18200031-Trần Đăng Khoa

**Báo cáo Timer**

**Bài 1. Nếu muốn định thời ở khoảng thời gian 500 ms giây và 100 ms giây thì PRx sẽ bằng bao nhiêu. Cho F = 4 Mhz và Prescale tự chọn. Viết code C tương ứng.**

Áp dụng công thức: **Tdelay = (PRx × Prescale) ÷ F(Hz)**

Tdelay = 0.5s Prescale = 64 và F = Fosc/2 = 4 Mhz thì **PRx= 31250**.

**Code:**

// CONFIG2

#pragma config POSCMOD = HS // Primary Oscillator Select (HS Oscillator mode selected)

#pragma config OSCIOFNC = OFF // Primary Oscillator Output Function (OSC2/CLKO/RC15 functions as CLKO (FOSC/2))

#pragma config FCKSM = CSDCMD // Clock Switching and Monitor (Clock switching and Fail-Safe Clock Monitor are disabled)

#pragma config FNOSC = PRI // Oscillator Select (Primary Oscillator (XT, HS, EC))

#pragma config IESO = ON // Internal External Switch Over Mode (IESO mode (Two-Speed Start-up) enabled)

// CONFIG1

#pragma config WDTPS = PS32768 // Watchdog Timer Postscaler (1:32,768)

#pragma config FWPSA = PR128 // WDT Prescaler (Prescaler ratio of 1:128)

#pragma config WINDIS = ON // Watchdog Timer Window (Standard Watchdog Timer enabled,(Windowed-mode is disabled))

#pragma config FWDTEN = ON // Watchdog Timer Enable (Watchdog Timer is enabled)

#pragma config ICS = PGx2 // Comm Channel Select (Emulator/debugger uses EMUC2/EMUD2)

#pragma config GWRP = OFF // General Code Segment Write Protect (Writes to program memory are allowed)

#pragma config GCP = OFF // General Code Segment Code Protect (Code protection is disabled)

#pragma config JTAGEN = OFF // JTAG Port Enable (JTAG port is disabled)

#include <xc.h>

#include <p24fj128ga010.h>

void Timer1\_Init(int period, int prescale){

T1CON = 0x00;

TMR1 = 0x00;

PR1 = period;

T1CONbits.TCS=0; //dùng clock nội (Fosc/2)

T1CONbits.TSYNC =0 ; // không đồng bộ clock ngoài

T1CONbits.TGATE=0; //không dùng GATE time

T1CONbits.TCKPS=prescale; //prescale

T1CONbits.TON=1;

}

int main() {

TRISA = 0;

PORTA = 0;

Timer1\_Init(31250,2); // PRx= 31250 ; Prescale = 64

while (1)

{

if(TMR1 == PR1){

PORTA=~PORTA;

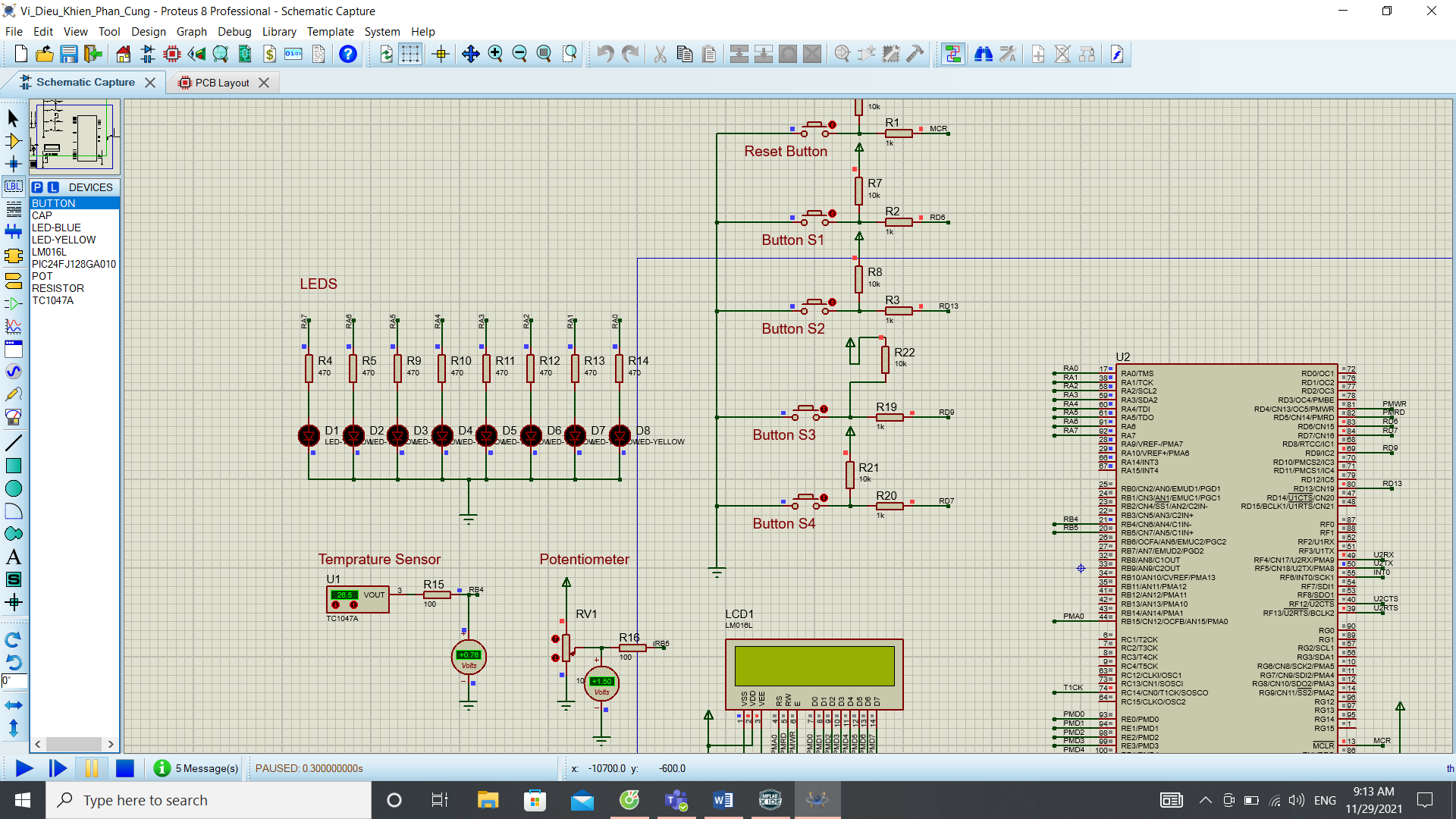
TMR1=0;

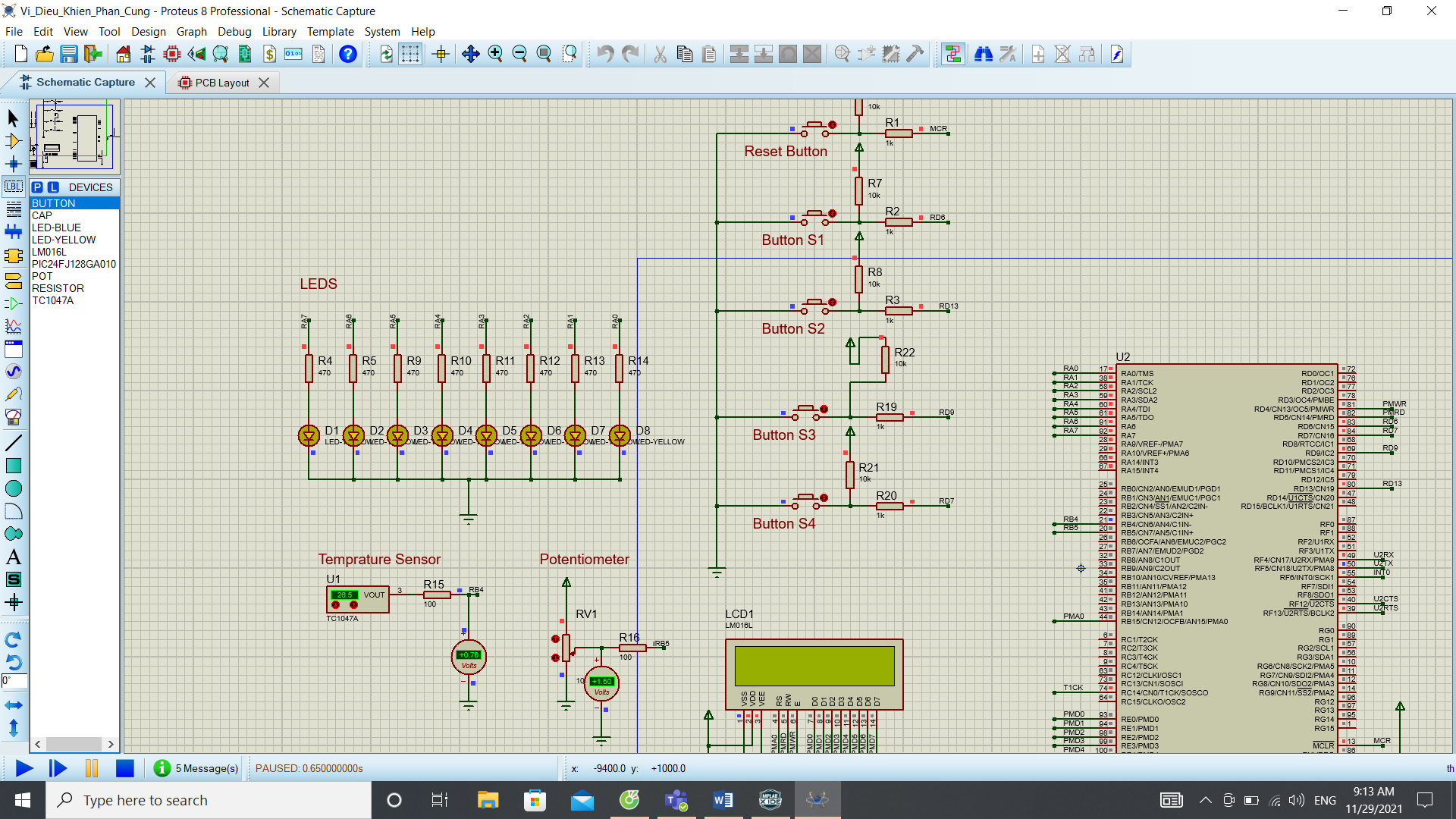
}

}

return 0;

}





**Tdelay = 0.1s Prescale = 64 và F = Fosc/2 = 4 Mhz thì PRx= 6250.**

/\*CONFIG như trên \*/

void Timer1\_Init(int period, int prescale){

T1CON = 0x00;

TMR1 = 0x00;

PR1 = period;

T1CONbits.TCS=0;

T1CONbits.TSYNC =0 ;

T1CONbits.TGATE=0;

T1CONbits.TCKPS=prescale;

T1CONbits.TON=1;

}

int main() {

TRISA = 0;

PORTA = 0;

Timer1\_Init(6250,2);

while (1)

{

if(TMR1 == PR1){

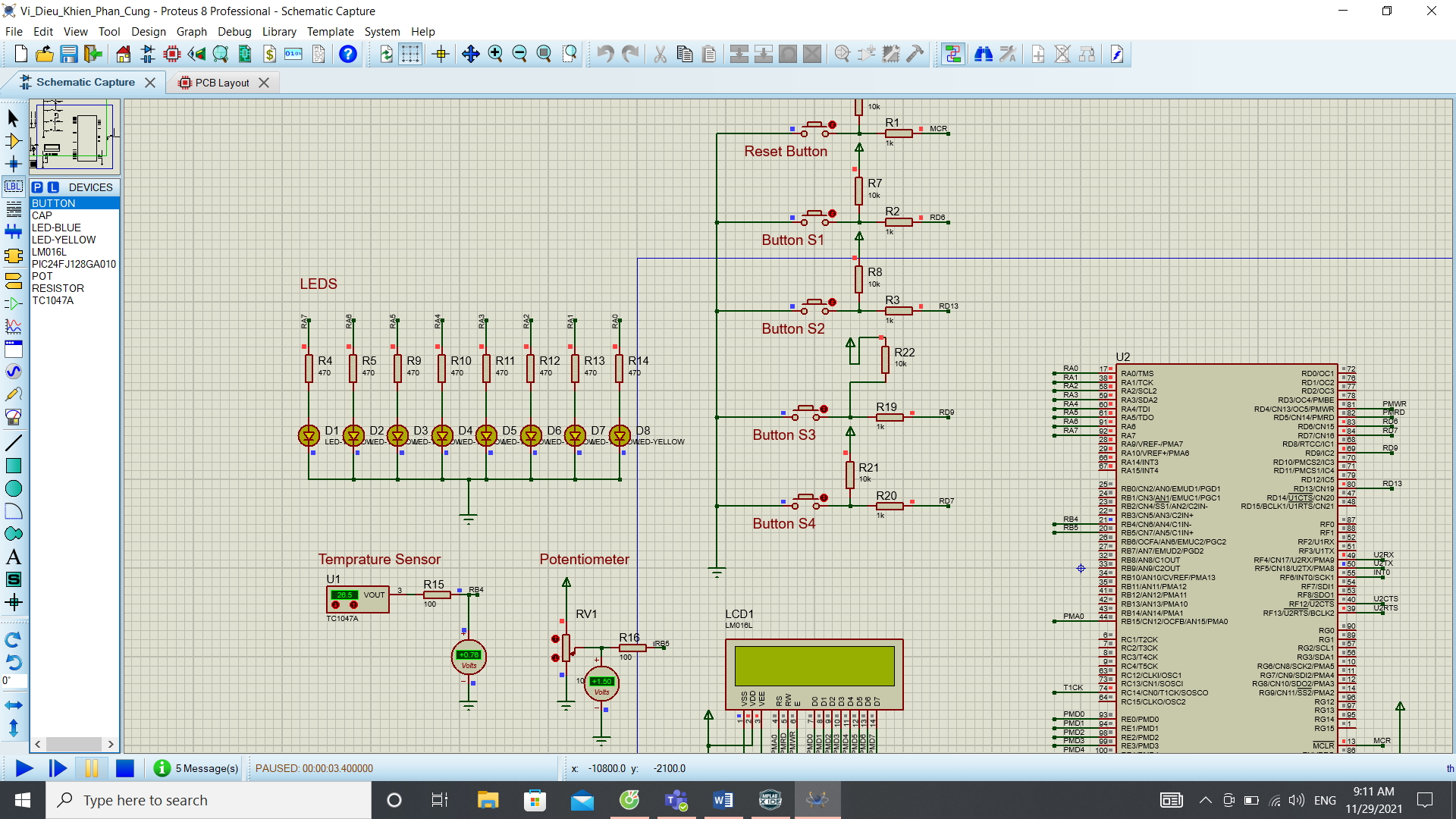
PORTA=~PORTA;

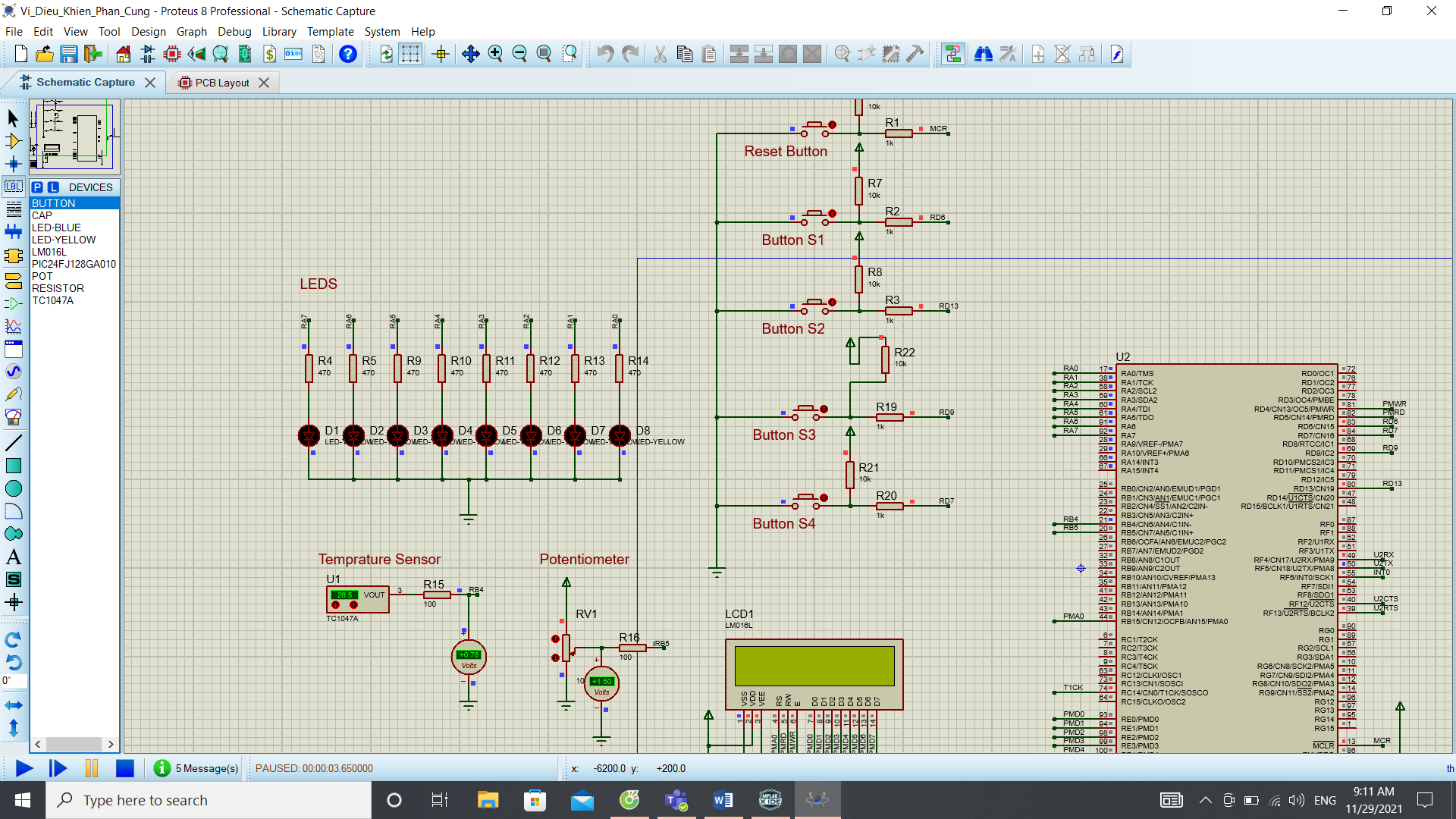
TMR1=0;

}

}

return 0;

} 



**Bài 2. Lập trình cho hệ thống thỏa mãn yêu cầu sau: Khi nhấn S1 1 lần thì LED7 chớp tốc độ 1s, nhấn 2 lần thì chớp tốc độ 1/2s, …, nhấn 8 lần thì chớp tốc độ 1/8s, nếu nhấn 9 lần thì quay lại trạng thái ban đầu.**

**Code:**

void Timer1\_Init(int period, int prescale){

T1CON = 0x00;

TMR1 = 0x00;

PR1 = period;

T1CONbits.TCS=0;

T1CONbits.TSYNC =0 ;

T1CONbits.TGATE=1;

T1CONbits.TCKPS=prescale;

T1CONbits.TON=1;

}

int main() {

TRISA = 0;

PORTA = 0;

TRISD = 0xffff; //PORTD là input

int PRx = 62500;

int count =0 ; // biến đếm

while (1)

{

if(TMR1 == PR1){

PORTA=~PORTA;

TMR1=0;

}

if ((PORTD & 0x40) == 0) // s1:RD6 (khi nhấn)

{

while ((PORTD & 0x40) == 0); // (nhấn giữ)

count++; // biến đếm cộng 1

PRx = 62500/count; // thay đổi thời gian chớp tắt led

if(count==9)count=0; // quay về T = 1s

Timer1\_Init(PRx,2); gọi hàm Timer1 với PRx đã thay đổi

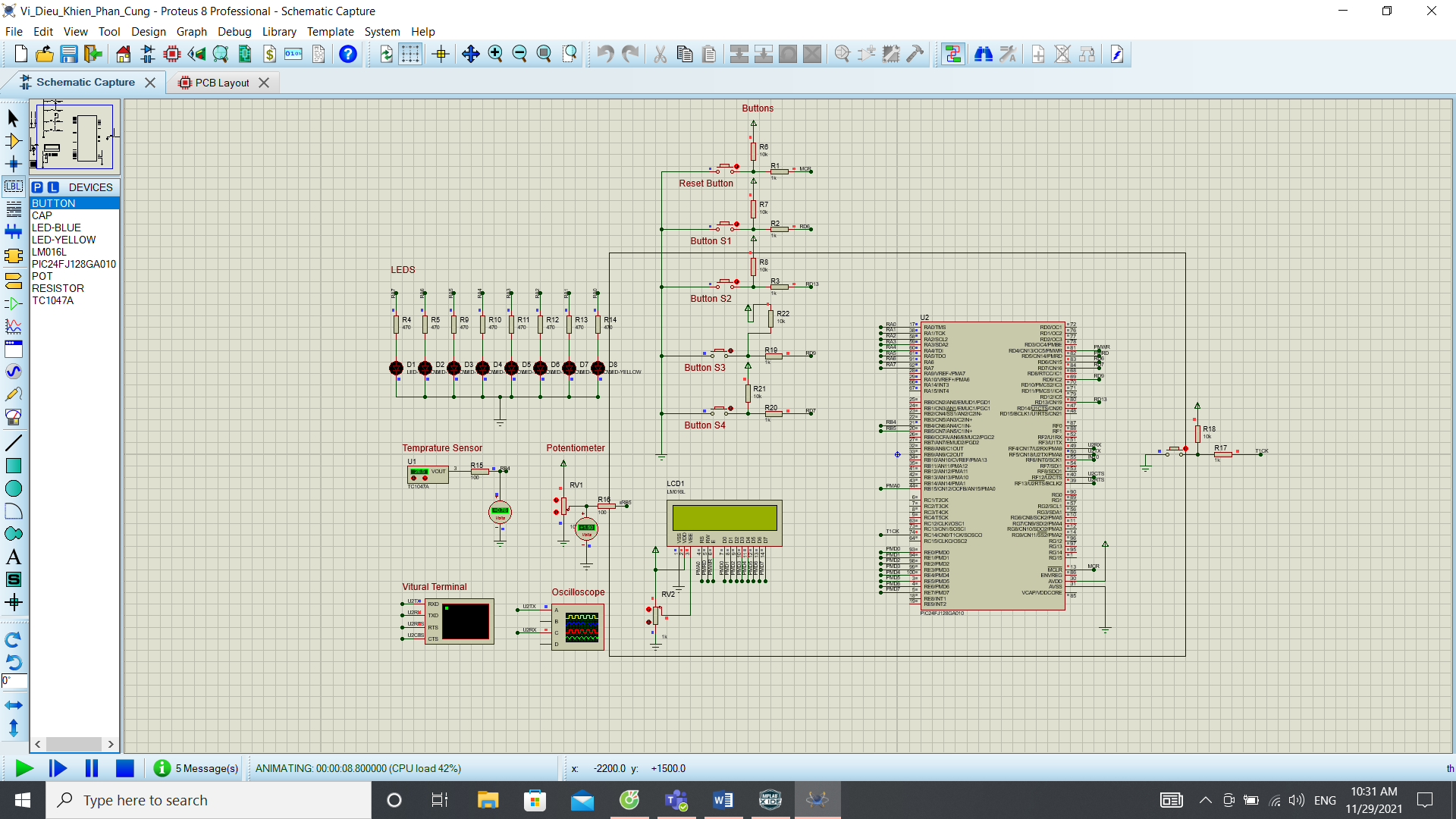
}

}

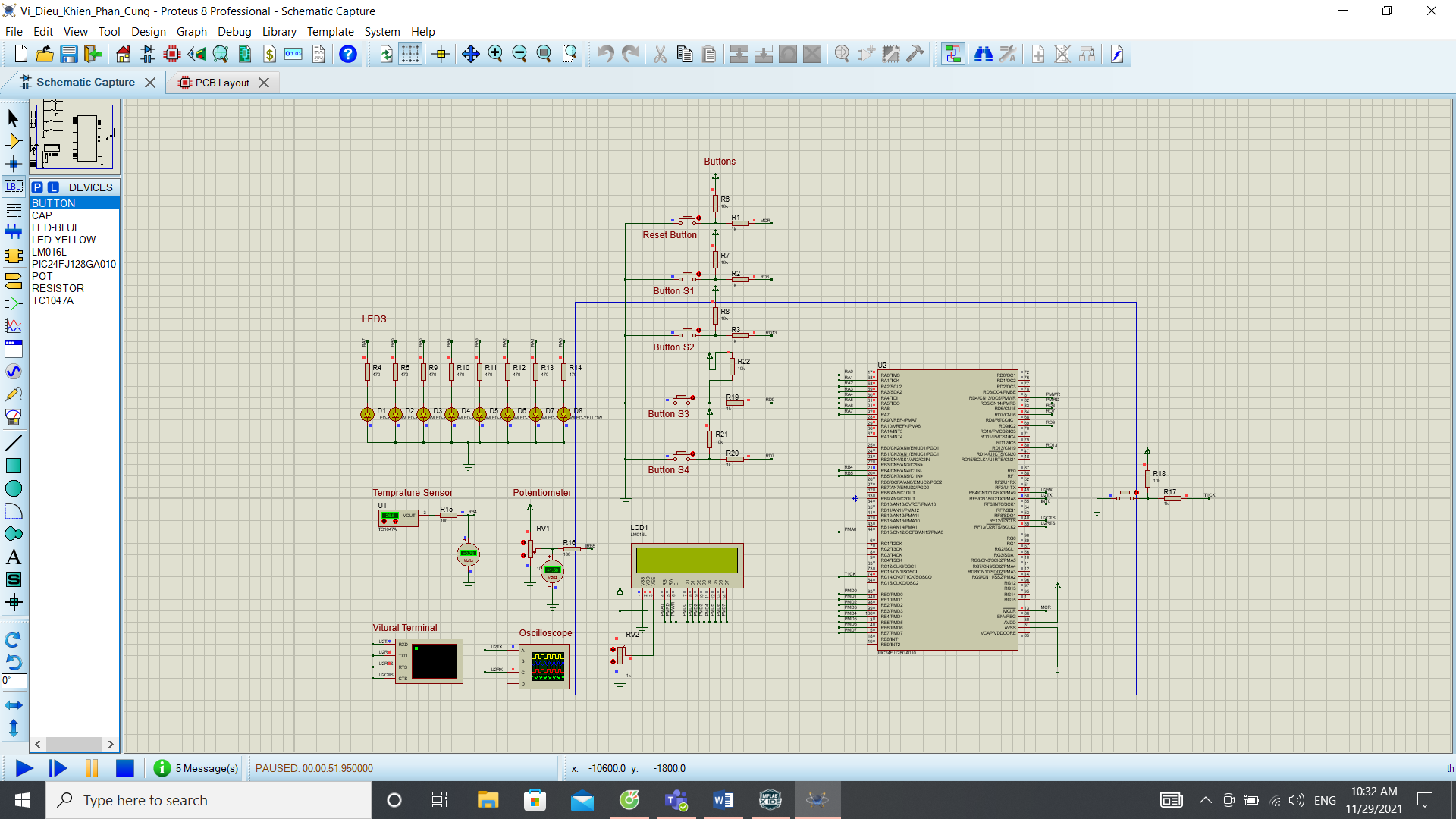
return 0;

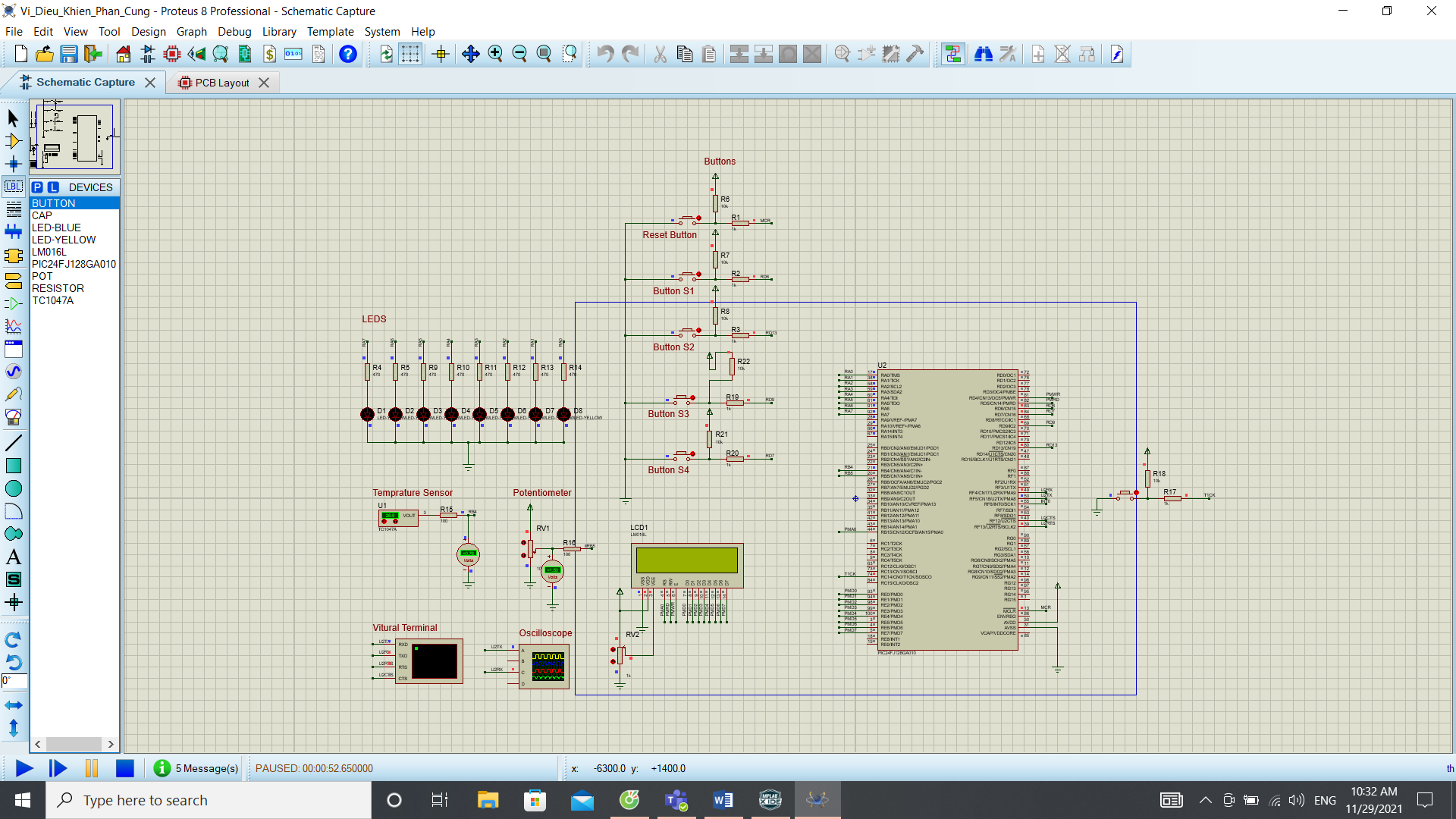
}

Ban đầu LED tắt hết:

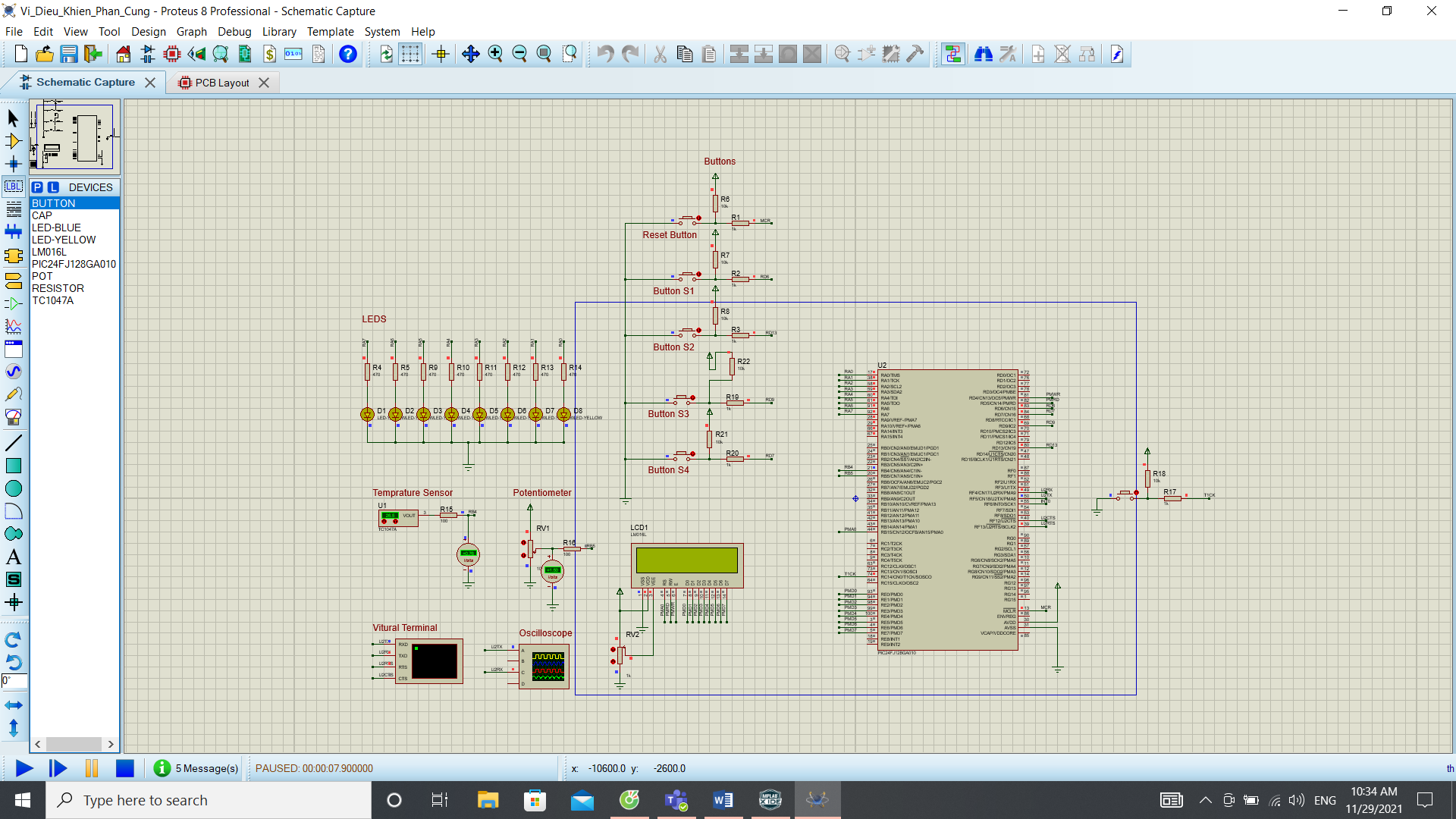


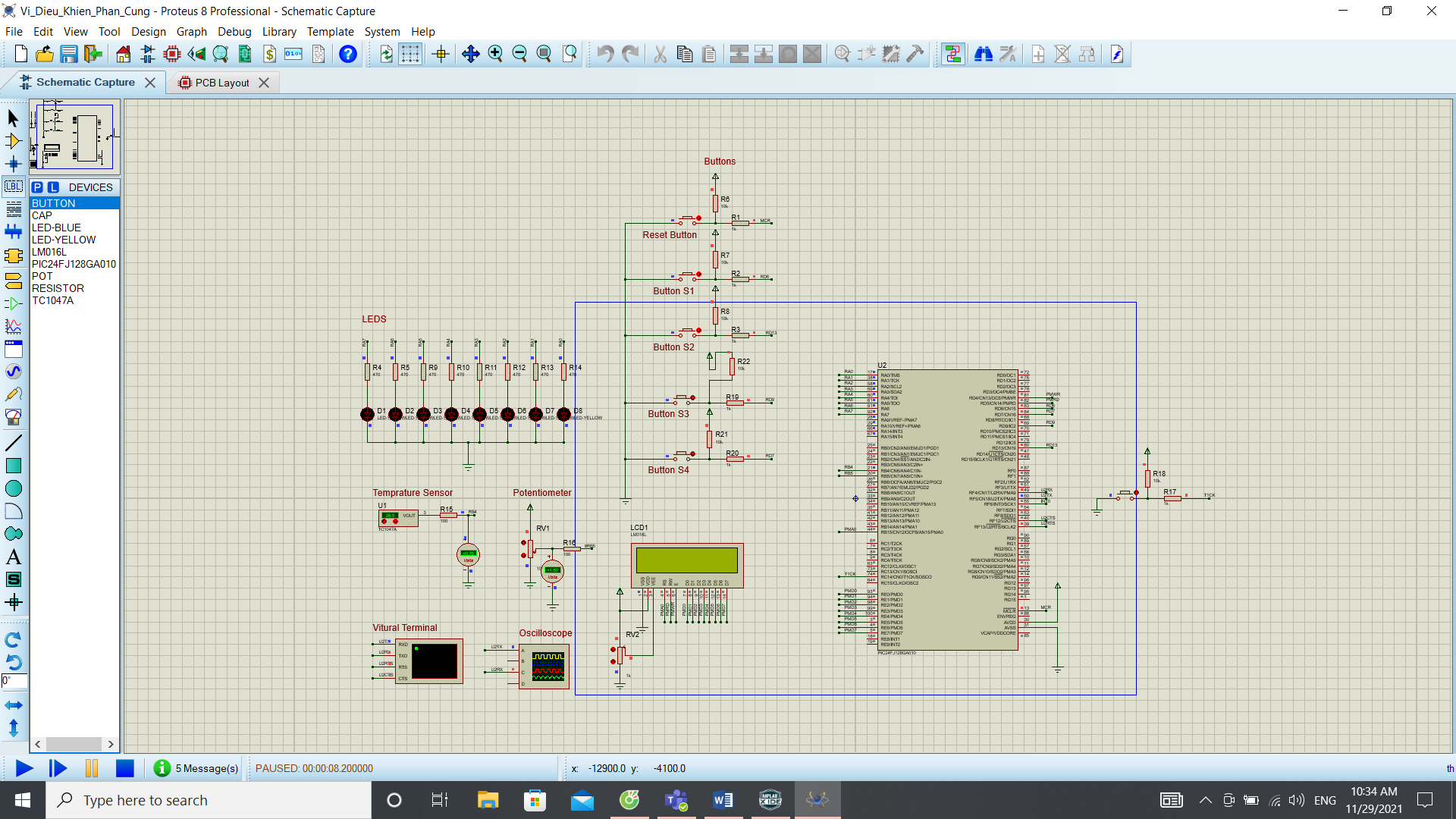
Sau khi nhấn S1 1 lần thì chớp tắt 1s:





Sau khi nhấn S1 2 lần thì chớp tắt 0.5s:





**Bài 3. Tạo thêm nút nhấn S5 và kết nối với chân TxCK. Lập trình cho hệ thống thoả mãn yêu cầu sau: đếm thời gian nhấn nút S5, nếu thời gian nhấn nút lớn hơn 3 giây, LED sáng, nếu thời gian nhấn nút lớn hơn 10 giây thì LED chớp tắt với tần số 0.5 giây.**

**Code:**

void Timer1\_Init(int period, int prescale){

T1CON = 0x00;

TMR1 = 0x00;

PR1 = period;

T1CONbits.TCS=0;

T1CONbits.TSYNC =0;

T1CONbits.TGATE=1;

T1CONbits.TCKPS=prescale;

T1CONbits.TON=1;

}

int main() {

TRISA = 0;

PORTA = 0;

int count=0;

int PRx = 31250; T =0.5s

Timer1\_Init(PRx,2);

while (1)

{

if(TMR1 == PR1){

//PORTA++;

count++;

if(count==3)PORTA=0xff; //nhấn 3s thì led sáng

if(count>10) //nhấn hơn 10s thì led chớp tắt 0.5s

{

PORTA=~PORTA;

}

TMR1=0;

}

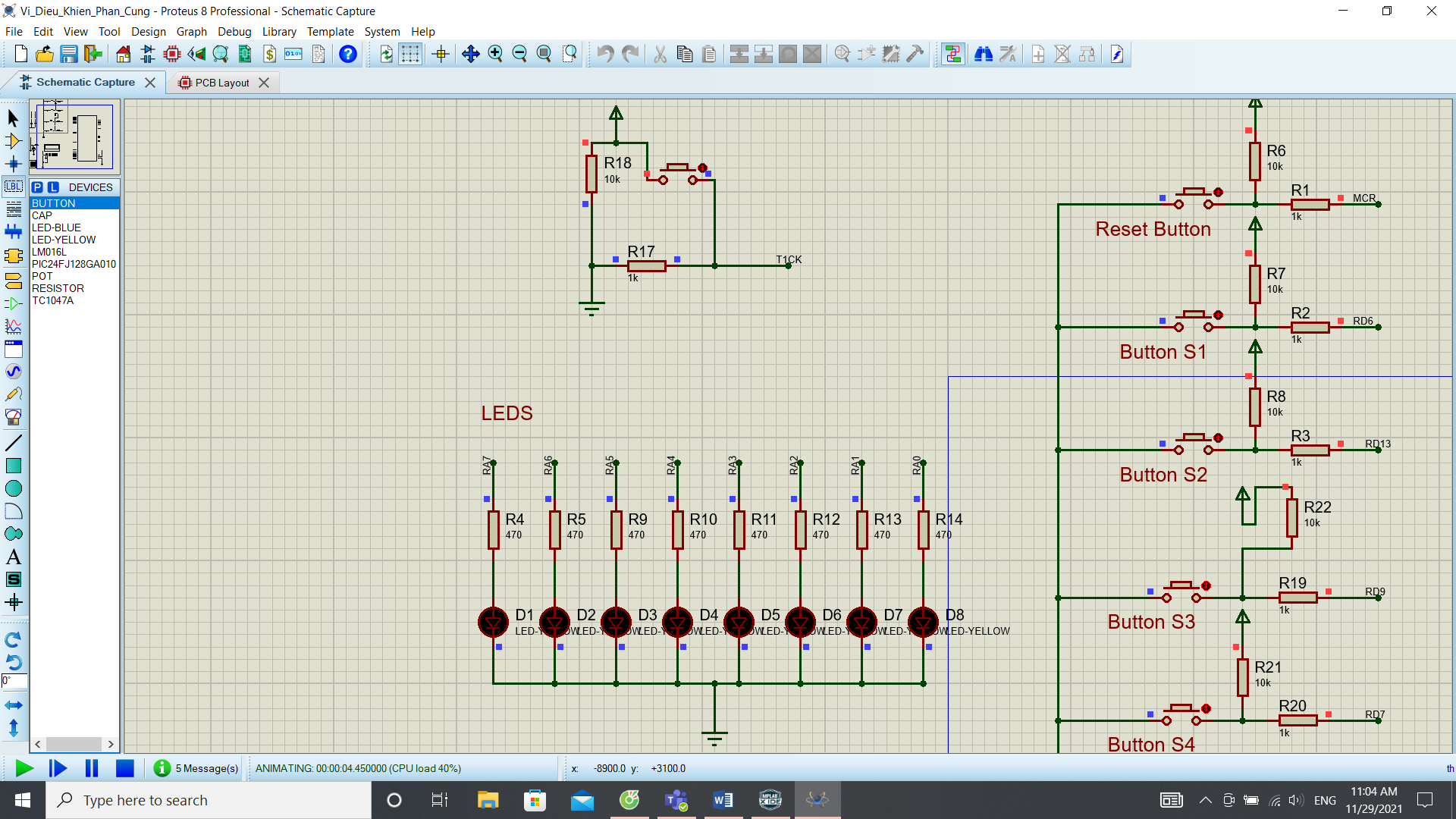
}

return 0;

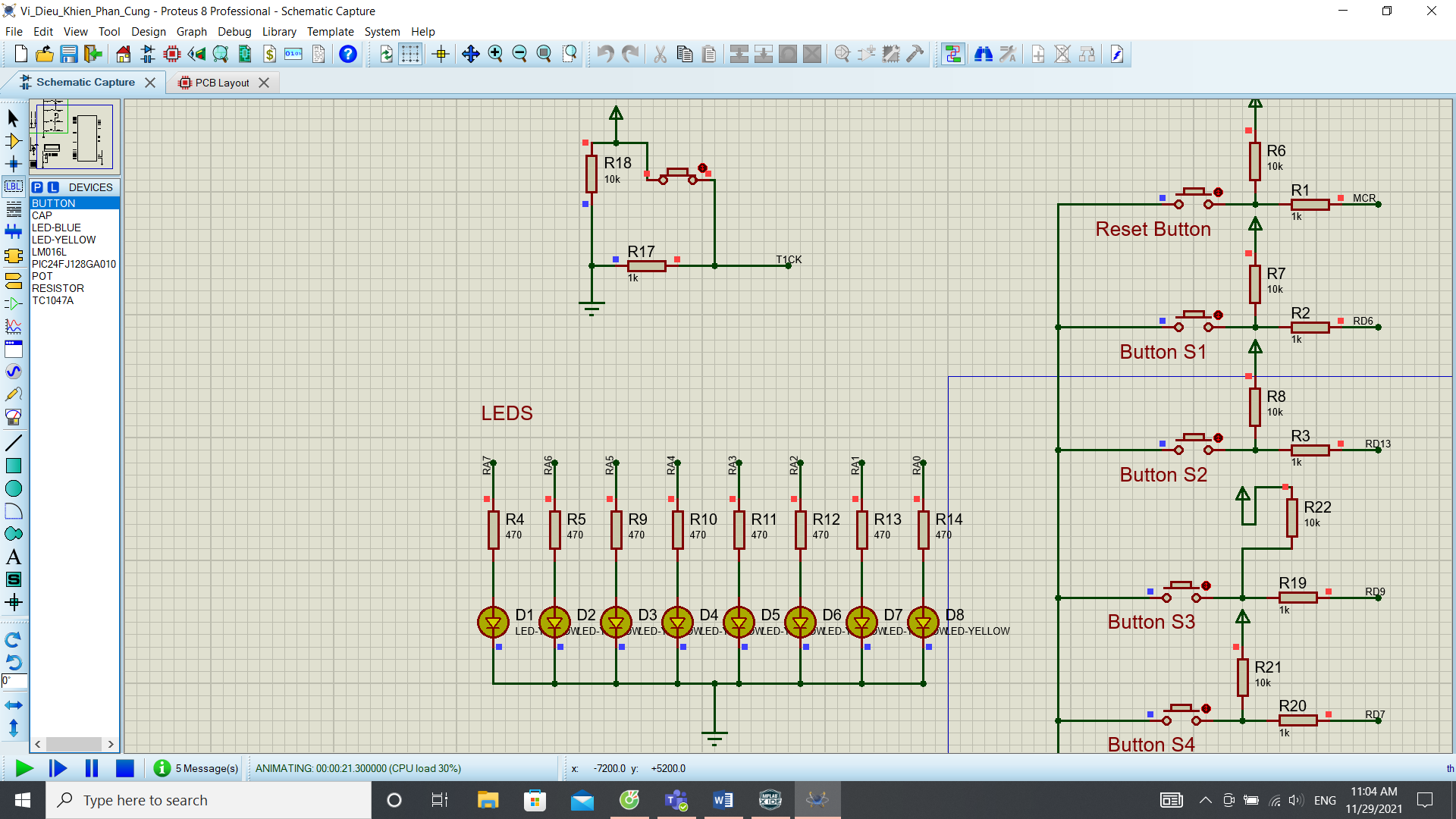
}

* **Khi nhấn trên 10s và thả:**
* Nếu thả lúc led sáng thì led tiếp tục sáng
* Nếu thả lúc led tắt thì led tiếp tục tắt
* Do bit T1CK đang ở mức 0
* **Khi tiếp tục nhấn giữ thì led tiếp tục chớp tắt:**
* Do biến count trong code vẫn tiếp tục đếm thời gian nhấn.

Lúc chưa nhấn



Lúc nhấn >= 3s



Lúc hơn 10s

