

```

//*****
// Random fill 44 bar LED *
// *
// Michael R Wild 28 Nov 2015 *
// *
// *
//*****

// Based on code by Mike Carter version 1 for STP08CL596

// 6 chips in the bar. 5 with 8 LED's each and one with 4 LED's

// Arduino setup
// Attach OE pin (output enable pin) to ground Don't forget!!!!
// Attach 5V power and ground

#define LE 4 // LW attached to 4
#define CLK 3 // CLK attached to 3
#define SDI 2 // SDU attached to 2

#define DEBUG 0

byte chip1, chip2, chip3, chip4, chip5, chip6; //This represents the chips in the
stick and are global as they are hardware

void setup() {

// Set pins to output
pinMode(SDI, OUTPUT);
pinMode(CLK, OUTPUT);
pinMode(LE, OUTPUT);

Serial.begin(9600);
randomSeed(analogRead(0)); //very random please

Serial.println("Startup\n");
}

byte twoToPower(byte powerValue) {

// Simple.

byte i = 0x01;
i = i<<powerValue;
return i;

}

byte twoToPowerMask(byte powerValue) {

// Simple.

byte i = 0x01;
i = ~(i<<powerValue);

return i;

}

```

```

void stickIt(int valueToDisplay) {
// Updating global local values by translating 1-44 to an LED

    if (valueToDisplay>44) {
        //ignore
    }
    else if (valueToDisplay>40) {
        chip6 = chip6^twoToPower((valueToDisplay-41));
    }
    else if (valueToDisplay>32) {
        chip5 = chip5^twoToPower((valueToDisplay-33));
    }
    else if (valueToDisplay>24) {
        chip4 = chip4^twoToPower((valueToDisplay-25));
    }
    else if (valueToDisplay>16) {
        chip3 = chip3^twoToPower((valueToDisplay-17));
    }
    else if (valueToDisplay>8) {
        chip2 = chip2^twoToPower((valueToDisplay-9));
    }
    else if (valueToDisplay>0) {
        chip1 = chip1^twoToPower((valueToDisplay-1));
    }
}

```

```

void unstickIt(int valueToDisplay) {
// Updating global local values by translating 1-44 to an LED

    if (valueToDisplay>44) {
        //ignore
    }
    else if (valueToDisplay>40) {
        chip6 = chip6&twoToPowerMask((valueToDisplay-41));
    }
    else if (valueToDisplay>32) {
        chip5 = chip5&twoToPowerMask((valueToDisplay-33));
    }
    else if (valueToDisplay>24) {
        chip4 = chip4&twoToPowerMask((valueToDisplay-25));
    }
    else if (valueToDisplay>16) {
        chip3 = chip3&twoToPowerMask((valueToDisplay-17));
    }
    else if (valueToDisplay>8) {
        chip2 = chip2&twoToPowerMask((valueToDisplay-9));
    }
    else if (valueToDisplay>0) {
        chip1 = chip1&twoToPowerMask((valueToDisplay-1));
    }
}

```

```

void clearIt() {

// Clear the variables that are the logical version of the display ships

```

```

chip1 = 0x0;
chip2 = 0x0;
chip3 = 0x0;
chip4 = 0x0;
chip5 = 0x0;
chip6 = 0x0;

}

void sendIt() {

// Send out the logical version of the chips to the display

//send the last chip first which has 4 LEDs. 15 will turn them all on
shiftOut(SDI, CLK, MSBFIRST, chip6);
// send the second to last chip
shiftOut(SDI, CLK, MSBFIRST, chip5);
// send the third to last chip
shiftOut(SDI, CLK, MSBFIRST, chip4);
// send the fourth to last chip
shiftOut(SDI, CLK, MSBFIRST, chip3);
//send to the second chip
shiftOut(SDI, CLK, MSBFIRST, chip2);
// shift out first chip
shiftOut(SDI, CLK, MSBFIRST, chip1);

}

void upIt(int delayValue) {
  for (int numberToDisplay = 0; numberToDisplay < 45; numberToDisplay++) {

    // take the latchPin low so
    // the LEDs don't change while you're sending in bits:
    digitalWrite(LE, LOW);

    // set to empty
    clearIt();

    stickIt(numberToDisplay);

    // shift out the values to display
    sendIt();

    //take the latch pin high so the LEDs will light up:
    digitalWrite(LE, HIGH);

    delay(delayValue);
  }
}

void downIt(int delayValue) {
  for (int numberToDisplay =44; numberToDisplay >0 ; numberToDisplay--) {

    // take the latchPin low so
    // the LEDs don't change while you're sending in bits:
    digitalWrite(LE, LOW);

    // set to empty

```

```

clearIt();

stickIt(numberToDisplay);

// shift out the values to display
sendIt();

//take the latch pin high so the LEDs will light up:
digitalWrite(LED, HIGH);

delay(delayValue);
}
}

void fillItRandom (int delayValue) {
// Fill bar randomly

int LEDs[44]; // 0-43

// use card shuffle logic
for (int i=0; i<44; i++) {
    LEDs[i] = i;
}
for (int i=0; i<44; i++) {
    int r= random(44); // 0-43 random
    int j = LEDs[i];
    LEDs[i] = LEDs[r];
    LEDs[r] = j;
}

clearIt();

for (int i=0; i<44; i++) {
    // take the latchPin low so
    // the LEDs don't change while you're sending in bits:
    digitalWrite(LED, LOW);

    int numberToDisplay = 1 + LEDs[i];

    stickIt(numberToDisplay);

    // shift out the values to display
    sendIt();

    //take the latch pin high so the LEDs will light up:
    digitalWrite(LED, HIGH);

    delay(delayValue);

}
delay(delayValue);
}

void emptyItRandom (int delayValue) {
// Fill bar randomly

int LEDs[44]; // 0-43

// use card shuffle logic

```

```

for (int i=0; i<44; i++) {
  LEDs[i] = i;
}
for (int i=0; i<44; i++) {
  int r= random(44); // 0-43 random
  int j = LEDs[i];
  LEDs[i] = LEDs[r];
  LEDs[r] = j;
}

lightItAll(delayValue); // fill it up

for (int i=0; i<44; i++) {
  // take the latchPin low so
  // the LEDs don't change while you're sending in bits:
  digitalWrite(LE, LOW);

  int numberToDisplay = 1 + LEDs[i];

  unstickIt(numberToDisplay);

  // shift out the values to display
  sendIt();

  //take the latch pin high so the LEDs will light up:
  digitalWrite(LE, HIGH);

  delay(delayValue);
}
delay(delayValue);
}

void lightItAll(int delayValue) {
//Just set it and light it up

clearIt();
// take the latchPin low so
// the LEDs don't change while you're sending in bits:
digitalWrite(LE, LOW);

// shift out the values to display
sendIt();

//take the latch pin high so the LEDs will light up:
digitalWrite(LE, HIGH);

for (int i=1; i<45; i++) { //This is the only way this works!
  stickIt(i);

  // take the latchPin low so
  // the LEDs don't change while you're sending in bits:
  digitalWrite(LE, LOW);

  // shift out the values to display
  sendIt();

  //take the latch pin high so the LEDs will light up:
  digitalWrite(LE, HIGH);
}
}

```

```

}

delay(delayValue);
}

void fibIt(int delayValue) {
//Just set it and light it up

clearIt();
// take the latchPin low so
// the LEDs don't change while you're sending in bits:
digitalWrite(LE, LOW);

// shift out the values to display
sendIt();

//take the latch pin high so the LEDs will light up:
digitalWrite(LE, HIGH);

int i=0;
int j=1;

while (j<45) { //Enjoy Fib

    int jHold = j;
    j = i + j;
    i = jHold;
    stickIt(j);

    // take the latchPin low so
    // the LEDs don't change while you're sending in bits:
    digitalWrite(LE, LOW);

    // shift out the values to display
    sendIt();

    //take the latch pin high so the LEDs will light up:
    digitalWrite(LE, HIGH);

    delay(delayValue);

}

delay(delayValue);
}

void primeIt (int delayValue) {
// Fill bar randomly

lightItAll(delayValue); // fill it up

for (int i=2; i<22; i++) {
    // take the latchPin low so
    // the LEDs don't change while you're sending in bits:

```

```
for (int j=(i+1); j< 45; j=j+1) {

    digitalWrite(LED, LOW);

    unstickIt(j);

    // shift out the values to display
    sendIt();

    //take the latch pin high so the LEDs will light up:
    digitalWrite(LED, HIGH);

    delay(delayValue);
}

}

delay(delayValue);
}

void loop() {

// Light it
    lightItAll(1000);
// Cylon it
    upIt(40);
    downIt(40);
// Fill it
    fillItRandom(100);
// Fib
    fibIt(500);
// Empty ot it
    emptyItRandom(100);
// Primes
    primeIt(100);
}
```