****

**UNIVERSITI TEKNOLOGI MALAYSIA**

**SCHOOL OF COMPUTING**

**FACULTY OF ENGINEERING**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Group Members | MOHAMAD AMIN HAZEEQ BIN HISHAM | |  | A19EC0087 |
|  | MUHAMMAD NURSYAKIRIN BIN AMIR HAIDI  IMRAN HAKIM BIN NORASMADI | |  | A19EC0105  A19EC0189 |
| Title |  | PHOTOCOPYING(XEROX ) MACHINE PROJECT (LAB 4 MINI PROJECT ) REPORT | | |
| Name of Lecturer |  | Dr RASHIDAH BINTI KADIR | | |
| Subject |  | SECR1013-02 LOGIK DIGITAL (DIGITAL LOGIC) | | |
| Section |  | 02 | | |
| Semester |  | 1 2019/2020 | | |

1.0 Table of Contents

|  |  |  |
| --- | --- | --- |
| Bil | Contents | Pages |
| 1. | 1.0 Table of Contents | 0 |
| 2. | 2.0 Report Content  2.1 Dedication & Acknowledgement | 1 |
| 3.0 The Background | 1 |
| 4.0 The Problem | 1 |
| 5.0 Objectives | 1 |
| 6.0 The Flowchart | 2 |
| 7.0 Components | 3 |
| 8.0 Materials and Software’s Used | 3 - 4 |
| 9.0 Circuit Implementation | 4 - 6 |
| 10.0 Physical System Implementation | 6 - 7 |
| 11.0 Discussion | 7 |
| 12.0 Conclusion | 8 |
| 13.0 References | 8 |
| 14.0 Appendix | 9-12 |

**2.0 Report of Content**

**2.1 Dedication & Acknowledgement**

**2.1.1 DEDICATION**

Thanks a lot to Anas Alif, Safwan and Fatimah Ad-Dzikrun to guide us the way to complete our circuit in part IV (3-bits up counter) especially our problem at wires that it supposed to go. My group helped each other and also with our course mate to make sure everyone was not left behind.

**2.1.2 ACKNOWLEDGEMENT**

We also give a very special thanks to our lecturer, Pn. Rashidah binti Kadir for teach us about this mini project (photocopying machine). Thank you again for teach us step-by-step before we start the project until we finished it successfully. Without her continuous supervision on her, we won’t be able to complete this project successfully.

**3.0 THE BACKGROUND**

We had been read the appendix in our lab book to familiarize ourselves with the WinCUPL compiler and the universal programmer. We also had some problem on doing this project so, we decided to ask for help with our lecturer and also teacher assistants to give the more explanations and guidance to complete our circuit. This mini project will implement 5 different components on a single GAL device, those components are 2-bit Count Up Counter, 2-bit Comparator, 3-bit Count Up Counter, 3-bit Comparator and Clock Disabler.

**4.0 THE PROBLEM**

Initially, we enter amount of copies, the counter will count the numbers of copies that has been photocopied. The machine will stop once the required number of copies produced.

**5.0 OBJECTIVES**

The objectives for this laboratory are to introduce the students to: - Do development of a PLD devices and do a simple Hardware Description Language

**6.0 THE FLOWCHART**

START

Insert input using switches

Insert input using switches

Display input in decimal form on LED display

Clock Disabler

Clock Generators



Display output of counter in decimal form on LED display

Counter

Burn JED File to GAL22V10 using Wellon programmer and tester

Complete program and generate JED file

Type code using WinCUPL

Start

Connect circuit on breadboard

End

Produced a report for project

Test and analyze the circuit

**7.0 COMPONENTS**

1. Input switches
2. 2-bit up counter
3. 2-bit comparator
4. 3-bit up counter
5. 3-bit comparator
6. 7-segments LED display
7. Clock disabler

**8.0 MATERIALS**

1. **Breadboard**: A breadboard has holes lined with metal, wires and electrical components can be plugged in to complete the circuit. Breadboard must be connected to the power supply (voltage) and also the ground segments which are located at the lower part of breadboard.
2. **GAL22V10:** It is a type of chip that manufactured by the Lattice Semiconductor Corporation. This chip has 24 pins:
   * Pin 12 is ground.
   * Pin 24 is VCC (5V).
   * Pins 1 until 11 and pin 13 are inputs.
   * Pins 14 until 23 can be identified as inputs or as outputs.
3. **ETS-5000 DIGITAL TRAINING KIT:** Used for placing breadboard on it, setting up**.** It is a type of tool to help and enhance the understanding of digital theory.

**8.1 SOFTWARE’S USED**

1. **Wellon Universal Programmer & Tester:** Kinds of software to program PLD. Basically burn the PLD with the desired coding from WinCUPL.
2. **WinCUPL 5.0 Software**: Software that used to implement coding for PLD**.**

Before start coding, we need to key in number of input and output pins so that the template will appear in the editor (make us easy to type our program).

1. **Handouts:**

* “WinCUPL user manual”
* “ATMEL22V10 Data Sheet"
* “How to use Hi-Lo Programmer”
* “How to use Wellon Programmer”
* “How to use Win CUPL 5”

**9.0 CIRCUIT IMPLEMENTATION**

This was the proposed circuit that was implemented. The block diagrams of the components that were required for the 2-bit up counter and 3-bit up counter are shown in Figure 1 and Figure 2 respectively. Both circuits are similar in their essential components which are the 2/3-bit counter, 2/3-bit comparator and clock disablers. The counter will determine the number of copies that has been made while the comparator will compare and determine whether the required number of copies has been met. Once both the numbers of copies have been met and is equal, the clock disabler will disable the clock and stop the counter from counting up. The number of copies that is currently printed and the required number of copies will be displayed onto two separate 7-segement LED displays respectively.

To implement the proposed circuit, the user (us) has to key in the required number of copies beforehand by using 2 switches for the 2-bit up counter and 3 switches for the 3-bit up counter. For the 2-bit up counter, the number of copies that can be entered ranges from 0 to 3 in decimal (00 to 11). On the other hand, the number of copies that can be entered ranges from 0 to 7 (000 to 111). The user will have to reset the counter (using the preset switch), then the user can start counting up by using the switches. Then the counter will count up and at the same time it’s output will be compared by the comparator with the values of the switches. If the value is not the same it will continue to count up. The counter will stop counting up only when both the values are equal, a signal will be generated to the clock disabler to stop the counter from counting up. Shown in Figure 3 and Figure 4 is the circuit implementation on the ETS500 Training kit for the 2-bit counter and 3-bit counter respectively.

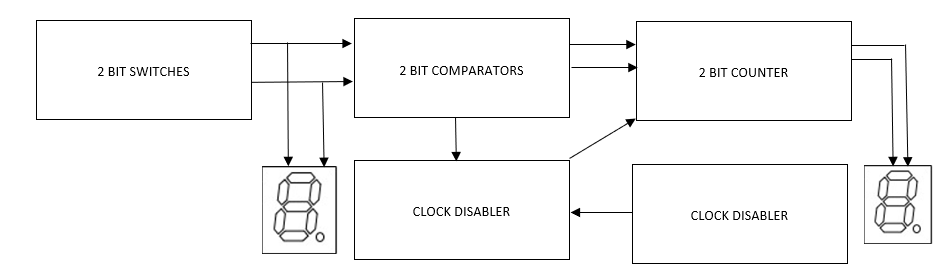


Figure 1 Block diagram of the 2-bit count up

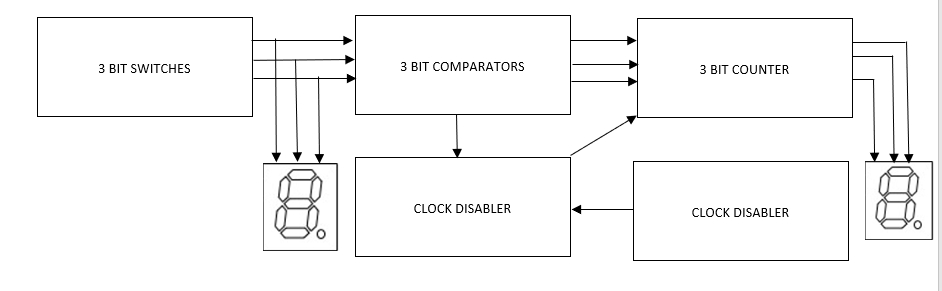


Figure 2 Block diagram 3-bit count up

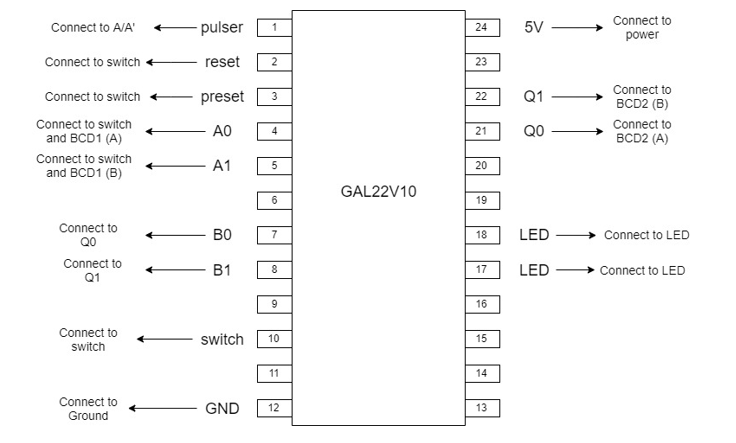


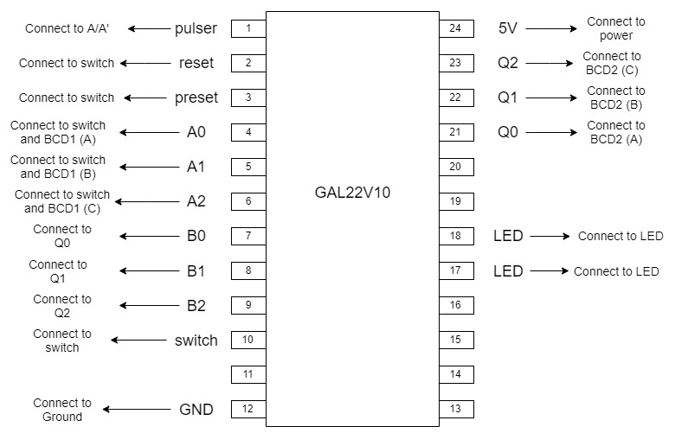
Figure 3: Circuit Implementation for 2-bit up count

Figure 4: Circuit Implementation for 3-bit up count

**10.0 PHYSICAL SYSTEM IMPLEMENTATION**

The first step for making 2 bit Synchronous Count-up Counter output to LED is we connect pin 12 to ground and pin 24 to voltage. Then we connect pin 1 to pulse which is at the left bottom of the Training kit and it can be either A or A’, B or B’. For this lab exercise we use A’. Next is connect pin 2,3,4,5 to switch 4,3,1,0 respectively and also connect pin 10 to switch 5. Next attach pin 4 and pin 5 to BCD1(A) and BCD1(B) respectively. Then connect pin 7, pin 8 to pin 21, pin 22 respectively. At the same time connect pin 21, 22 to BCD2(A) and BCD2(B) respectively. For LED output connect pin 18 and 17 to LED 0, 1 respectively. Next connect BCD1(C), BCD1(D) ,BCD2(C) and BCD2(D) to the ground. Finally make a jumper at ground if needed to ensure there is connection .For a purpose to change this circuit into 3 bit we just need to use extra two unused pin in this ic. For extra input pin 6, 9, 10, 11 is available and so we need an extra output .For that we have pin 13,14,15,16,19,20,23 available. But it depends what program you have implement on your ic. For this lab session we use pin 6,9,23. We connect pin 6 to switch 2 and pin 9 to pin 23. At the same time connect pin 6, 23 to BCD1(C) and BCD2(C) respectively. Lastly, BCD1(D),BCD2(D) connected directly to the ground and then you should get your circuit as in figure 1.

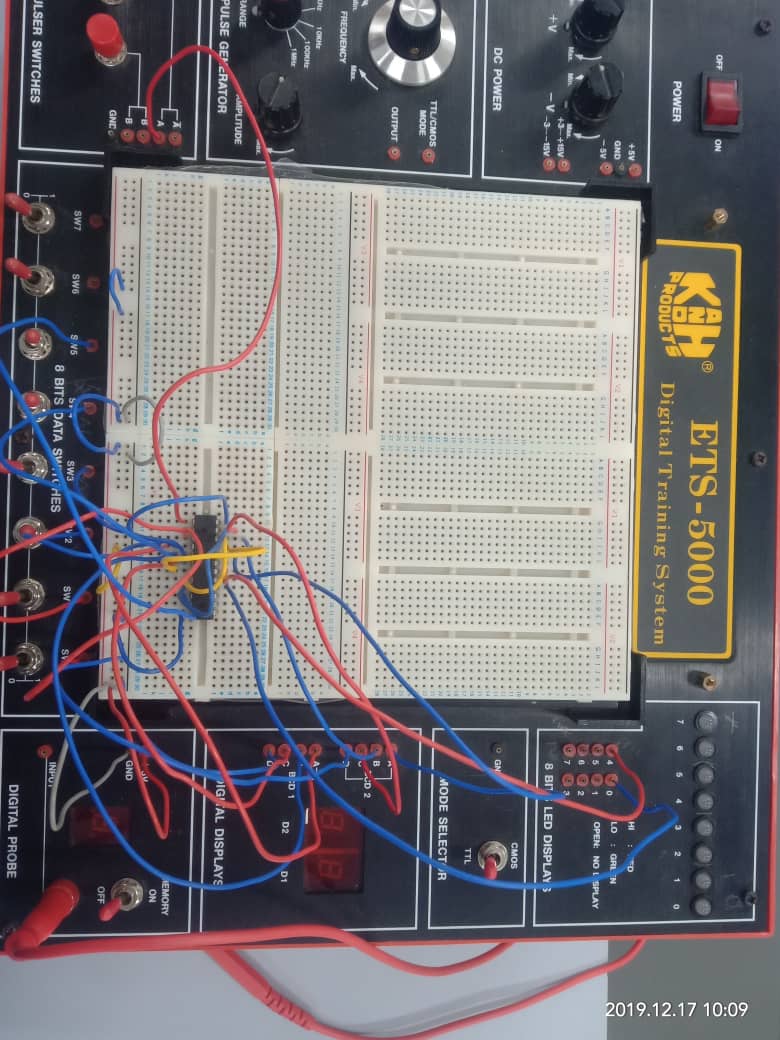


Figure 1: 3 bit Synchronous Count-up Counter

**11.0 DISCUSSION**

The main objectives of this lab were to introduce the students to the development of a PLD device and a simple hardware description language. These objectives were met throughout the entirety of doing lab 4. Before implementing the circuit for our lab 4, we had to understand the rough flow of what was to be done before and during the lab 4. The day before the lab, our group had already been briefed and read the flow of lab 4. Not only that, using WinCUPL our group saved the program used for the 2-bit counter and 3-bit counter into a pen drive to be brought the next day. On lab day, the program saved were burned into the PLD using Wellon Programmer and Tester. After assembling the corresponding circuits, we the user had to test the circuit by using the switches to enter the maximum range of the number of copies. Then resetting the counter, using the switches to start counting up and then comparing the output with the value of the counter. The value will continue counting up when the values are not the same.

One of our group’s strength was in our numbers, we had an extra member compared to other groups which made checking the program and circuit after assembling easier, and faster. Not only that, when something wrong or not in plan occurred, we were patient and we rationally analyzed and solve the problems faced. We were able to successfully implement and test the 2-bit up counter circuit. But the problems started to arise after assembling the 3-bit up counter circuit. After numerous times of rewiring and reprogramming we were only able to partially complete the output. Our group and Madam Rashidah our lecturer were unable to locate the problem after checking our wiring and program numerous times. The project could be further improved by adding more another functions to enable the circuit to count down.

**12.0 CONCLUSION**

In this new era that we knew a lot of small component can do such a lot of thing. For example we have USB drive that can store a lot of data. Another magnificent creation is integrated circuit(IC) which can be function amplifier or microprocessor. This is because there is a lot of circuit in combined in the ic . How exactly they manage to implement circuit exactly to ic? The reason is digital logic. In this lab exercise it enhanced our communication skill and teamwork between us. The result due to this skill is we manage to complete this lab exercise in the short amount of time. As we know that when we trying to invent something maybe more smaller than ic that can do a lot of thing we need digital logic knowledge. Not only that, this subject also enhanced your critical thinking skill. As a computer science student we need to develop something new and can be used by other people. Therefore we can start by creating component that is more function and smaller than ic. But it can only when you have high critical thinking skill and high knowledge about implement a lot of circuit in one component. This is when digital logic takes part and it is important and we need to master it as now it’s a trend of technology that some small component can do a lot of things.

**13.0 REFERENCES**

1. Eric Coates MA BSc. (2007 – 2019) .Digital electronics. England, LEARN ABOUT ELECTRONICS. Retrieved from <http://www.learnabout-electronics.org/Digital/dig43.php>.
2. Steven Melendez (2018). The function of breadboard, Techwalla. Retrieved from this link <https://www.techwalla.com/articles/the-functions-of-a-breadboard>
3. Abd. Bahrim Yusoff, Mazleena Salleh., Mohd Fo'ad Rohani, Ismail Fauzi Isnin (2019). DIGITAL LOGIC. Johor, Malaysia, DESKTOP PUBLISHER.

**14.0 Appendix**

**Program source codes for 2-bit XEROX system :**

Name Xerox Machine 2bit ;

PartNo 00 ;

Date 17/12/2019 ;

Revision 01 ;

Designer Engineer ;

Company UTM ;

Assembly None ;

Location ;

Device G22V10 ;

/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\* INPUT PINS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

PIN 1 = clk ; /\* clock \*/

PIN 2 = reset ; /\* reset \*/

PIN 3 = preset ; /\* preset \*/

PIN 4 = a0 ; /\* Comparator A \*/

PIN 5 = a1 ; /\* \*/

PIN 7 = b0 ; /\* Comparator B \*/

PIN 8 = b1 ; /\* \*/

PIN 10 = startPrt ; /\* Start Printing \*/

/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\* OUTPUT PINS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

PIN 17 = diffCmp ; /\* XOR (A B not equal HIGH) \*/

PIN 18 = sameCmp ; /\* XNOR (A B equal HIGH) \*/

PIN 21 = q0 ; /\* output counter \*/

PIN 22 = q1 ; /\* output counter \*/

/\*\*\*\*\* Function Comparator\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

sameCmp = !(a0$b0)&!(a1$b1);

diffCmp = !sameCmp ;

/\*\*\*\* Function Clock Enabler \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

clkEn=startPrt & diffCmp;

/\*\*\* Function Counter 2 Bit UP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

field count =[q1..0];

$define s0 'b' 00

$define s1 'b' 01

$define s2 'b' 10

$define s3 'b' 11

count.ar=reset; /\* connect reg AR to reset (Asyn Mode) \*/

count.sp=preset; /\* connect reg AR to preset (Syn Mode) \*/

sequence count{

present s0 if clkEn next s1;

default next s0;

present s1 if clkEn next s2;

default next s1;

present s2 if clkEn next s3;

default next s2;

present s3 if clkEn next s3;

default next s3;

}

**Program source codes for 3-bit XEROX system**

Name Xerox Machine 3bit ;

PartNo 00 ;

Date 17/12/2019 ;

Revision 01 ;

Designer Engineer ;

Company UTM ;

Assembly None ;

Location ;

Device G22V10 ;

/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\* INPUT PINS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

PIN 1 = clk ; /\* clock \*/

PIN 2 = reset ; /\* reset \*/

PIN 3 = preset ; /\* preset \*/

PIN 4 = a0 ; /\* Comparator A \*/

PIN 5 = a1 ; /\* \*/

PIN 6 = a2 ; /\* \*/

PIN 7 = b0 ; /\* Comparator B \*/

PIN 8 = b 1 ; /\* \*/

PIN 9 = b2 ;

PIN 10 = startPrt ; /\* Start Printing \*/

/\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\* OUTPUT PINS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

PIN 17 = diffCmp ; /\* XOR (A B not equal HIGH) \*/

PIN 18 = sameCmp ; /\* XNOR (A B equal HIGH) \*/

PIN 21 = q0 ; /\* output counter \*/

PIN 22 = q1 ; /\* output counter \*/

PIN 23 = q2 ;

/\*\*\*\*\* Function Comparator\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

sameCmp = !(a0$b0)&!(a1$b1)&!(a2$b2);

diffCmp = !sameCmp ;

/\*\*\*\* Function Clock Enabler \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

clkEn=startPrt & diffCmp;

/\*\*\* Function Counter 2 Bit UP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

field count =[q2..0];

$define s0 'b' 000

$define s1 'b' 001

$define s2 'b' 010

$define s3 'b' 011

$define s4 'b' 100

$define s5 'b' 101

$define s6 'b' 110

$define s7 'b' 111

count.ar=reset; /\* connect reg AR to reset (Asyn Mode) \*/

count.sp=preset; /\* connect reg AR to preset (Syn Mode) \*/

sequence count {

present s0 if clkEn next s1;

default next s0;

present s1 if clkEn next s2;

default next s1;

present s2 if clkEn next s3;

default next s2;

present s3 if clkEn next s4;

default next s3;

present s4 if clkEn next s5;

default next s4;

present s5 if clkEn next s6;

default next s5;

present s6 if clkEn next s7;

default next s6;

present s7 if clkEn next s7;

default next s7;}