



12+

018

Night Rider

main.c

```
typedef union {WORD w; BYTE x[2];} uniword;
typedef union {DWORD dw; BYTE x[4];} unidword;

const WORD Night[16] = {
    0x4000,0x1000,0x0400,0x0100,
    0x0040,0x0010,0x0004,0x0001,
    0x0001,0x0004,0x0010,0x0040,
    0x0100,0x0400,0x1000,0x4000
};

void main()
{
    unidword N;
    uniword Light;
    WORD count;

    PRT0DM0&=~0x55; PRT0DM1&=~0x55; PRT0DM2&=~0x55; // Pull-Down
    PRT2DM0&=~0x55; PRT1DM2&=~0x55; PRT2DM2&=~0x55; // Pull-Down

    for (count=0;;count++) {
        PRT0DR = ~Light.x[0];
        PRT2DR = ~Light.x[1];
        if (!(count%1024)) { N.dw >>= 8; N.x[0] = (N.x[1]+1)%16; }
        switch(count%8) {
            case 0: Light.w = Night[N.x[0]] | Night[N.x[1]] ; break;
            case 1: Light.w = Night[N.x[0]] | Night[N.x[2]] ; break;
            case 2: Light.w = Night[N.x[0]] | Night[N.x[1]] ; break;
            case 3: Light.w = Night[N.x[0]] | Night[N.x[3]] ; break;
            case 4: Light.w = Night[N.x[0]] | Night[N.x[1]] ; break;
            case 5: Light.w = Night[N.x[0]] | Night[N.x[2]] ; break;
            case 6: Light.w = Night[N.x[0]] | Night[N.x[1]] ; break;
            case 7: Light.w = Night[N.x[0]] ; break;
        }
    }
}
```

