

做一个 GPS 钟

3120000450 刘千惠

## 一、实验目的和要求

用串口连接一个 GPS 模块，从 GPS 得到实时时间，在 7 段数码管或 LCD 上显示。

## 二、实验器材

### 硬件

- 实验板一块；
- 5V/1A 电源一个；
- microUSB 线一根；
- USB-TTL 串口线一根（FT232RL 芯片或 PL2303 芯片）。
- GPS 模块

以下为自备（可选）器材：

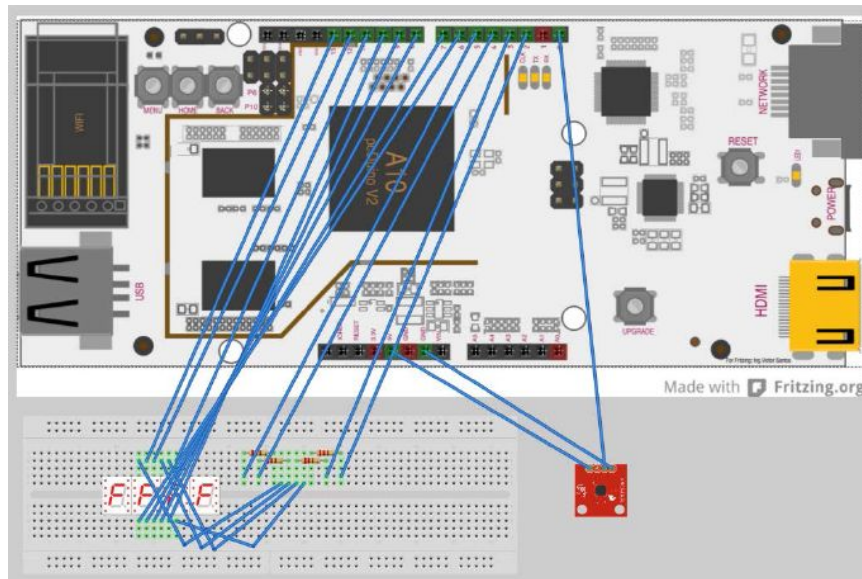
- PC（Windows/Mac OS/Linux）一台；
- 七段数码管或 LCD；
- 以太网线一根（可能还需要路由器等）。

### 软件

- PC 上的 USB-TTL 串口线配套的驱动程序；
- PC 上的串口终端软件，如 minicom、picocom、putty 等；
- PC 上的 SSH 软件，如 putty 等。

## 三、实验步骤

### 1.如图进行连线



2. 在/etc/network/interfaces 设置 pcduino 的静态 IP。为防止串口冲突造成 pcduino 板损坏，本实验 pc 和 pcduino 间使用 ssh 通信。

```
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

#iface wlan3 inet dhcp
#iface wlan3 inet static
#address 192.168.137.1
#netmask 255.255.255.0

#auto eth0
#iface eth0 inet static
#address 222.205.57.254
#gateway 222.205.57.1
#netmask 255.255.255.0
#network 222.205.57.0
#dns-nameservers 10.10.0.21

auto eth0
iface eth0 inet static
address 192.168.137.137
netmask 255.255.255.0
gateway 192.168.137.1
dns-nameservers 192.168.137.1
```

3. 编写 gps 测试代码，输出 gps 返回的全部信息，以测试 gps 状态。

```

#include <core.h>

void setup() {
    Serial.begin(9600);
    delay(1000);
}

void loop() {
    char temp;
    while (Serial.available() > 0) {

        temp = Serial.read();
        printf("%c", temp);

    }
}

```

4. 编写实际实现的代码(主要原理是利用 LED 闪烁时不显示数字的时间来捕捉 GPRMC 标签并获得后面的时间)，并修改 Makefile 后进行 make。

```

        selectDigit(i);
        if (count/FlashDelay%2)
        {
            setSegment(currentTime/deno%10);
            deno = deno / 10;
            if (i==1)
            {
                digitalWrite(pinP, LOW);
            }
        }
        delayMicroseconds(DigitDelay);
    }

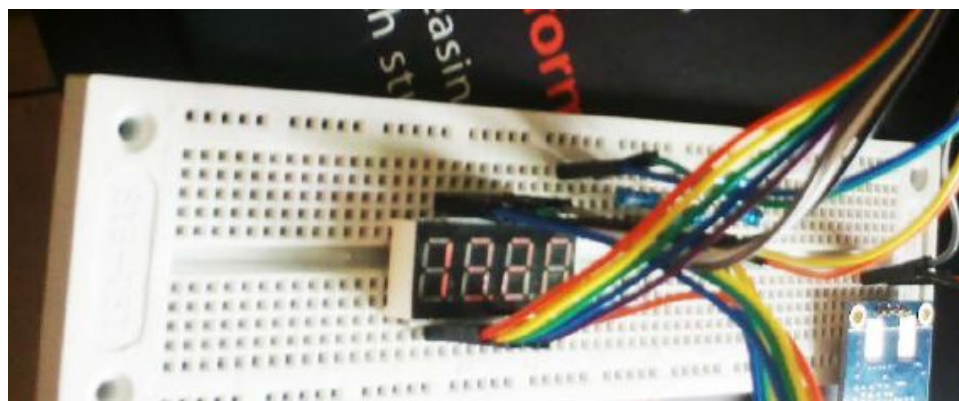
    if (!(count/FlashDelay%2))
    {
        GPSCheck();
    }
    count++;
}

```

5. 将 gps 模块放置在空旷的地方，运行 gps 测试代码进行测试。在等待一段时间后，可以看到 GPRMC 和 GPGLA 行获得了 utc 时间。

```
$GPGSV,3,1,09,03,,,21,09,,,21,11,,,22,14,,,17*7B
$GPGSV,3,2,09,18,,,17,19,,,28,28,,,08,29,,,23*74
$GPGSV,3,3,09,30,,,16*74
$GPGLL,,,,,052003.00,V,N*4E
$GPRMC,052004.00,V,,,,,,100514,,,N*7F
$GPVTG,,,,,,N*30
$GPGGA,052004.00,,,,,0,00,99.99,,,,,*65
$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30
$GPGSV,2,1,07,03,,,17,09,,,19,11,,,19,14,,,10*75
$GPGSV,2,2,07,18,,,12,29,,,20,30,,,09*77
$GPGLL,,,,,052004.00,V,N*49
```

#### 四、实验结果



加入了闪烁功能，在闪烁不显示数字时进行 GPS 数据捕捉



1 分钟后再观察，能够获取时间的更新



程序如下：

```
#include <core.h>
int pinD1 = 5;
int pinD2 = 4;
int pinD3 = 3;
int pinD4 = 2;
int pinA = 13;
int pinB = 11;
int pinC = 9;
int pinD = 7;
int pinE = 6;
int pinF = 12;
int pinG = 10;
int pinP = 8;
int FlashDelay = 500;
int DigitDelay = 100;
int currentTime = 0;
int count = 0;
int cathode[8] = {pinA, pinB, pinC, pinD, pinE, pinF, pinG, pinP};
int anticathode[4] = {pinD1, pinD2, pinD3, pinD4};
int numbers[10][7] = {{0, 0, 0, 0, 0, 0, 1},
```

```

{1, 0, 0, 1, 1, 1, 1},
{0, 0, 1, 0, 0, 1, 0},
{0, 0, 0, 0, 1, 1, 0},
{1, 0, 0, 1, 1, 0, 0},
{0, 1, 0, 0, 1, 0, 0},
{0, 1, 0, 0, 0, 0, 0},
{0, 0, 0, 1, 1, 1, 1},
{0, 0, 0, 0, 0, 0, 0},
{0, 0, 0, 0, 1, 0, 0}
};

void setup()
{
  pthread_t id;
  int i;
  for (i=0;i<8;i++)
  {
    pinMode(cathode[i], OUTPUT);
  }
  for (i=0;i<4;i++)
  {
    pinMode(anticathode[i], OUTPUT);
  }
  Serial.begin(9600);
}

void clearLED()
{
  int i;
  for (i=0;i<8;i++)
  {
    digitalWrite(cathode[i], HIGH);
  }
}

void setSegment(int x)
{
  int i;
  for (i=0;i<7;i++)
  {
    digitalWrite(cathode[i], numbers[x][i]);
  }
}

void selectDigit(int x)
{
  int i;
  for (i=0;i<4;i++)

```

```

{
digitalWrite(anticathode[i], LOW);
}
digitalWrite(anticathode[x], HIGH);
}
void GPScheck()
{
char temp;
int match = 0;
int utctime;
int indent;
int wait = 0;
while (wait<1000)
{
wait++;
while (Serial.available()>0)
{
temp = Serial.read();
switch (temp)
{
case 'G': if (match==0) match++; else match = 0; break;
case 'P': if (match==1) match++; else match = 0; break;
case 'R': if (match==2) match++; else match = 0; break;
case 'M': if (match==3) match++; else match = 0; break;
case 'C': if (match==4) match++; else match = 0; break;
case ',': if (match==5) match++; else match = 0; break;
}
if (match>5)
{
utctime = 0;
indent = 100000;
while (Serial.available()> 0) {
temp = Serial.read();
if (temp==',' || indent==0) break;
utctime = utctime+(temp-48)*indent;
indent = indent / 10;
}
if (indent==0) currentTime = (utctime/10000+8)%24*100+utctime/100%100;
match = 0;
//printf("%d\n",currentTime);
wait=1001;
}
}
}
}

```

```

}
void loop()
{
  int i;
  int deno;
  deno = 1000;
  for (i=0;i<4;i++)
  {
    clearLED();
    selectDigit(i);
    if (count/FlashDelay%2)
    {
      setSegment(currentTime/deno%10);
      deno = deno / 10;
      if (i==1)
      {
        digitalWrite(pinP, LOW);
      }
      delayMicroseconds(DigitDelay);
    }
  }
  if (!(count/FlashDelay%2))
  {
    GPScheck();
  }
  count++;
}

```