REPORT FOR
MINI PROJECT DIGITAL LOGIC
(LAB 4)

SECR 1013 DIGITAL LOGIC

2019/2020 SEM 1

SECTION 02

GROUP MEMBERS:

HAM JING YI   *A19EC0048*

TEE HUI YOU   *A19EC0170*

LECTURER: DR RASHIDAH KADIR

Table of Content

Dedication & Acknowledgement

This project is dedicated to our digital logic lecturer Dr. Rashidah who have never failed to give us all the needs and explanation during the time we developed our mini project. We could say that without her explanation and determination, we could not have completed this project. To our lecturer, for always reminding us in some process during the lab 4 project, for the advices she gives us during the project, thank you. We dedicate this mini project to all the assistants who work so hard to provide helps when we in need.

We would like to thank UTM for giving us a chance to do this Xerox machine mini project as we really learn a lot from this opportunity. To our parents, we would like to thank them for supporting us in our daily lives, send us to UTM, giving us some useful advices when we face the troubles.

The Background

This mini project will implement 3 different components on a single GAL device .Those components are 3-bit Count Up Counter, 3-bit Comparator and Clock Disabler. Before doing this experiment, students have to familiarize ourselves with the WinCUPL compiler and the universal programmer.

The Problem

User will initially enter the amount of copies, the counter will count the number of copies that has been photocopied. The machine will stop once the required number of copies produced. If number of copies produced is equal to the number of required copies, the count-up counter will stop counting.
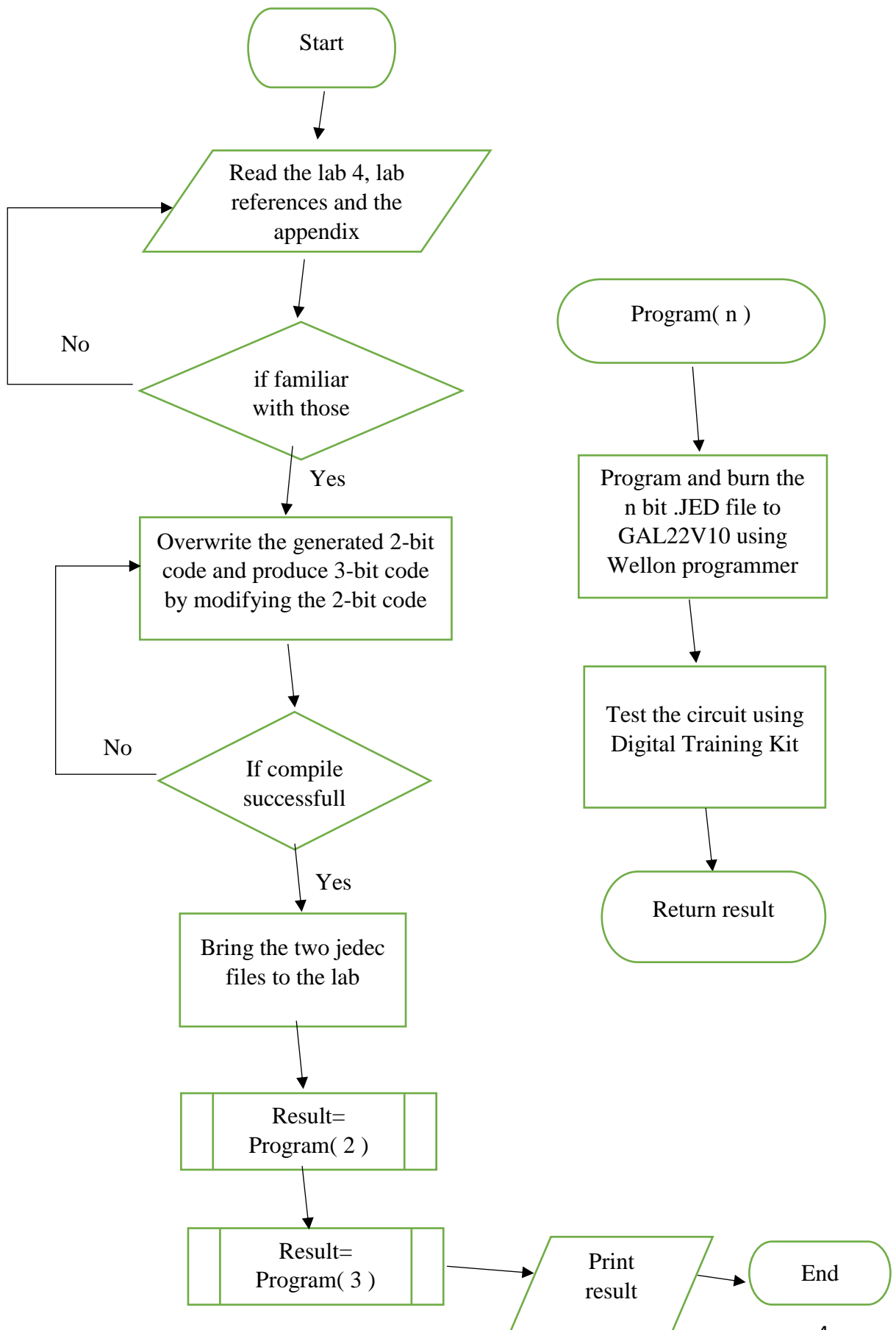
Objectives

The objectives of this laboratory are to introduce the students to the development of a PLD device and a simple Hardware Description Language.

Components

There are 4 main components such as switches, counter, comparator and clock disabler.

Flowchart

```
                    ┌─────────────┐
                    │    Start    │
                    └──────┬──────┘
                           │
                           ▼
            ╱─────────────────────────╲
           ╱   Read the lab 4, lab     ╲
          ╱   references and the        ╲
           ╲   appendix                 ╱
            ╲─────────────────────────╱
                           │
                           ▼
                    ◇──────────────◇
         No    ◇                      ◇
      ◄────────◇    if familiar       ◇
               ◇    with those        ◇
                    ◇──────────────◇
                           │ Yes
                           ▼
            ┌─────────────────────────┐
            │ Overwrite the generated  │
            │ 2-bit code and produce   │
            │ 3-bit code by modifying  │
            │ the 2-bit code           │
            └─────────────────────────┘
                           │
                           ▼
                    ◇──────────────◇
         No    ◇                      ◇
      ◄────────◇   If compile         ◇
               ◇   successfull        ◇
                    ◇──────────────◇
                           │ Yes
                           ▼
            ┌─────────────────────────┐
            │ Bring the two jedec      │
            │ files to the lab         │
            └─────────────────────────┘
                           │
                           ▼
            ┌──┬───────────────────┬──┐
            │  │   Result=         │  │
            │  │   Program( 2 )    │  │
            └──┴───────────────────┴──┘
                           │
                           ▼
            ┌──┬───────────────────┬──┐
            │  │   Result=         │  │
            │  │   Program( 3 )    │  │
            └──┴───────────────────┴──┘
```

Program(n)

Program and burn the n bit .JED file to GAL22V10 using Wellon programmer

Test the circuit using Digital Training Kit

Return result

Print result

End

Materials and Software's Used

MATERIALS

- Breadboard
- GAL22V10
- ETS-5000 Digital Training Kit
- Wellon Universal Programmer and Tester
- WinCUPL 5.0 Software

LAB REFERENCES

- "WinCUPL user manual"
- "GAL22V10 Data Sheet"
- "How to use Wellon Programmer"
- "How to use WinCUPL 5"

Breadboard is a piece of platform on the training kit provided and it is used to build and test electronic circuits. The breadboard contains a lot of holes that are arranged in row. We wired some parts of the breadboard so that electricity can be flowed from component to componenet. The GAL22V10 is a series of programmable-logic device (PLD) and it is a CMOS device. It has 12 input pins.

WinCUPL is used for typing the codes. Students are provided with a complete 2 bit Xerox machine codes and modification is needed to modify the 2 bit code into 3 bit codes. Students have to start a new project in WinCUPL, overwrite the generated code. After a successfully compilation, .pdf file, .jed file and .abs file will be created and the .jed file will be needed to program the PLD device using Wellon Programmer. After the 2-bit are successfully burned into the GAL22V10 device using Wellon Universal Programmer and Tester, the circuit has to be tested by connecting the IC to the Digital Training Kit. The processes are the same for the 3 bit codes.

Circuit Implementation
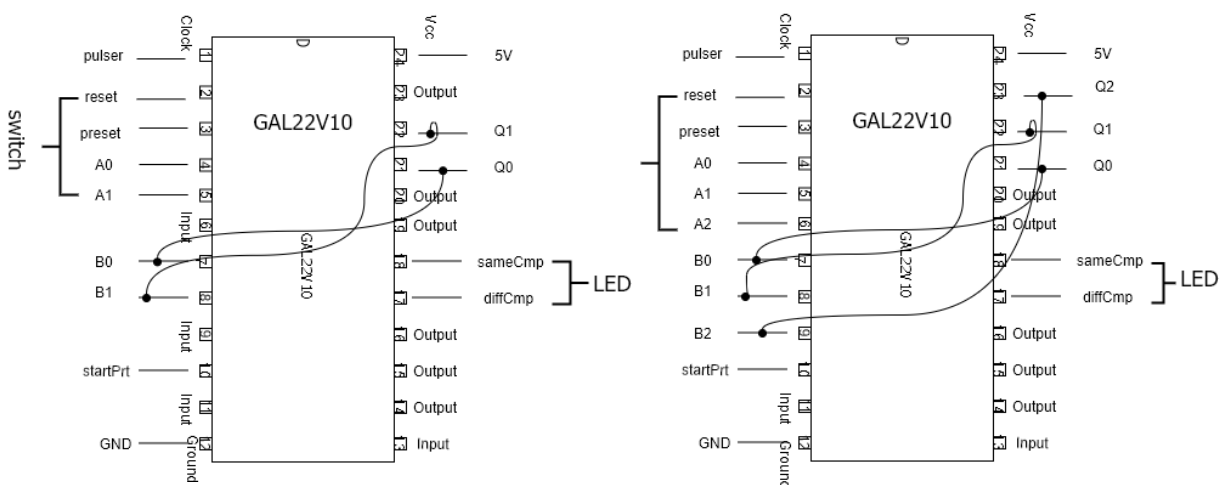
Circuit of 2-bit comparator   Circuit of 3-bit comparator

Figure 1

From Figure 1, Pin 1 of the training kit is connected to the pulser. As the pulser is pressed, there is an increment in the counter ($00\rightarrow01\rightarrow10\rightarrow11$). To implement the 2-bit counter to 2-bit counter, pin 6 is connected to switch representing A2. Then, pin 9 representing B2 is connected to pin 23 which is Q2. The increment of 3-bit counter will be in the order of $000\rightarrow001\rightarrow010\rightarrow011\rightarrow100\rightarrow101\rightarrow110\rightarrow111$.

Physical System Implementation

Figure 2 shows the circuit is connected according to the circuit implementation which is Figure 1.
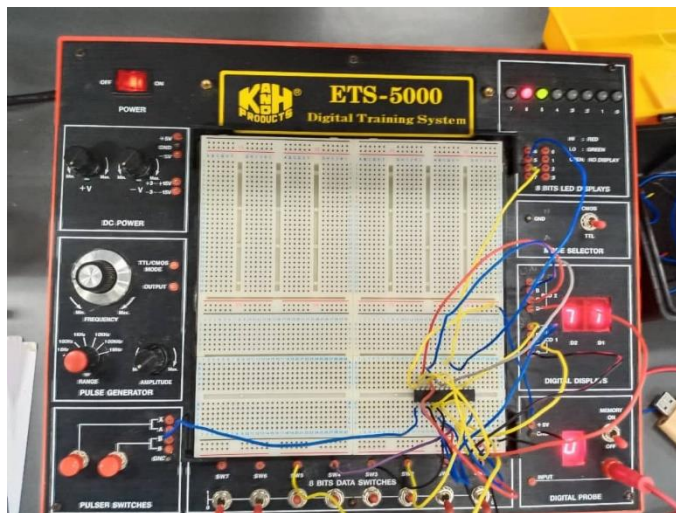
Figure 2

➢ Switches: Switches provided by ETS 5000 Digital Training Kit is used to control the input.

➢ Counter: A count up 3 bits and 2 bits counter that was controlled by clock is implemented as a WinCUPL code that will be programmed into GAL. The code is attached in the APPENDIX.

➢ Comparator: A combinational circuit that can be designed using XNOR gate, Figure 3 shows a 2-bit equality Comparator. We modified it to 3-bit equality Comparator to suit our machine requirement. The circuit is then translated to a WinCUPL code as attached in the APPENDIX.
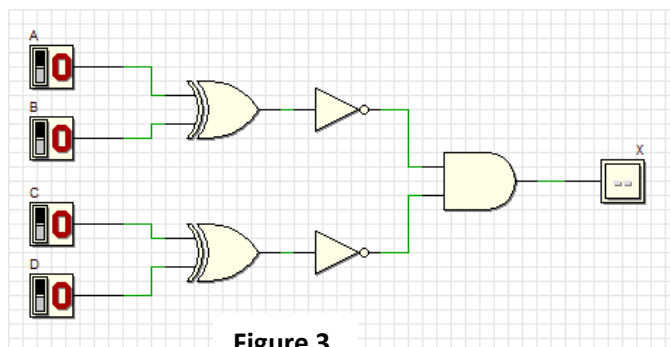

Figure 3

➢ Clock Disabler: A circuit to disable the clock signal. The code is designed to suit this function as attached in the APPENDIX.

Discussion

We learned a lot of things and gain a lot of knowledge through this mini project. After went through this lab 4 mini project, we know the development of a PLD device, and how a simple Hardware Description Language did its important role in the photocopying machine. We know that a good and functional photocopying machine really require a lot of complicated components. All the wires, switches and the components have to be in their right places in order to get a precise result. A breadboard, flip-flop, comparator, count-up counter, switch, clock disabler are just a very simple basic to build up a Xerox machine. We still have to put a lot of effort if we want a better thing. However, the comparator and count-up counter do play a crucial role, the Xerox machine may face a lot of problems in calculating and counting the amount of printed paper without the proper functioning comparator and count-up counter. 2-bit codes is a requirement to build up a 2-bit up counter and comparator. The maximum decimal number of 2-bit binary is 3. When count up to maximum 3, stop signal will be generated by the clock disabler to stop the count–up and operation of machine. So, applying the 2-bit codes to test the circuit we will get a smallest value of 0 and largest value of 3. In order to get 3-bit codes, we have to modify the 2-bit codes using WinCUPL.

Our greatest strength is our listening skill. We really have to pay full attention and stay focus about what our lecturer said to avoid careless mistakes in the project. Besides, full attention is needed in connection of wires on the breadboard, switch and the small components. Let us think about if we do it simply and careless, all connection will have a massive problem and the machine will have a messy value on the screen. With those communication skills, listening skill and problem-solving skill, we manage to build up a circuit and a correct connection between the breadboard, counter, comparator, switch, clock disabler and the LED screen. After testing the circuit with input 0 to 3 for 2-bit and 0 to 7 for 3 bit, we get a correct output on the LED screen and the high and low output of the comparator and counter. During the last part of the lab, we manage to answer all the questions given by lecturer and explain what are the functions and meaning of the green light and the red light, which refers to the comparator component.

There are a lot of problems during this project. The first problem that we face is compilation failed of the codes. There are some problems with the codes provided in the book (lab). Both of us spent a lot of time trying to figure out what is the problem and where is it. Luckily, lecturer knew it and manage to provide us codes that there is no compilation error.

We are grateful as she really help us to save a lot of time. The second problem is the connection of the wiring. There is some problems with the wires and the effect is connection will be loose. We spend some time to find a better wire from the wires provided and try to fix the problem. Moreover, we face some technical problems like wrong connection of the wiring. Sometimes, we are having a wrong decision but after some thinking and discussion, we do some correction to the wiring and fix them to the right place. Finally, we manage to get the correct output.

Our new function suggestions for improvements is basic adder. We will apply the parallel full adder as the half-adder only meant for adding 1-bit number whereas parallel full adder is used to add more than 1 bit number. In our opinion, a subtraction circuit could be built by applying the concept of 2'complement (A-B). The LED screen could be designed to have 3 columns and the first column is for the number of user entered, second is for the number of printed paper, third is for the number of paper waiting for printing. Besides, concept of encoding and decoding could be applied. Circuit is connected to a number keypad and encoder else decoder is used. We just have to press a number in the range of [0,7] for 3-bit binary number and no need to press the 3 switches. Xerox machine nowadays is improved by adding scanning, printing and even can do these stuffs online by connecting to WiFi.

Conclusion

During this mini project, communication skill is very important. We have to discuss how a component works and solve the problems like what should we do if there are some weird outputs appear on the LED screen. Our strong communication skills helped a lot in discussing and solving problems. We managed to overcome all of the problems faced throughout the mini project with the help of lecturer, and other course mates. We realised that teamwork is an undeniable value needed to complete a project. Discussion and communication is always the best way to solve any conflict faced.

Digital Logic is an important subject which provide advance exposure to Computer Science students about the components of computer. It is the basic of digital computing and provides a fundamental understanding to us about how circuits and hardware communicate and work within a computer system. It also represents the signals and sequences of transmission of data in a digital circuit through numbers. As a Computer Science student, we have to be familiar with the components of a computer system. Admittedly, Digital Logic is among the most suitable subject to prepare the fresh graduates towards the future workplace.

REFERENCES

(Wikipedia, 2019)

https://en.wikipedia.org/wiki/GAL22V10

APPENDIX

A) 2-bit Xerox System

```
Name      XeroxMachine2BIT ;
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;}
```

B) 3-bit Xerox System

```
Name      XeroxMachine2BIT;
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 ;                /* Comparator B */
PIN 7 = b0 ;                /* */
PIN 8 = b1;
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;
}
```