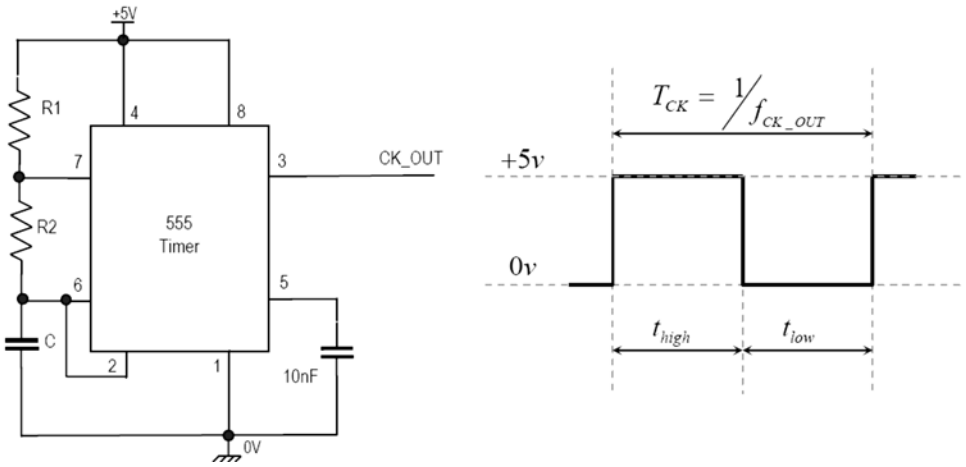

Figure 2: A free running clock pulse using 555 Timer.

Construct the free-running clock pulse generator using the schematic diagram in Fig. 2. Select R1, R2 and C suitable for an output frequency of approximately 1 Hz according to given equations. (***Hint****. First select a suitable value (from those available) for C and then choose R1 and R2 accordingly. There is a wider range of available values for resistors as compared to those for capacitors*).

C. Single Pulser Circuit

Many digital circuits have a user input in the form of a pushbutton. Pushbutton is a simple device which makes a connection when it is pressed and breaks the connection when released. The press can be converted to a digital signal using the configuration given below.

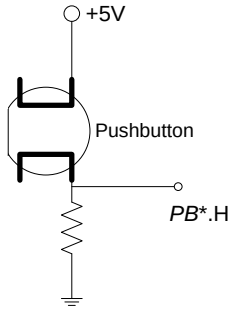

Figure 3: Pushbutton configuration to obtain an active high signal.

However, mechanical contacts are subject to contact bounce, which can create problems with practical systems (it is less of an issue for systems running on a low frequency clock though). Also, the user input is not synchronized with the clock and can remain asserted for several clock cycles. In many circuits, we might want to clean up the signal from the pushbutton such that it is *synchronized to the clock* and is of *exactly one clock cycle duration*. A single pulser circuit based on D flip-flops which serves the purpose is given below (Fig. 4a). The relevant waveforms are given in Fig. 4b to help you understand the functioning of the circuit.

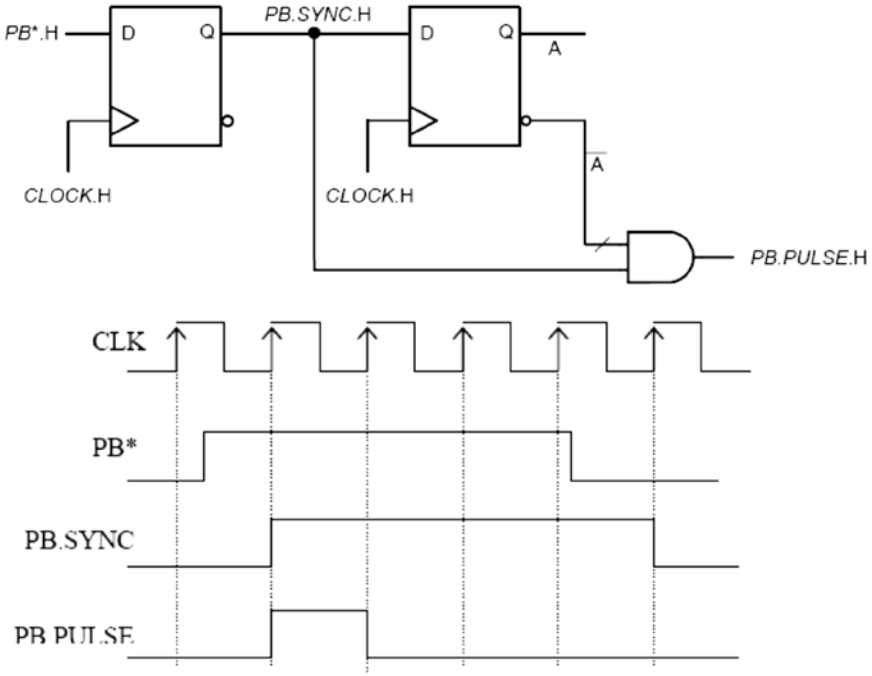

Figure 4: (a) A single pulser circuit (b) resultant waveform.

III. Tasks for D1 Introduction 2

- **Synchronous sequential circuit**: Output responds to inputs at discrete time instants governed by a clock input. Synchronous counter is a typical synchronous sequential circuit.
- A Mod-X counter counts from 0 to X-1. For example, a mod-6 counter give an output sequence as follow: 0,1,2,3,4,5,0,1,2,3… A mod-16 synchronous counter is shown in figure 5.

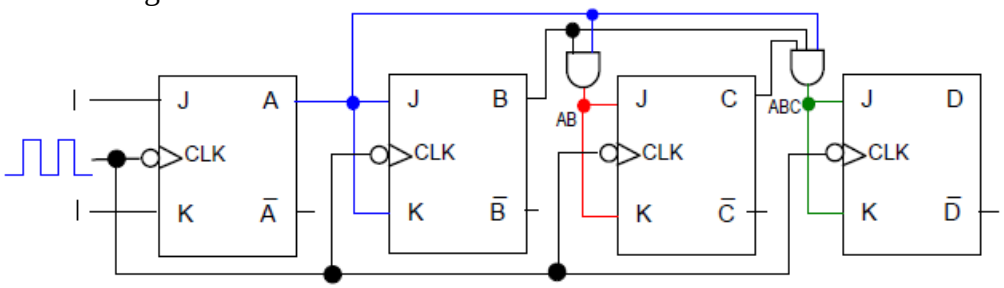

Figure 5: Mod-16 Synchronous Counter

In D1 Introduction 2 sessions, students are required to design and implement two mod-6 synchronous counters by the given components.

A. Design A Mod-6 Synchronous Counter by using flip-flops and NAND logic gates

- Given two JK flip-flop chips and NAND gates, design a mod-6 synchronous counter.

B. Design A Mod-6 Synchronous Counter by using two MSI chips (optional)

- Design a mod-6 counter by any two MSI chips.
- No logic gate is allowed to use in part B

Submit a **one-page report** to the assigned GA with the circuit diagram of the counter

IV. D1 PROJECT DETAILS

I. Project Guidelines and Grading Information

- You will progressively be building an application through the rest of your D1 sessions (D1A, D1B, and D1C).

- You need to be familiar with the list of ICs given in Section IV-B.

- Please follow the considerations given in Section IV-C while designing your circuit.

- You will be assessed continuously over the course of the project. The rubrics used for assessment are
  (i). Ability to execute a design process.
  (ii). Ability to verify that a design achieves desired needs.
  (iii). Neatness of your circuit and observance of wiring rules.

  (i) and (ii) are assessed based on the correctness of your design, your understanding of the design process and the requisite background knowledge (as can be inferred from your response to the questions asked by GAs), as well as the functionality of your circuit.

  The expectation level on (iii) will be higher if you are unable to demonstrate the assigned task during the lab session itself.

  In addition to the above, there is a small discretionary component, awarded by GAs based on your motivation, elegance of design, extra features implemented (if any) etc.

- The **30%** weight carried by D1 project is split as **5%**, **15%** and **10%** for D1A, D1B and D1C sessions respectively.

- The tasks for D1A, D1B and D1C are given in the D1 project manual.

- If the absence is due to *valid reasons*, please submit your **original** medical certificate / excuse letter / other documented evidence the very next day after the last day you are excused for by the document. Late submission will not be entertained. **It is noted that hall activities and extra curriculum activities are not considered as valid reasons**. Please verify with Teaching Assistant (TA) first if you are not sure.

- An appropriate penalty applies on absentee's final grade unless you were *absent with valid reasons*. There will not be any make-up sessions per se for absentees.

- The final D1 Assessment will be carried in week 13 (**10-14 Nov 2014**). Each student has 10 minutes to present and demonstrate their D1 project. GAs will test students' understanding of the design during the presentation.

- A 5-page report will be handed in to the assigned GA before the presentation is started.

- The report (maximum of 5 pages) covering the whole project. There is no specific format for the report, but as is with the report of a small project, it may have a brief introduction, design methodology, parts used and justification for their use, neat & labeled schematic(s), brief explanation of sub-circuits, experiment results (state clearly if the result matches the design requirements – if it doesn't, explain why not), understanding/experience gained from this project, possible improvements, etc.

- We will perform necessary normalizations to ensure that *no student is at a disadvantage because of being assigned a specific project or being under a particular GA.*

II. List of ICs

- A list of popular TTL integrated circuits is given below.
- Make sure that you understand the working of *all of them very well* ***before*** coming for the lab sessions.
- You are encouraged to remember *or* bring printouts/e-copies of relevant parts of the datasheet (pinouts, function table/timing diagram) for all of the ICs for D1 sessions.
- Datasheets can be obtained from the lab or from the manufacturers' websites (you can simply Google part number+datasheet).
- You can use devices which are not listed below if they are available in the lab - please check the availability with the lab staff.

Sl.No.	Description	Part Number
1	Quad 2-input NAND gates	7400
2	Quad 2-input NOR gates	7402
3	Hex inverters	7404
4	Quad 2-input AND gates	7408
5	Dual 4-input NAND gates	7420
6	Dual 4-input AND gates	7421
7	Quad 2-input OR gates	7432
8	Quad 2-input XOR gates	7486
9	Triple 3-input AND gates	7411
10	Dual 4-input NAND gates	7420
11	Dual 4-input AND gates	7421
12	Triple 3-input NOR gates	7427
13	BCD to 7-segment Decoder	7447/7448
14	4 bit magnitude comparator	7485
15	3 to 8 Line decoder / demultiplexer	74138
16	Dual 2 to 4 line decoder / demultiplexer	74139
17	Quad 2 to 1 line data selector/multiplexer	74157
18	Quad D-type flip flop	74175
19	Synchronous 4-bit up/down counter with mode control	74191
20	Synchronous 4-bit up/down counter with dual clock	74193
21	Dual D-type flip flop with preset and clear	7474
22	4 bit universal bi-directional shift register	74194
23	4 bit binary adder	74283

III. Design Considerations and Hints

1. You may use a single pulser circuit (implemented in D1Intro) to clean up the signal from a push-button to obtain a pulse that is high for exactly one clock cycle.

2. The clock inputs of **all** your sequential devices should be connected to a single free running clock signal; example shown in Fig. 6.

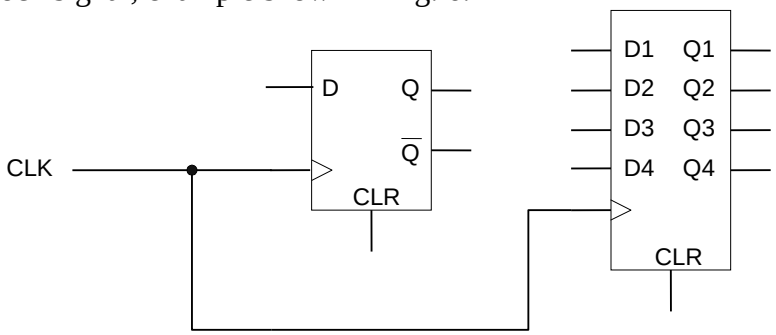

Figure 6: The same clock signal must be used for all sequential device clock inputs.

3. Do **not** connect the inputs (DIP and push button switches) to the clock inputs of sequential devices. Clock inputs must be connected to the clock signal only (Fig. 7).

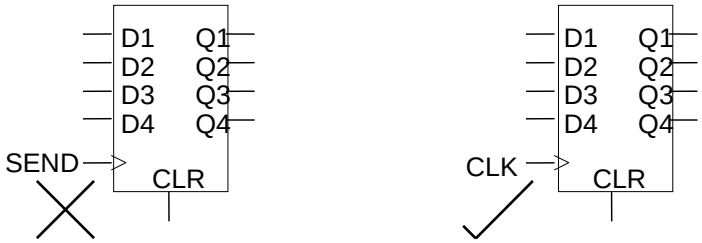

Figure 7: A clock pulse should be connected to the clock inputs of all the sequential devices.

4. Do not gate the clock input (Fig. 8).

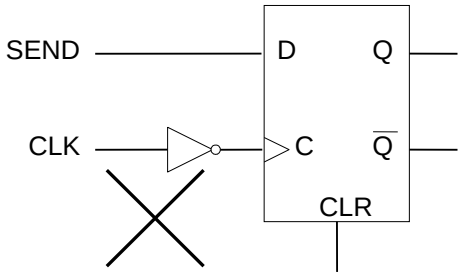

Figure 8: Clock signals must go directly into sequential device clock inputs.

[Hint : To inhibit counting, use *enable* function rather than gating the clock.]

5. Design your circuit **on your own**, after building up a good knowledge on the basic building blocks of digital circuits i.e. flip-flops, counters, registers, multiplexers, decoders etc. Identify the functional units you need, and find the most appropriate IC/combination of ICs to map it to.

6. Depending on your design, using **MSI chips** (such as decoders, comparators etc.) may dramatically reduce the number of components required for the circuit compared to using only basic gates such as NAND's, NOR's and inverters. MSI chips tend to yield clearer and more efficient designs.

7. You are encouraged to use **simulation software** such as Xilinx ISE's schematic option or Circuitlogix (free student version) or Multisim (30 day evaluation version or free Analog devices edition) to gain better understanding of the working of various components/ICs through simulation.

8. Your **design** has to be implemented on the **2 strips of breadboard** provided to you. You may not be able to accommodate more than 12 ICs on them.

9. From our experience, **7416x ICs** might have problems clocking from 555 timer. If you design using 7416x and experience problems, you might want to use the TTL clock from the Function Generator instead of the 555-based clock. However, this will limit your ability to work on your circuit outside the designated lab sessions.

10. Good luck and have fun!