```
/*
 * Stepper Motor Traverser Control (pushbutton and led panel)
 * This programme drives a unipolar or bipolar stepper motor for use with a model rai track
traverser.
 *
 * Created 19 November 2022
 * Amended 2 January 2023
 * Amended 17 January 2023
 */

  //  Libraries required

#include<Stepper.h>

  //  define motor characteristics

const int stepsPerRevolution = 200;
Stepper DMProject(stepsPerRevolution,8,9,10,11);
//const float trackspace = 118.125; // steps between tracks = 5.4/32*700
const float trackspace = 116.25; // steps between tracks = 5.4/32*700

  //  define arrays and variables

const int ledPin[7] = {22,24,26,28,30,32,34};
const int buttonPin[7] = {50,48,46,44,42,40,38};
const int location[7] = {-3*trackspace,-2*trackspace,-
1*trackspace,0,1*trackspace,2*trackspace,3*trackspace};

  int currentbuttonState[7] = {0,0,0,1,0,0,0};
  int currentledState[7] = {0,0,0,1,0,0,0};
  int nextlocation[7] ={0,0,0,0,0,0,0};
  int nextbuttonState[7] = {0,0,0,0,0,0,0};
  int nextledState[7] = {0,0,0,0,0,0,0};
  int currentlocation = 3;


void setup()
{

  // initialize the serial port
  Serial.begin(9600);
  DMProject.setSpeed(20); // recheck in situ

  //initialise led and button pins
  for (int p=0;p<7;p++)
  {
    pinMode(ledPin[p], OUTPUT);
  pinMode(buttonPin[p], INPUT_PULLUP);
  digitalWrite(ledPin[p],0);
 }

}




void loop()
{
  Serial.println("START LOOP");


 int testposition;
 int origin = currentlocation;

 int b; // location variable (button loop counter)

   Serial.print("old currentlocation = ");
   Serial.println(currentlocation);

   // initialise leds
   digitalWrite(ledPin[currentlocation],1);

   // initialise arrays
   for (int a = 0; a<7; a++)
   {
    currentbuttonState[a] = 0;
    currentledState[a] = 0;
    nextbuttonState[a] = 0;
    nextledState[a] = 0;
    nextlocation[a] = 0;
    currentledState[currentlocation] = 1;
   }

   //Check for activation of button
   for ( b=0 ; b<7 ;b++)

    {
         Serial.print("b = ");
         Serial.println(b);
         testposition = (digitalRead(buttonPin[b]));
         Serial.print("testposition = ");
         Serial.println(testposition);
```

```
        //test if change to button position
        if (testposition == 0 ) //&& currentlocation != b)
        {
          // turn off led
          digitalWrite(ledPin[currentlocation],0);

          // Amend arrays and variables for next position/situation
          currentledState[b] = 1;
          currentbuttonState[b] = 1;
          nextledState[b] = 0;
          nextbuttonState[b] = 0;
          nextlocation[b] = 0;
          currentlocation = b;

        // Serial print new values
        Serial.print("b = ");
        Serial.println(b);
        Serial.print("origin = ");
        Serial.println(location[origin]);
        Serial.print("neworigin = ");
        Serial.println(location[b]);
        Serial.print("new currentlocation = ");
        Serial.println(location[origin]);


        //Ammend pins
          digitalWrite(ledPin[b],1);

        // step to new location on button state changes
        int steps = location[b] - location[origin];
        Serial.print("steps = ");
        Serial.println(steps);
        //delay(3000);
        DMProject.step(steps);
        }
    }
}
```