

# オシロ文字

## 概要

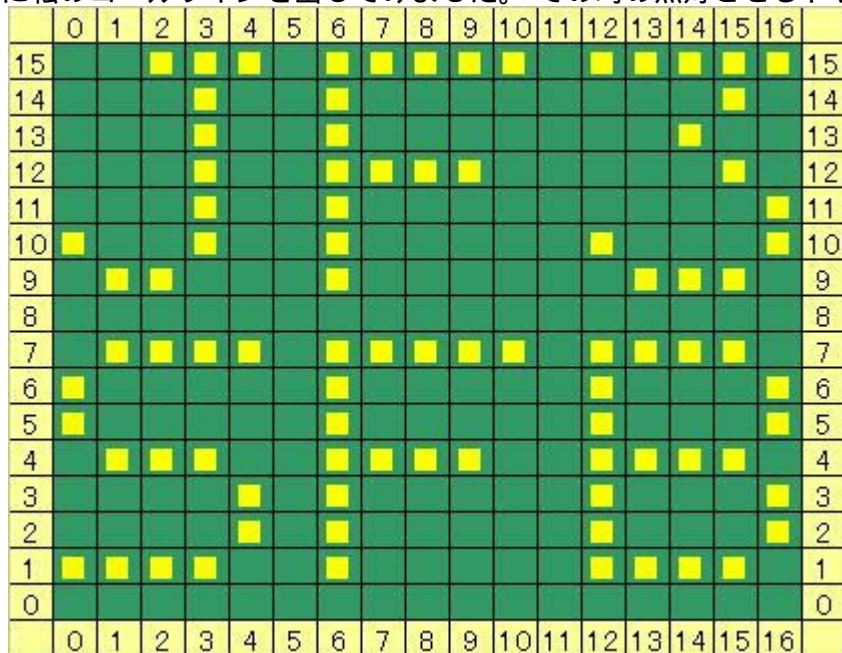
実用性はとても低いと思われるのですが、オシロスコープの画面に文字を表示させて見ました。

## 動作原理

オシロスコープの「X-Y」機能を利用し、PICからX軸用のアナログデータとY軸用のアナログデータを高速に出力することにより、オシロスコープの画面に文字を表示させるものです。

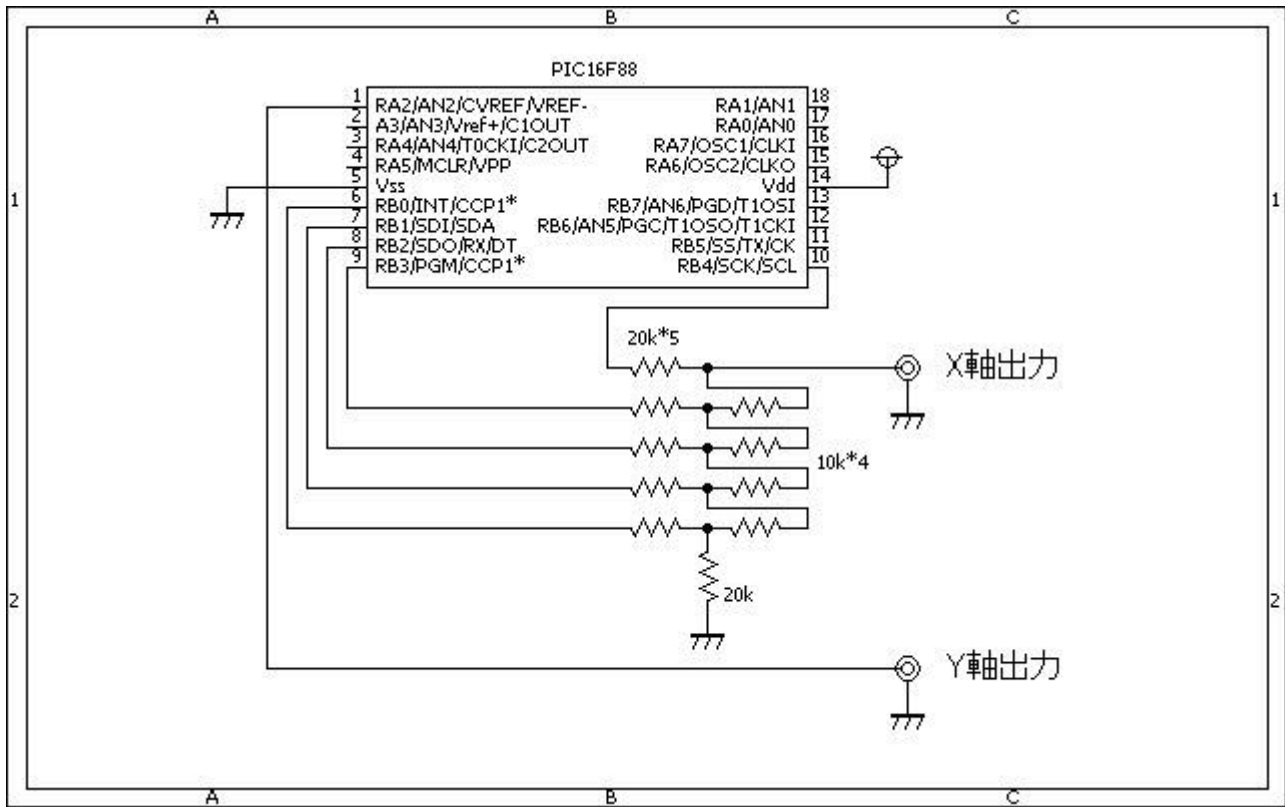
- X軸用にはR-2R型抵抗ラダーを使用したD/A変換を使用します。(5ビット精度)
- Y軸用にはPIC内臓のCVR(Comparator Voltage Reference)モジュールを使用します。(4ビット精度)

画面に私のコールサインを出してみました。その時の点灯させるドットの位置は次の図のようになります。



す。

## 回路図



## ソースコード

osclloStr.c

```
//*****  
*  
/*  
『オシロ文字』  
*/  
//*****  
*  
void DotSet(unsigned char X, unsigned char Y)  
{  
    PORTB = X;  
    CVRCON = 0xE0 | Y;  
}  
  
//*****  
*  
void main()  
{  
    OSCCON = 0b01110000; // クロックは8Mhz  
    CMCON = 0b00000111; // コンパレータは未使用  
    ANSEL = 0b00000000; // □□□変換は未使用  
    TRISA = 0b11111111;
```

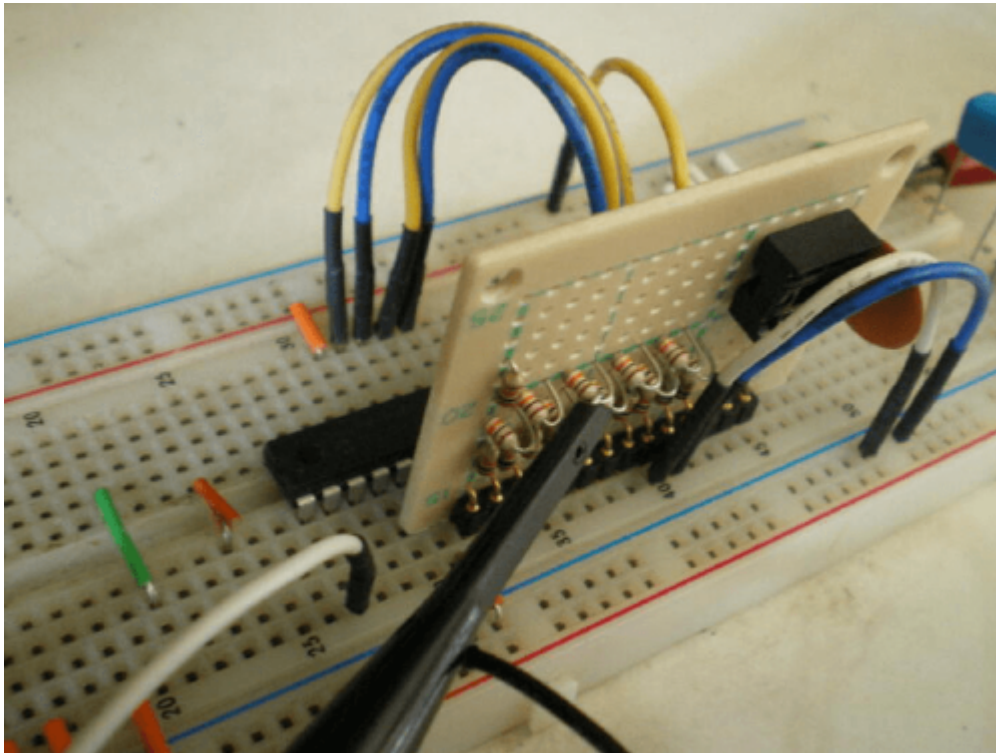
```
TRISB = 0b00000000;  
//  
CVRCON = 0xE0;  
PORTB = 0x00;  
while (1) {  
    DotSet(2, 15);  
    DotSet(3, 15);  
    DotSet(4, 15);  
    DotSet(6, 15);  
    DotSet(7, 15);  
    DotSet(8, 15);  
    DotSet(9, 15);  
    DotSet(10, 15);  
    DotSet(12, 15);  
    DotSet(13, 15);  
    DotSet(14, 15);  
    DotSet(15, 15);  
    DotSet(16, 15);  
    //  
    DotSet(3, 14);  
    DotSet(6, 14);  
    DotSet(15, 14);  
    //  
    DotSet(3, 13);  
    DotSet(6, 13);  
    DotSet(14, 13);  
    //  
    DotSet(3, 12);  
    DotSet(6, 12);  
    DotSet(7, 12);  
    DotSet(8, 12);  
    DotSet(9, 12);  
    DotSet(15, 12);  
    //  
    DotSet(3, 11);  
    DotSet(6, 11);  
    DotSet(16, 11);  
    //  
    DotSet(0, 10);  
    DotSet(3, 10);  
    DotSet(6, 10);  
    DotSet(12, 10);  
    DotSet(16, 10);  
    //  
    DotSet(1, 9);  
    DotSet(2, 9);  
    DotSet(6, 9);  
    DotSet(13, 9);  
    DotSet(14, 9);  
    DotSet(15, 9);  
    //  
}
```

```
DotSet(1, 7);  
DotSet(2, 7);  
DotSet(3, 7);  
DotSet(4, 7);  
DotSet(6, 7);  
DotSet(7, 7);  
DotSet(8, 7);  
DotSet(9, 7);  
DotSet(10, 7);  
DotSet(12, 7);  
DotSet(13, 7);  
DotSet(14, 7);  
DotSet(15, 7);  
//  
DotSet(0, 6);  
DotSet(6, 6);  
DotSet(12, 6);  
DotSet(16, 6);  
//  
DotSet(0, 5);  
DotSet(6, 5);  
DotSet(12, 5);  
DotSet(16, 5);  
//  
DotSet(1, 4);  
DotSet(2, 4);  
DotSet(3, 4);  
DotSet(6, 4);  
DotSet(7, 4);  
DotSet(8, 4);  
DotSet(9, 4);  
DotSet(12, 4);  
DotSet(13, 4);  
DotSet(14, 4);  
DotSet(15, 4);  
//  
DotSet(4, 3);  
DotSet(6, 3);  
DotSet(12, 3);  
DotSet(16, 3);  
//  
DotSet(4, 2);  
DotSet(6, 2);  
DotSet(12, 2);  
DotSet(16, 2);  
//  
DotSet(0, 1);  
DotSet(1, 1);  
DotSet(2, 1);  
DotSet(3, 1);
```

```
DotSet(6, 1);  
DotSet(12, 1);  
DotSet(13, 1);  
DotSet(14, 1);  
DotSet(15, 1);  
}  
}  
  
//*****  
*
```

## 動作確認

D/Aモジュールは、10ビット精度のものですが、5ビット分だけを使用しています。



コールサインを表示させてみました。



如何ですか? 応用すれば何か面白いものが作れるかも。。。 😊

### 著作権表示 **copyright notice**

このページは稲崎様の閉鎖したHPのコピーで、著作権は稲崎様にあります。 [詳細](#) This page is a copy of Mr. Inasaki's closed website, and the copyright is held by him. [Details](#)

From:  
<http://www.deepsky.jp/wiki/> - うごくといいな

Permanent link:  
<http://www.deepsky.jp/wiki/doku.php?id=elechobby:picdic:pic16f88:74&rev=1588323232>

Last update: **2025/10/17 14:28**

