**BÀI CHUẨN BỊ THÍ NGHIỆM 5**

BÀI TỔNG HỢP LẮP MẠCH VÀ GIAO TIẾP VỚI FPGA

Tên: MSSV:

Ngày: Lớp:

**YÊU CẦU CHUẨN BỊ:**

**Phần I: Tạo Clock 1 Hz cho Kit DE 2**

Kit De 2 cung cấp chân CLOCK\_50 được nối tới một bộ tạo clock 50 MHz. Nếu nối module tới chân này, ta sẽ nhận được CLOCK 50 MHz (Giao động 50 triệu lần 1 giây). Trong các bài toán thực tế, ta thường thực hiện các thay đổi ở mức 1 giây (Clock 1 Hz).

Để tạo Clock 1 Hz trên Kit DE 2, hoạt động thực tế, ta có thể dung một bộ nạp thanh ghi giá trị 50M, sau đó trừ đi 1 cho mỗi clock. Sau khi hết 50 M Clock, một bộ so sánh giá trị thanh ghi với 0 sẽ cho ra kết quả 1 và cứ thế lặp lại.

Kết quả của bộ so sánh đó chính là một tính hiệu flag cho kết quả tương tự Clock 1 Hz. Nếu ta đặt các module có enable hoạt động với tính hiệu flag này, chúng sẽ hoạt động (nhảy số,…) mỗi 1 giây.

**Yêu cầu:** Sinh viên viết mã phần cứng SystemVerilog cho module Clock 1 Hz thực hiện nhấp nháy đèn LEDR0 trên Kit De2 mỗi 1 giây.

Đoạn mã System Verilog tham khảo cho module này:

///////////////////////////////////////////////

// Ho Chi Minh University of Technology

// Faculty of Electrical & Electronics Engineering

// Department of Electronics

// EE1016 - EE1010 : Ky Thuat So

//

// Lab 5 : Thiet ke he thong den giao thong

// Thuc hien giao tiep DE2 - GPIO - Breadboard - ICs/LEDs/LCDs

// Module change 50 MHz to 1 sec

//

///////////////////////////////////////////////

module secgen (

    input clk,

    input rst,

    output sec

);

///////////////////////////////////////////////

parameter CLKCONST = 50000000; //clock 50 MHz

///////////////////////////////////////////////

reg [25:0] temp; // 26-bit 67M

wire tmpzr;

assign tmpzr = temp == 0;

always\_ff @(posedge clk) begin

    if (rst)

    temp <= CLKCONST; //50M

    else if (tmpzr)

    temp <= CLKCONST;

    else

    temp <= temp - 1;

end

assign sec = tmpzr;

endmodule

**Phần II: Viết máy trạng thái đèn giao thông điều khiển**

Hệ thống đèn giao thông điều khiển được theo yêu cầu của Thí nghiệm 1 Lab 5 yêu cầu hoạt động theo một máy trạng thái.

**Yêu cầu:** Sinh viên hãy tham khảo đề bài và vẽ máy trạng thái tương ứng cho bộ điều khiển chính của hệ thống này:

**Phần III: Viết máy trạng thái đèn giao thông tự động**

Hệ thống đèn giao thông tự động theo yêu cầu của Thí nghiệm 2 Lab 5 yêu cầu hoạt động theo một máy trạng thái.

**Yêu cầu:** Sinh viên hãy tham khảo đề bài và vẽ máy trạng thái tương ứng cho bộ điều khiển chính của hệ thống này:

**Phần IV: Thành phần GPIO của Kit DE2**

Sinh viên tham khảo datasheet kit DE2 Terrasic và tìm hiểu thứ tự của GPIO\_0 và GPIO\_1, cách thức lắp đèn LED tới GPIO\_0 của kit DE2.

**Yêu cầu:** Sinh viên vẽ cách thức lắp mạch nối với GPIO\_0 của Kit DE2 tới LED (lưu ý tính thẩm mỹ)

**Phần V: LED 7 đoạn 5161BS**

Sinh viên tham khảo datasheet IC led 7 đoạn 5161BS (Cho LED Common Anode)

**Yêu cầu:** Sinh viên vẽ cách thức lắp mạch nối với GPIO\_0 của Kit DE2 tới LED 7 đoạn (lưu ý tính thẩm mỹ)

**Phần VI: IC giải mã led 7 đoạn 74LS47**

Sinh viên tham khảo datasheet IC giải mã led 7 đoạn 74LS47 (Cho LED Common Anode)

**Yêu cầu:** Sinh viên vẽ cách thức lắp mạch nối với GPIO\_0 của Kit DE2 tới IC 74LS47 tới LED 7 đoạn (lưu ý tính thẩm mỹ)

**Phần VII: Giao tiếp LCD 16x2**

Sinh viên cần tham khảo datasheet LCD 16x2, đa phần sử dụng chip điều khiển HD44780 với sơ đồ chân cắm như sau:



Hình 1 LCD 16 x 2

LCD yêu cầu sinh viên gửi các mã Hex code dưới dạng các command, trong đó bao gồm mã HEX code khởi động và mã HEX Code để thực hiện ghi / xóa nội dung

A picture containing text, crossword puzzle, receipt

Description automatically generated

**Các lệnh cho mô-đun LCD 16x2**

Mô-đun LCD 16 × 2 có một bộ hướng dẫn lệnh đặt trước. Mỗi lệnh sẽ làm cho mô-đun thực hiện một nhiệm vụ cụ thể. Các lệnh thường được sử dụng và chức năng của chúng được đưa ra trong bảng dưới đây

|  |  |
| --- | --- |
| Lệnh | Chức năng |
| 0F | LCD bật, con trỏ bật, con trỏ nhấp nháy bật |
| 01 | Xoá toàn màn hình |
| 02 | Quay về màn hình chính |
| 04 | Giảm con trỏ |
| 06 | Tăng con trỏ |
| 0E | Màn hình bật, con trỏ nhấp nháy tắt |
| 80 | Bắt con trỏ trở về vị trí đầu tiên của hàng 1 |
| C0 | Bắt con trỏ trở về vị trí đầu tiên của hàng 2 |
| 38 | Sử dụng 2 hàng và ma trận 5x7 |
| 83 | Con trỏ hàng 1 vị trí 3 |
| 3C | Kích hoạt dòng 2 |
| 08 | Tắt màn hình hiển thị và con trỏ |
| C1 | Nhảy đến dòng 2 vị trí 1 |
| OC | Bật màn hình hiển thị, tắt con trỏ |
| C2 | Nhảy đến hàng 2, vị trí 2 |

**Khởi tạo LCD**

Các bước phải được thực hiện để khởi tạo màn hình LCD được đưa ra dưới đây và các bước này là phổ biến cho hầu hết các ứng dụng.

B1: Gửi 38H đến dòng dữ liệu 8 bit để khởi tạo

B2: Gửi 0FH để bật LCD, con trỏ BẬT và con trỏ nhấp nháy ON.

B3:Gửi 06H để tăng vị trí con trỏ.

B4: Gửi 01H để xóa màn hình và trả về con trỏ.

**Đưa dữ liệu vào LCD**

Các bước để gửi dữ liệu đến mô-đun LCD được đưa ra dưới đây. Mô-đun LCD có các chân RS, R / W và E. Chính trạng thái logic của các chân này làm cho mô-đun xác định xem đầu vào dữ liệu đã cho là lệnh hay dữ liệu được hiển thị.

Đặt R / W mức thấp.

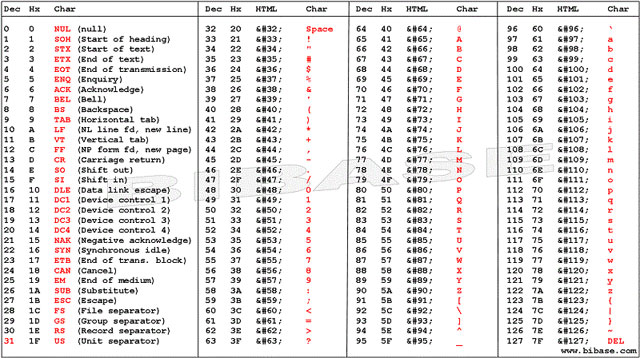
Đặt RS = 0 nếu byte dữ liệu là lệnh và tạo RS = 1 nếu byte dữ liệu là dữ liệu sẽ được hiển thị.

Đặt byte dữ liệu trên thanh ghi dữ liệu.

Xung E từ cao xuống thấp.

Lặp lại các bước trên để gửi dữ liệu khác.

Dữ liệu chữ cái và chữ số, ký hiệu sinh viên có thể tham khảo bảng mã ASCII



**Yêu cầu:** Sinh viên tìm hiểu ý nghĩa chân cắm của LCD. Sinh viên viết mã phần cứng SystemVerilog có chức năng nạp vào LCD dòng chữ “Chuan bi TN5”

Tham khảo module LCD:

///////////////////////////////////////////////

// Ho Chi Minh University of Technology

// Faculty of Electrical & Electronics Engineering

// Department of Electronics

// EE1016 - EE1010 : Ky Thuat So

//

// Lab 5 : Thiet ke he thong den giao thong

// Thuc hien giao tiep DE2 - GPIO - Breadboard - ICs/LEDs/LCDs

// module simple lcds

//

///////////////////////////////////////////////

module lcddisp\_ex(

  input clk,

  output reg lcd\_e, lcd\_rs,

  output reg [7:0] data

);

//////////////////////////////////////////////////////

//preset lcd write

integer j = 1;

reg [7:0] Datas [1:15];

always @(posedge clk) begin

Datas[1]   =  8'h38;    //-- control instruction : configure - 2 lines, 5x7 matrix --

Datas[2]   =  8'h0C;    //-- control instruction : Display on, cursor off --

Datas[3]   =  8'h06;    //-- control instruction : Increment cursor : shift cursor to right --

Datas[4]   =  8'h01;    //-- control instruction : clear display screen --

Datas[5]   =  8'h80;    //-- control instruction : force cursor to begin at first line --

Datas[6]   =  8'h54;    //-- T --

Datas[7]   =  8'h68;    //-- h --

Datas[8]   =  8'h69;    //-- i --

Datas[9]   =  8'h20;    //--   --

Datas[10]  =  8'h4E;    //-- N --

Datas[11]  =  8'h67;    //-- g --

Datas[12]  =  8'h68;    //-- h --

Datas[13]  =  8'h69;    //-- i --

Datas[14]  =  8'h65;    //-- e --

Datas[15]  =  8'h6D;    //-- m --

end

///////////////////////////////////////////////////////////

integer i = 0; //

always @(posedge clk) begin

//-- Delay for writing data

  if (i <= 1000000) begin

  i = i + 1; lcd\_e = 1;

  data = Datas[j];

  end

  else if (i > 1000000 & i < 2000000) begin

  i = i + 1; lcd\_e = 0;

  end

  else if (i == 2000000) begin

  j = j + 1; i = 0;

  end

  else i = 0;

 //-- LCD\_RS signal should be set to 0 for writing commands and to 1 for writing data

  if (j <= 5 )

    lcd\_rs = 0;

  else if (j > 5 & j< 18)

    lcd\_rs = 1;

  else if (j == 18)

    lcd\_rs = 0;

  else if (j > 27) begin

    lcd\_rs = 1;

    j = 5;

    end

  else

   lcd\_rs = 1;

  //else if (j > 27) begin

   // lcd\_rs = 1;

  ///  j = 5;

  //end

end

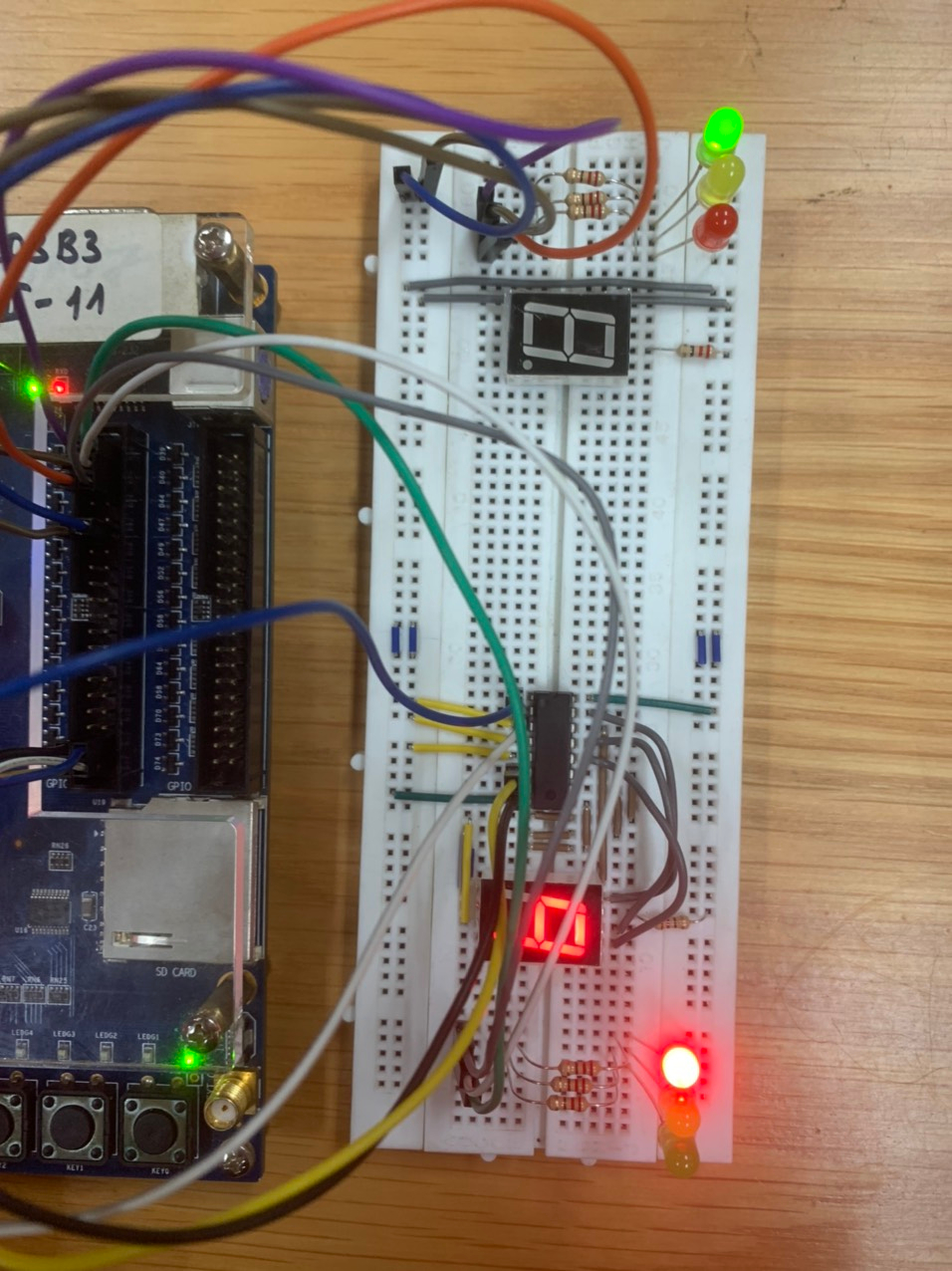
endmodule

Module LCD Sinh viên viết:

Ngoài ra, với chân Contrast Voltage, sinh viên cần lắp biến trở để điều chỉnh. Sinh viên vẽ lại sơ đồ mạch kết nối GPIO với LCD

**Phần VIII: Mạch lắp tham khảo**

Sinh viên tham khảo hình mạch lắp mẫu dưới đây để có thể làm sơ đồ lắp mạch tương tự cho thí nghiệm 1, 2 và 3



Hình Hình ví dụ 1

A picture containing text, electronics

Description automatically generated

Hình Hình ví dụ 2