```c
/*
* Welding Visual Effect Version 1.2
* COPYRIGHT © 2013 S.D. "Hoffy" Hofmeister
* http://www.hoffysworld.com
* Provided under a Creative Commons Attribution, Non-Commercial Share Alike,3.0 Unported License
* TARGETED TO MSP430 LANUCHPAD W/MSP430G2553 PROCESSOR
**************************************************************************************************
* Design Notes:
*
* This code uses a random number between 0 and and the defined count
* to select between 10 defined effects for two high brightness LEDs.
*
* This has been designed to use two high brightness white light LEDs in order to make the effect
* more intense but you can operate with one of you choose.
*
* I have purposely left room for a lot of growth and improvement on this project,  have fun!
***************************************************************************************************
* Circuit Pinout:
* PIN 1.0 = Anode of LED #1   > Cathode to Ground
* PIN 1.1 = UNASSIGNED - UART
* PIN 1.2 = UNASSIGNED - UART
* PIN 1.3 = Anode of LED #2 > Cathode to Ground
* PIN 1.4 = UNASSIGNED
* PIN 1.5 = UNASSIGNED
* PIN 1.6 = UNASSIGNED
* PIN 1.7 = UNASSIGNED
* PIN 2.0 = UNASSIGNED
* PIN 2.1 = UNASSIGNED
* PIN 2.2 = UNASSIGNED
* PIN 2.3 = UNASSIGNED
* PIN 2.4 = UNASSIGNED
* PIN 2.5 = UNASSIGNED
***************************************************************************************************
**************************************************************************************************/
#include <msp430g2553.h>
#include <stdlib.h>

volatile long reps = 0;
volatile long t = 0;
unsigned short int res;
//-----------------------------------
// Effect Adjustments

// Modify these numbers to alter effect
// XXXXXXXXX_rep = number of flashes
// gapX    = How long LEDs are on
// gapXa   = How long LEDs are off

// Original value comments have been added so you can restore them to the original value

unsigned short int count = 50;  // (must remain higher than 10)  Low number effect more often,
higher less often
// number higher than 10 creates a greater number ofgaps an pauses between the effects

volatile long one_rep = 20;     //20
int gap1 = 500;             //500
int gap1a = 1000;               //1000

volatile long two_rep = 5;      //5
int gap2 = 500;             //500
int gap2a = 1000;               //1000

volatile long three_rep = 10;   //10
int gap3 = 500;             //500
int gap3a = 1000;               //1000

volatile long four_rep = 15;        //15
int gap4 = 250;             //250
int gap4a = 500;                //500

volatile long five_rep = 20;        //10
int gap5 = 500;             //500
int gap5a = 1000;               //500

volatile long six_rep = 5;      //5
int gap6 = 250;             //250
int gap6a = 500;                //500

volatile long seven_rep = 5;    //5
int gap7 = 250;             //250
int gap7a = 250;                //500

volatile long eight_rep = 10;   //10
int gap8 = 500;             //500
int gap8a = 500;                //500

volatile long nine_rep = 20;    //20
int gap9 = 250;             //250
int gap9a = 500;                //500

volatile long ten_rep = 5;      //5
int gap10 = 500;                //500
int gap10a = 500;               //500
//----------------------------------------------
/*
 * Main Code
 */
void main(void){

    // Configure Pins and stop Watchdog

WDTCTL = WDTPW + WDTHOLD;  //Stop Watchdog Timer

P1DIR |= BIT0;  //Set Port 1 Pin 0 as Output
P1OUT &= ~BIT0; //Set Port 1 Pin 0 as OFF

P1DIR |= BIT3; //Set Port 1 Pin 3 as Output
P1OUT &= ~BIT3;//Set Port 1 Pin 3 as OFF
```

```c

//---------------------------------------------
while ( 1 ) { // Continuous Loop

res = (rand() % count); // returns random value

switch (res){  //  switches between each of the ten choices based on the returned number above

                case 1:
                       for (reps; reps < one_rep; reps++) // repeat choice based on the number of reps set above
                       {
                       P1OUT |= BIT0;  //ON
                       P1OUT |= BIT3;  //ON

                       for (t; t < gap1; t++)   //delay loop
                       {
                       }

                       P1OUT &= ~BIT0; //OFF
                       P1OUT &= ~BIT3; //OFF
                       t=0;  //RESET COUNTER

                       for (t; t < gap1a; t++)      //delay loop
                       {
                       }
                       }
                       t=0; //RESET COUNTER
                       reps=0; //RESET REPS
                       break; // Leave choice and get new random number

                case 2:
                        for (reps; reps < two_rep; reps++) // repeat choice based on the number of reps set above
                        {
                        P1OUT |= BIT0; //ON
                        P1OUT |= BIT3; //ON

                        for (t; t < gap2; t++)      //delay loop
                        {
                        }


                        P1OUT &= ~BIT0; //OFF
                        P1OUT &= ~BIT3; //OFF
                        t=0; //RESET COUNTER
                        for (t; t < gap2a; t++)     //delay loop
                        {
                        }

                        }
                        t =0; //RESET COUNTER
                        reps=0; //RESET REPS
                        break; // Leave choice and get new random number


                case 3:
```

```c
                for (reps; reps < three_rep; reps++) // repeat choice based on the number of reps set above
                {
                P1OUT |= BIT0; //ON
                P1OUT |= BIT3; //ON

                for (t; t < gap3; t++)      //delay loop
                {
                }

                P1OUT &= ~BIT0; //OFF
                P1OUT &= ~BIT3; //OFF
                t=0; //RESET COUNTER

                for (t; t < gap3a; t++)     //delay loop
                {
                }
                }
                t=0; //RESET COUNTER
                reps=0; //RESET REPS
                break; // Leave choice and get new random number


        case 4:
               for (reps; reps < four_rep; reps++) // repeat choice based on the number of reps set above
               {
               P1OUT |= BIT0; //ON
               P1OUT |= BIT3; //ON

               for (t; t < gap4; t++)   //delay loop
               {
               }

               P1OUT &= ~BIT0; //OFF
               P1OUT &= ~BIT3; //OFF
               t=0; //RESET COUNTER

               for (t; t < gap4a; t++)      //delay loop
               {
               }
               }
               t=0; //RESET COUNTER
               reps=0; //RESET REPS
               break; // Leave choice and get new random number


        case 5:
               for (reps; reps < five_rep; reps++) // repeat choice based on the number of reps set above
               {
               P1OUT |= BIT0; //ON
               P1OUT |= BIT3; //ON

               for (t; t < gap5; t++)   //delay loop
               {
               }
```

```

                       P1OUT &= ~BIT0; //OFF
                       P1OUT &= ~BIT3; //OFF
                       t=0; //RESET COUNTER

                       for (t; t < gap5a; t++)      //delay loop
                       {
                       }
                       }
                       t=0; //RESET COUNTER
                       reps=0; //RESET REPS
                       break; // Leave choice and get new random number


                case 6:
                       for (reps; reps < six_rep; reps++) // repeat choice based on the number of reps set above
                       {
                       P1OUT |= BIT0; //ON
                       P1OUT |= BIT3; //ON

                       for (t; t < gap6; t++)   //delay loop
                       {
                       }

                       P1OUT &= ~BIT0; //OFF
                       P1OUT &= ~BIT3; //OFF
                       t=0; //RESET COUNTER

                       for (t; t < gap6a; t++)      //delay loop
                       {
                       }
                       }
                       t=0; //RESET COUNTER
                       reps=0; //RESET REPS
                       break; // Leave choice and get new random number


                case 7:
                       for (reps; reps < seven_rep; reps++) // repeat choice based on the number of reps set above
                       {
                       P1OUT |= BIT0; //ON
                       P1OUT |= BIT3; //ON

                       for (t; t < gap7; t++)   //delay loop
                       {
                       }

                       P1OUT &= ~BIT0; //OFF
                       P1OUT &= ~BIT3; //OFF
                       t=0; //RESET COUNTER

                       for (t; t < gap7a; t++)      //delay loop
                       {
                       }
                       }
```

```c
               t=0; //RESET COUNTER
               reps=0; //RESET REPS
               break; // Leave choice and get new random number


        case 8:
               for (reps; reps < eight_rep; reps++) // repeat choice based on the number of reps set above
               {
               P1OUT |= BIT0; //ON
               P1OUT |= BIT3; //ON

               for (t; t < gap8; t++)   //delay loop
               {
               }

               P1OUT &= ~BIT0; //OFF
               P1OUT &= ~BIT3; //OFF
               t=0; //RESET COUNTER

               for (t; t < gap8a; t++)      //delay loop
               {
               }
               }
               t=0; //RESET COUNTER
               reps=0; //RESET REPS
               break; // Leave choice and get new random number


        case 9:
               for (reps; reps < nine_rep; reps++) // repeat choice based on the number of reps set above
               {
               P1OUT |= BIT0; //ON
               P1OUT |= BIT3; //ON

               for (t; t < gap9; t++)   //delay loop
               {
               }

               P1OUT &= ~BIT0; //OFF
               P1OUT &= ~BIT3; //OFF
               t=0; //RESET COUNTER

               for (t; t < gap9a; t++)      //delay loop
               {
               }
               }
               t=0; //RESET COUNTER
               reps=0; //RESET REPS
               break; // Leave choice and get new random number


        case 10:
                for (reps; reps < ten_rep; reps++) // repeat choice based on the number of reps set above
                {
                        P1OUT |= BIT0; //ON
                        P1OUT |= BIT3; //ON

                        for (t; t < gap10; t++)     //delay loop
                        {
                        }

                        P1OUT &= ~BIT0; //OFF
                        P1OUT &= ~BIT3; //OFF
                        t=0; //RESET COUNTER

                        for (t; t < gap10a; t++)    //delay loop
                        {
                        }
                        }
                        t=0; //RESET COUNTER
                        reps=0; //RESET REPS
                        break; // Leave choice and get new random number

}  // END switch
}  //END while
}  //END main

```