

Triển khai hệ thống máy tính **DE10 - Standard Computer** trên board DE10 - Standard. Địa chỉ của các thành phần trong hệ thống được cho ở bảng bên dưới.

Thành phần	Địa chỉ
LEDR_BASE	0xFF200000
SW_BASE	0xFF200040
KEY_BASE	0xFF200050
HEX3_HEX0_BASE	0xFF200020
HEX5_HEX4_BASE	0xFF200030
SDRAM_END	0x03FFFFFF

*Lưu ý: Không sử dụng các câu lệnh nhân/chia đối với tất cả đề bên dưới.*

## ĐỀ 1

Với **N** là số **4-bit** nhập từ **SW[5:2]**, viết chương trình assembly cho hệ thống để tính

$$r16 = 1 + 3 + 5 + \dots + (2*N + 1).$$

Kết quả xuất ra **RED\_LED**.

Lưu ý: Viết chương trình con để thực hiện việc tính tổng  $r16 = 1 + 3 + 5 + \dots + (2*N + 1)$ .

## ĐỀ 2

Với **N** là số **4-bit** nhập từ **SW[3:0]**, viết chương trình assembly cho hệ thống để tính

$$r16 = 2 + 4 + 6 + \dots + (2*N).$$

Kết quả xuất ra **RED\_LED**.

Lưu ý: Viết chương trình con để thực hiện việc tính tổng  $r16 = 2 + 4 + 6 + \dots + (2*N)$ .

## ĐỀ 3

Với **N** là số **4-bit** nhập từ **SW[3:0]**, **M** là số **4-bit** được nhập từ **SW[7:4]**.  
Viết chương trình assembly cho hệ thống để tính **r16** là **ước chung lớn nhất (UCLN)** của **N** và **M**, và xuất kết quả ra **RED\_LED**.

Lưu ý: viết hàm con thực hiện công việc tính ước chung lớn nhất.

## ĐỀ 4

Với số **N** và **M** là các số **4-bit** và được nhập từ Slider Switch, trong đó **N** được nhập từ **SW[4:1]** và **M** được nhập từ **SW[8:5]**. Viết chương trình assembly cho hệ thống tính phép tính **r16 = M mod N** và xuất kết quả ra **RED\_LED**.

Lưu ý: Viết chương trình con thực hiện công việc tính **r16 = M mod N**.

## ĐỀ 5

Với **N** là số **4-bit** được nhập từ Slider Switch ở vị trí **SW[3:0]** và **M** là số **2-bit** được nhập từ **SW[5:4]**. Viết chương trình assembly tính **r16 = N^M** và xuất kết quả ra **RED\_LED**.

Lưu ý: Viết chương trình con thực hiện công việc tính **r16 = N^M**.

## ĐỀ 6

Với **N** là số **6-bit** được nhập từ Slider Switch ở vị trí **SW[5:0]** và **M** là số **3-bit** được nhập từ **SW[8:6]**. Viết chương trình assembly để thực hiện phép tính chia lấy nguyên **r16 = N / M**. Kết quả xuất ra **RED\_LED**.

Lưu ý: Viết chương trình con thực hiện công việc chia lấy nguyên **r16 = N/M**.

## ĐỀ 7

Với 2 số **a** và **b** đều **4-bit** được đọc từ Slider Switch, trong đó **a = SW[3:0]** và **b = SW[7:4]**. Viết chương trình assembly để thực hiện tính biểu thức **r16 =  $a^2 + 3b + 25$** . Xuất kết quả r16 ra RED\_LED.

Lưu ý: **Viết chương trình con thực hiện công việc nhân hai số rồi áp dụng để thực hiện biểu thức  $r16 = a^2 + 3b + 25$ .**

## ĐỀ 8

Số **N** có dạng là **0xab** được nhập từ **SW[7:0]**. Tiến hành chương trình assembly kiểm tra xem có bao nhiêu số **chia cho 3 dư 1** trong các số a,b. Kết quả xuất ra **RED\_LED**.

Lưu ý: **Viết chương trình con thực hiện kiểm tra có bao nhiêu số chia hết cho 3 dư 1.**

## ĐỀ 9

Với số **N 10-bit** được nhập từ **SW[9:0]**. Tiến hành viết chương trình assembly đếm số lần xuất hiện của dãy bit **0b1101**. Kết quả xuất ra **RED\_LED**.

Lưu ý: **Viết chương trình con thực hiện đếm số lần xuất hiện của dãy bit 0b1101.**

## ĐỀ 10

**N** là số **8-bit** nhập từ **SW[9:2]**. Viết chương trình assembly cho hệ thống để kiểm tra giá trị của **N**. Nếu **N chia hết cho 0x11** và  **$0x18 < N < 0x40$** . Nếu thỏa 1 trong 2 điều kiện thì **RED\_LED[3:0] sáng**, nếu thỏa hết các điều kiện thì **RED\_LED[7:4] sáng**, nếu không thỏa 2 điều kiện thì **RED\_LED[9:0] tắt**.

Lưu ý: **Viết chương trình con thực hiện công việc kiểm tra N có chia hết cho 0x11 không.**

## ĐỀ 11

Với **N** là số **8-bit** được nhập từ Slider Switch ở vị trí **SW[9:2]**. Viết chương trình assembly thực hiện kiểm tra xem **N** có chia hết cho **0x14** và lớn hơn **0x41** không. Nếu thỏa 2 điều kiện thì **RED\_LED[9:0]** **sáng**, nếu thỏa 1 trong 2 điều kiện thì **RED\_LED[3:0]** **sáng**, nếu không thỏa 2 điều kiện **RED\_LED[9:0]** **tắt**.

Lưu ý: Viết chương trình con **kiểm tra số N có chia hết cho 0x14 không**.

## ĐỀ 12

Viết chương trình assembly dịch chuỗi bit đang hiển thị trên **RED\_LED** sang trái nếu **KEY[1]** được bấm, dịch sang phải nếu **KEY[2]** được bấm và nếu **KEY[3]** được bấm thì đọc số từ Slider switch ghi ra **RED\_LED**.

Lưu ý: Viết chương trình con thực hiện chức năng **kiểm tra nút nhấn nào được nhấn trong 3 nút KEY[1], KEY[2], KEY[3]**.

## ĐỀ 13

Với **N** là số **8-bit** được đọc từ 8 bit cuối Slider Switch – **SW[7:0]**. Thực hiện viết chương trình assembly kiểm tra **N** có phải là số nguyên tố hay không, nếu là số nguyên tố thì toàn bộ **RED\_LED** sáng, ngược lại thì tắt.

Lưu ý: Viết hàm con thực công việc kiểm tra số nguyên tố.

## ĐỀ 14

Thực hiện viết chương trình assembly đếm xem có bao nhiêu Slider Switch được bật lên **1** trong tổng số 10 Slider Switch **SW[9:0]**. Kết quả đếm xuất ra **HEX[1:0]** trong định **thập phân (hệ 10)**.

Lưu ý: Trong bài sử dụng ít nhất một chương trình con.

## ĐỀ 15

Thực hiện viết chương trình assembly đọc giá trị A từ **SW[3:0]**, B từ **SW[7:4]**, thực hiện **A - B** và giải mã ra 3 led 7 đoạn **HEX[2:0]** dạng thập phân.

Lưu ý: Sử dụng hàm con để giải mã ra led 7 đoạn.