New code done.

There is now a file exporter for the gimp.

it is HERE (not yet but meh)

that is all.

Archived... or to be Archived

Current working code

ledsign.h

```
#ifndef LEDSIGN_H
#define LEDSIGN_H

#include "mbed.h"

void setup();
void writeColour();
void writeArray(int * pointer);
void writeTop(int topAddress);
void writeBot(int botAddress);

#endif
```

main.cpp

```
#include "ledsign.h"

#define B 0
#define R 1
#define G 2
#define O 3


int sign_a[32][128] = {
{B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, B, B, B, B, B, R, R, R, R, R, R, R, R, R, R, R, R, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
```

```
B, B, B},
{B, B, B, B, B, B, B, B, R, R, R, O, O, O, O, O, O, O, O, O, O, R, R, R, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, R, R, B, B,
R, R, R, B, B, R, R, R, R, B, B, B, B, B, B, B, B, R, R, R, R, R, B, B, B,
B, B, B, B, B, B, B, B, B, R, R, R, B, B, B, R, R, R, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, B, B, B,
B, R, B, B, B, B, B, R, B, R, B, B, B, B, B, B, R, B, B, B, R, R, B, B, B,
B, B, B, B, B, B, B, B, B, B, R, R, B, B, B, R, R, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R,
R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, B, B, B,
B, R, B, B, B, B, B, R, B, R, B, B, B, B, B, R, B, B, B, B, B, R, B, B, B,
B, B, R, R, R, B, B, B, B, B, R, B, R, B, R, B, R, B, B, B, B, R, R, R, R,
B, B, B, B, B, R, R, B, R, R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, B, B, B,
B, R, B, B, B, B, R, B, B, R, B, B, B, B, B, R, B, B, B, B, B, B, B, B, B,
B, B, R, R, R, B, B, B, B, B, R, B, R, B, R, B, R, B, B, B, B, B, B, B, B,
R, B, B, B, B, B, R, R, B, B, B, R, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, O, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, R, R, R,
R, R, B, B, B, B, R, B, B, B, R, B, B, B, B, R, B, B, B, B, B, B, B, B, B,
B, B, R, R, R, B, B, B, B, B, R, B, R, B, R, B, R, B, B, B, B, R, R, R, R,
R, B, B, B, B, B, R, B, B, B, B, R, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, O, R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, B, B, B,
B, R, B, B, B, B, R, R, R, R, R, B, B, B, B, R, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, R, B, B, R, B, B, R, B, B, B, R, R, B, B, B,
R, B, B, B, B, B, R, B, B, B, B, R, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, O, O, R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, B, B, B,
B, R, B, B, B, R, B, B, B, B, R, R, B, B, B, R, B, B, B, B, B, B, B, B, B,
B, B, R, R, R, B, B, B, B, B, R, B, B, B, B, B, R, B, B, B, R, B, B, B, B,
R, B, B, B, B, B, R, B, B, B, B, R, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, R, R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, B, B, B,
B, R, B, B, B, R, B, B, B, B, B, R, B, B, B, B, R, B, B, B, B, R, B, B, B,
B, B, R, R, R, B, B, B, B, B, R, B, B, B, B, B, R, B, B, B, R, B, B, B, R,
R, B, B, B, B, B, R, B, B, B, B, R, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R,
R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, R, R, R, B, B,
```

```
R, R, R, B, R, R, R, R, B, R, R, R, R, B, B, B, B, R, R, R, R, B, B, B, B,
B, B, R, R, R, B, B, B, B, R, R, R, B, B, B, R, R, R, B, B, B, R, R, R, B,
R, R, B, B, B, R, R, R, B, B, R, R, R, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, R,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, R, B, R,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, B, B, B, R,
B, B, B, B, B, B, B, B, B, B, G, G, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, G, G, B, B, B, B,
B, B, B},
{B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, R, B, B, B, B, R,
B, B, B, B, B, B, B, B, B, B, B, G, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, G, B, B, B, B,
B, B, B},
{B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, B, B, B, B, B, B, R,
B, B, B, B, B, B, B, B, B, B, B, G, G, G, G, B, B, B, B, G, G, G, B, B, B,
B, G, G, G, G, B, G, G, G, G, G, G, B, B, B, G, G, G, B, B, G, G, G, G, G,
B, B, B, B, B, B, B, B, B, B, G, G, G, G, B, B, B, G, G, B, G, G, B, B, B,
G, G, G, G, G, B, B, B, B, B, B, B, G, G, B, G, G, B, B, B, G, B, G, G, G,
B, B, B},
{B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, R, B, B, B, B, B, R,
B, B, B, B, B, B, B, B, B, B, B, G, B, B, B, G, B, B, B, B, B, G, B, B, B,
G, B, B, B, G, B, B, G, B, G, B, G, B, B, B, B, B, G, B, B, B, G, B, B, B,
G, B, B, B, B, B, B, B, B, G, B, B, B, B, G, B, B, B, G, G, B, B, B, B, G,
B, B, B, G, B, B, B, B, B, B, B, B, B, G, B, B, G, B, B, B, G, B, G, B, B,
B, B, B},
{B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, B, B, B, B, R,
B, B, B, B, B, B, B, B, B, B, B, G, B, B, B, G, B, B, G, G, G, G, B, B, B,
G, B, B, B, B, B, B, G, B, G, B, G, B, B, G, G, G, G, B, B, B, G, B, B, B,
G, B, B, B, B, B, B, B, B, G, B, B, B, B, G, B, B, B, G, B, B, B, B, B, G,
B, B, B, G, B, B, B, B, B, B, B, B, B, G, B, B, G, B, B, B, G, G, G, B, B,
B, B, B},
{B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, R, B, B, R,
B, B, B, B, B, B, B, B, B, B, B, G, B, B, B, G, B, B, G, B, B, G, B, B, B,
G, B, B, B, G, B, B, G, B, G, B, G, B, B, G, B, B, G, B, B, B, G, B, B, B,
G, B, B, B, G, G, B, B, B, G, B, B, B, B, G, B, B, B, G, B, B, B, B, B, G,
B, B, B, G, B, B, B, G, G, B, B, B, B, G, B, B, G, B, B, B, G, B, B, G, B,
```

```
B, B, B},
{B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R, R, R,
B, B, B, B, B, B, B, B, B, B, G, G, G, B, G, G, G, B, G, G, G, G, G, B, B,
B, G, G, G, G, B, G, G, G, G, B, G, G, B, G, G, G, G, G, B, G, G, G, B, G,
G, G, B, B, G, G, B, B, B, B, G, G, G, G, B, B, B, G, G, G, G, B, B, B, B,
G, G, G, G, B, B, B, G, G, B, B, B, B, G, G, G, G, G, B, G, G, B, B, G, G,
B, B, B},
{B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R, R,
R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, G, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
R, R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
G, G, G, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, O, R, R, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, O, O, R, R, B, B, B, B, B, B, B, B, G, G, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, O, R, R, B, B, B, B, B, B, B, B, G, B, B, G, B, G, G, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, G, G, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, G, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
O, R, R, B, B, B, B, B, B, B, B, B, G, B, G, G, B, B, G, G, G, G, B, B, B,
G, G, G, B, B, G, G, G, G, B, G, G, G, G, G, G, B, B, G, G, G, B, G, G, G,
G, G, B, B, G, G, G, G, B, B, G, G, G, G, B, B, B, G, G, G, B, B, G, G, G,
G, B, B, B, G, B, B, B, B, B, G, G, G, B, B, G, G, B, G, G, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O,
R, R, B, B, B, B, B, B, B, B, B, B, G, G, B, G, B, B, G, B, B, G, B, B, B,
B, B, G, B, G, B, B, B, B, B, B, G, B, G, B, G, B, B, B, B, G, B, B, G, B,
B, G, B, G, B, B, B, B, B, B, G, B, B, G, B, B, G, B, B, B, G, B, G, B, B,
B, B, B, G, G, G, G, B, B, G, B, B, B, G, B, B, G, G, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, B, R, R, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, R,
R, B, B, B, B, B, B, B, B, B, B, B, G, G, B, G, B, B, G, B, B, G, B, B, G,
```

```
G, G, G, B, G, B, B, B, B, B, B, G, B, G, B, G, B, G, G, G, G, B, B, G, B,
B, G, B, G, B, B, B, B, B, B, G, B, B, G, B, B, G, G, G, G, G, B, G, G, G,
G, B, B, B, G, B, B, B, B, G, G, G, G, G, B, B, G, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, B, B, R, R, R, O, O, O, O, O, O, O, O, O, O, O, O, R, R, R,
B, B, B, B, B, B, B, B, B, B, B, B, G, G, G, G, B, B, G, B, B, G, B, B, G,
B, B, G, B, G, B, B, B, G, B, B, G, B, G, B, G, B, G, B, B, G, B, B, G, B,
B, G, B, G, B, B, B, G, B, B, G, B, B, G, B, B, G, B, B, B, B, B, B, B, B,
G, B, B, B, G, B, B, B, B, G, B, B, B, B, B, B, G, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, B, B, B, B, R, R, R, R, O, O, O, O, O, O, R, R, R, R, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, G, G, B, B, B, G, G, B, B, G, G, B, G,
G, G, G, G, B, G, G, G, B, B, G, G, B, G, B, G, B, G, G, G, G, B, G, G, B,
B, G, G, B, G, G, G, B, B, G, G, B, B, G, G, B, B, G, G, G, G, B, G, G, G,
G, B, B, B, G, G, G, G, B, B, G, G, G, G, B, G, G, G, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, B, B, B, B, B, B, R, R, R, R, R, R, R, R, R, R, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, G, G, G, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B},
{B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B, B,
B, B, B}
};

// Sign Output Pins

BusOut address(p17, p18, p19, p20); // Address 0 to 16
BusOut colour(p15, p16);            // 0 = off, 1 = red, 2 = green, 3 =
orange

DigitalOut abTop(p14);              // bank A or B switch for Top Row - 0 =
A, 1 = B
DigitalOut clkTop(p13);             // clock for Top Row
DigitalOut weTop(p28);              // Write Enable for Top Row
DigitalOut aeTop(p27);              // Address Enable for Top Row
DigitalOut enbTop(p26);             // Enable for Top Row

DigitalOut abBot(p25);              // bank A or B switch for Bottom Row - 0
= A, 1 = B
DigitalOut clkBot(p24);             // clock for Bottom Row
DigitalOut weBot(p23);              // Write Enable for Bottom Row
DigitalOut aeBot(p22);              // Address Enable for Bottom Row
DigitalOut enbBot(p21);             // Enable for Bottom Row

int main() {
```

```
    setup();

    writeArray((int*)sign_a);

}

void setup() {
    address = 0;
    colour = 0;
    abTop = 0;
    clkTop = 0;
    weTop = 0;
    aeTop = 0;
    enbTop = 0;
    abBot = 0;
    clkBot = 0;
    weBot = 0;
    aeBot = 0;
    enbBot = 0;

    //simple test pattern

    enbTop = 1;
    enbBot = 1;

    abTop = 1; // set top bank to A
    abBot = 1; // set bottom bank to A
    colour = 3; // set colour to orange
    writeColour();
    wait(1);
    abTop = 0; // set top bank to A
    abBot = 0; // set bottom bank to A
    colour = 2; // set colour to orange
    writeColour();
    wait(1);
    abTop = 1; // set top bank to A
    abBot = 1; // set bottom bank to A
    colour = 1; // set colour to orange
    writeColour();
    wait(1);
    abTop = 0; // set top bank to A
    abBot = 0; // set bottom bank to A
    colour = 0; // set colour to orange
    writeColour();
    wait(1);
}

void writeColour() {

    for (int i=0; i<128; i++) { // clock in 128 bits to turn all the LED's
on
```

```
        clkTop = 1;
        clkBot = 1;
        wait_us(10);
        clkTop = 0;
        clkBot = 0;
        wait_us(10);
    }

    for (int i=0; i<16; i++) {
        writeTop(i);
        writeBot(i);
    }

}

void writeArray(int * pointer) {

    abTop = 1;
    abBot = 1;

    //top half image in array

    for(int ad = 0; ad < 16 ; ad++) {

        for(int row = 0; row < 128 ; row++) {

            colour = *(pointer++/* (128*ad) + row*/);
            wait_us(1);
            clkTop = 1;
            wait_us(1);
            clkTop = 0;
            wait_us(1);

        }

    writeTop(ad);

    }

    //bottom half image in array

    for(int ad = 0; ad < 16 ; ad++) {

        for(int row = 0; row < 128 ; row++) {

            colour = *(pointer++/* (128*(ad+16)) + row*/);
            wait_us(1);
            clkBot = 1;
            wait_us(1);
            clkBot = 0;
            wait_us(1);
```

```
        }

    writeBot(ad);

    }

    abTop = 0;
    abBot = 0;

}


void writeTop(int topAddress) {
    address = topAddress;
    aeTop = 1;
    wait_us(1);
    weTop = 1;
    wait_us(1);
    weTop = 0;
    wait_us(1);
    aeTop = 0;
    wait_us(1);
}

void writeBot(int botAddress) {
    address = botAddress;
    aeBot = 1;
    wait_us(1);
    weBot = 1;
    wait_us(1);
    weBot = 0;
    wait_us(1);
    aeBot = 0;
    wait_us(1);
}
```

First Test

This first test SHOULD flash the first line of the display between Red and Green every half a second

```
/*********************************************
*LED Sign Testing - Part 1                   *
*Single Line of Red                          *
*                                            *
*TBSliver                                    *
*********************************************/

int pinSE = 2;
int pinABB = 3;
```

```
int pinA3 = 4;
int pinA2 = 5;
int pinA1 = 6;
int pinA0 = 7;
int pinGR = 8;
int pinCLK = 9;
int pinWE = 10;
int pinRD = 11;
int pinAE = 12;
int pinENB = 13;

void setup()
{
  pinMode(pinSE, OUTPUT);
  pinMode(pinABB, OUTPUT);
  pinMode(pinA3, OUTPUT);
  pinMode(pinA2, OUTPUT);
  pinMode(pinA1, OUTPUT);
  pinMode(pinA0, OUTPUT);
  pinMode(pinGR, OUTPUT);
  pinMode(pinCLK, OUTPUT);
  pinMode(pinWE, OUTPUT);
  pinMode(pinRD, OUTPUT);
  pinMode(pinAE, OUTPUT);
  pinMode(pinENB, OUTPUT);

  digitalWrite(pinSE, HIGH);
}

void loop()
{
  // set address to 0
  digitalWrite(pinA3, LOW);
  digitalWrite(pinA2, LOW);
  digitalWrite(pinA1, LOW);
  digitalWrite(pinA0, LOW);

  // set Red to on for all of the first row in bank A
  digitalWrite(pinGR, LOW);
  digitalWrite(pinABB, HIGH);
  delay(10);
  for(int i=0; i>16; i++)
  {
    shiftOut(pinRD, pinCLK, LSBFIRST, 255);
  }

  // write data to memory
  delay(10);
  digitalWrite(pinAE, HIGH);
  delay(10);
  digitalWrite(pinWE, HIGH);
```

```
  delay(10);
  digitalWrite(pinWE, LOW);
  delay(10);
  digitalWrite(pinAE,LOW);
  delay(10);

  // set Green to on for all of the first row in bank B
  digitalWrite(pinRD, LOW);
  digitalWrite(pinABB, LOW);
  delay(10);
  for(int i=0; i>16; i++)
  {
    shiftOut(pinGR, pinCLK, LSBFIRST, 255);
  }

  // write data to memory
  delay(10);
  digitalWrite(pinAE, HIGH);
  delay(10);
  digitalWrite(pinWE, HIGH);
  delay(10);
  digitalWrite(pinWE, LOW);
  delay(10);
  digitalWrite(pinAE,LOW);
  delay(10);

  // switch between bank A and B to show a change
  while(1)
  {
    digitalWrite(pinABB, HIGH);
    delay(500);
    digitalWrite(pinABB, LOW);
    delay(500);
  }
}
```

LED Sign

```
Each Module is a LUM-256HML350
13 Pin connector

chip on board is:
BU12005-01
212 H03

In Datasheet, pinout is (2)
Pin CN1 CN2 CN3
1   SEin    ENBout  GND
2   A/BBin  AEout   VLED
```

```
3   A3in    RDout   VLED
4   A2in    WEout   GNDLED
5   A1in    CLKout  GNDLED
6   A0in    GRout   VDD
7   GND GND
8   GRin    A0out
9   CLKin   A1out
10  WEin    A2out
11  RDin    A3out
12  AEin    A/BBin
13  ENBin   SEout

Pin Funtion             0           1
SEin    Memory Bank Selection Mode  Auto, change at 15/23   Selected with
A/BBin
A/BBin  Memory Bank Selection       Bank A          Bank B
A3in    RAM Address, bit 3      0           8
A2in    RAM Address, bit 2      0           4
A1in    RAM Address, bit 1      0           2
A0in    RAM Address, bit 0      0           1
GND Ground
GRin    Green Data input        LED Off         LED On
CLKin   Clock input                     Shifted in on leading edge of pulse
WEin    Write Control           Data Not Written    Shift Register contents
written to memory
RDin    Red Data input          LED Off         LED On
AEin
ENBin   Enable Display          Display Off     Display On
```

Category:Projects Category:Equipment Category:Cleanup 2015