|  |  |
| --- | --- |
| **姓 名** | 张贤，陈伟钿 |
|  | 张璘，张忠伦 |
| **学 号** | 201211911332,201211921404 |
|  | 201211911331,201211911333 |
| **专 业** | 电子科学与技术 |
| **班 级** | 电科1123，电科1124（伟钿） |

**近红外光波长的光功率计设计**

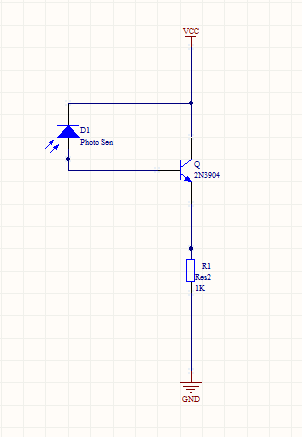
**摘要：**这是一个简单的测近红外光功率的仪器，由光纤端口输入光信号至光电传感器InGaAs-PIN光电二极管，然后通过放大滤波将噪声消除。再通过一个ADC0809数模转换芯片将模拟电压量转化为数字量。并且使用最经典的STC89C51单片机。使得开发成本降低。总得来说该设备具有成本低。效率高，误差较小等优点，可以满足普通情况下对精准度要求不高的场所。

## 系统总体设计方案

近红外光功率计的基本工作流程图如图1所示[1]，它首先把传输过来的光信号投射在InGaAs-PIN光探测器的光敏面上以将其转变为电流，再经过I/V变换电路和放大电路得到电压信号。然后把这个信号送到低通滤波器进行滤波及响应度补偿放大，以得到与功率值相对应的直流电压，之后再将该电压经A/D转换，以得到表示功率大小的数字量，最后通过STC89C516+进行数据处理和判断后，将数据送入显示模块进行功率显示或指示。

图1

## 系统分析与各模块或单元电路的设计

1. 光电转换电路

首先选取一种适合近红外波段的光电检测器。InGaAs-PIN是一种比较好的材料，因为InGaAs-PIN是一种低噪声、高响应度的光电检测器，具有较高的测量灵敏度。InGaAs-PIN的光响应度可达0.8，波长范围为1.3~1.6μm。此外，暗电流、上升时间、带宽、偏置电压等方面，InGaAs-PIN无疑是一种很好的选择。但是一般的PIN产生的光电流很小，不能直接用于测量，需要通过适当的低噪声放大后。才能进行处理。右边图2为原理图。

图2

1. I/V变换与滤波放大电路

I/V变换可以使用基本的LM741进行放大，如下图所示[2]。由于经过三级管放大后的光电流，存在着一定的噪声，所以在放大电路的输入端，通过一个低通滤波器，即前面一个LM741作为滤波器，滤除高频的信号，使有用的低频信号能够通过。之后再将放大后的信号通过一个I/V变换电路，对信号进行放大。放大倍数可以通过调节电压部分的多路开关进行调节。经过放大后的信号由第二片LM741的第六引脚输出。总体电路如三所示。

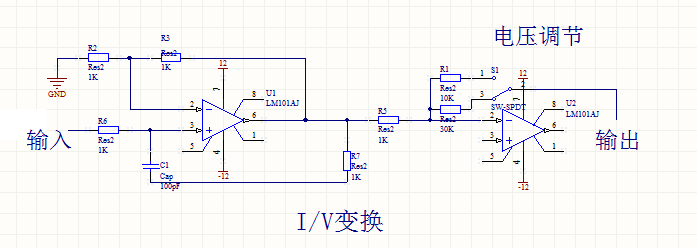


图3

1. AD转换模块

将模拟量转换成数字量是A/D模块的功能。将由放大滤波电路放大后的信号，输入数模转换模块，数模转换模块将模拟量转换成数字量，输入单片机。具体电路图如图4所示。图中，经过放大的信号由26口（IN0）输入，经过ADC由OUT1~OUT8输出。EOC是状态端，连接单片机。START是控制端，也是连接单片机，读取其状态。在8路模拟输入中，我们只用一路模拟输入。所以我们把地址线都接地了。

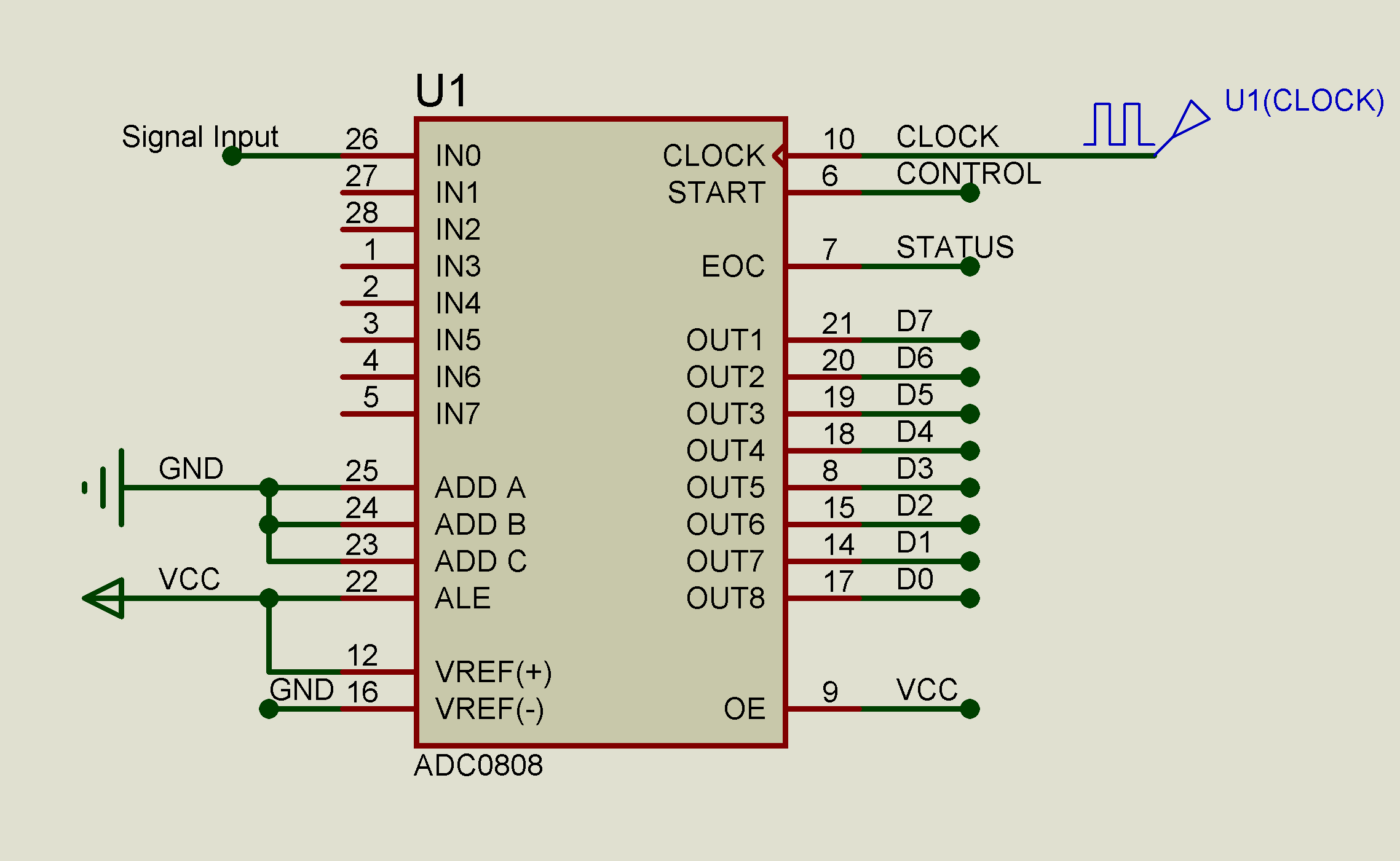


图4

1. 单片机控制系统。

单片机控制系统主要负责ADC输出的读取。将读取的数据（从P1口读取），然后内部将读取的数字量转换成dbm分贝毫瓦。然后控制LCD模块进行显示。电图图如图5所示。

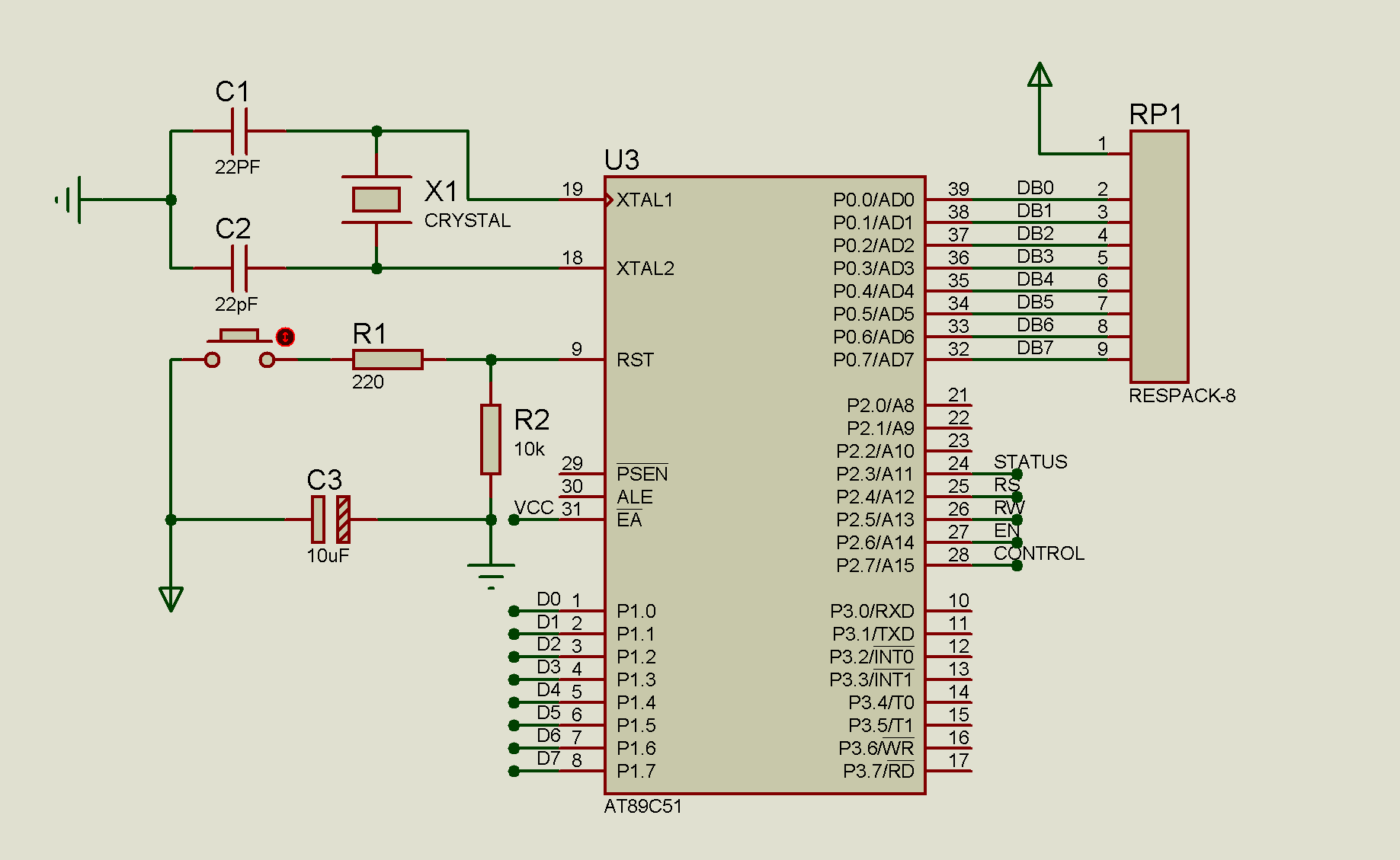


图5

1. LCD显示模块

其实LCD模块就是显示而已，主要功能还是由80C51来控制其显示。如图6所示。

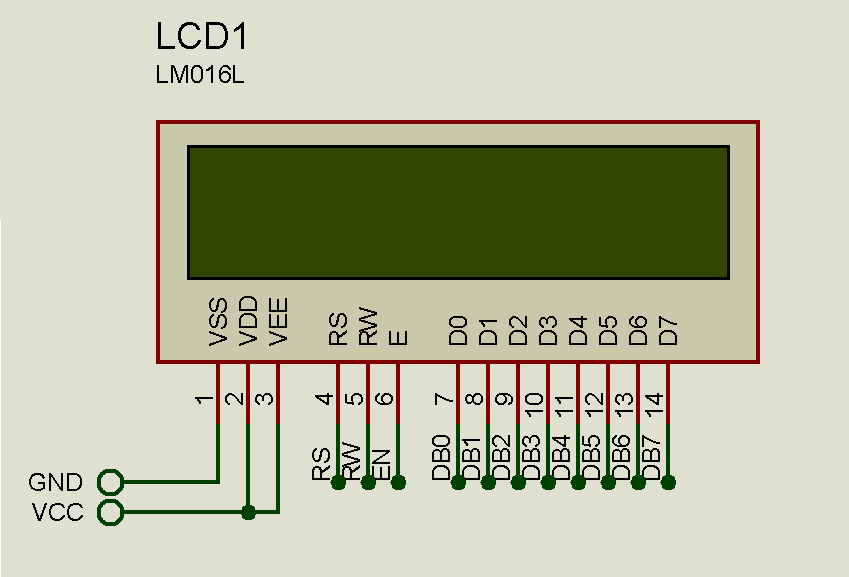


图6

## 参数计算

根据前面的相关电路，结合书本知识和相关硬件的参数，再结合相关公式以便于计算分析：

1.光电转换电路

（1） InGaAs-PIN （微型封装）光电探测器的相关参数：

特点： 微型封装，封装尺寸≤2.41 mm 高响应 工作电压 5V 超低暗电流 单针脚密封 工作温度 -40～+85℃ 应用： 光纤通信 数据/图像传输 光纤传感 光测量仪器仪表，如表1 [2]所示。

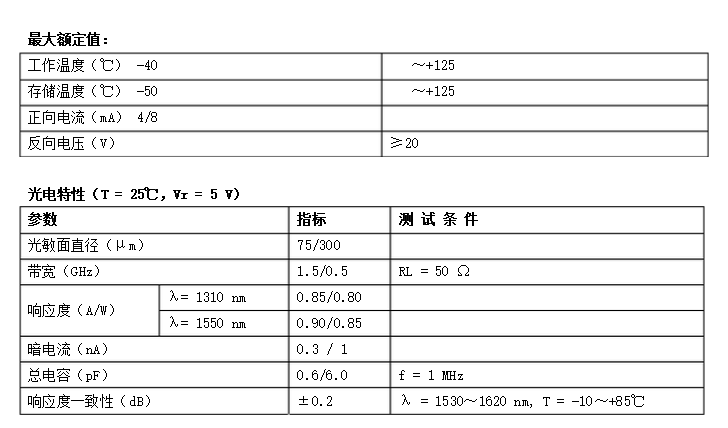


表1

（2）涉及公式：

1. 噪声：光电探测器在光电探测时，要受到无用信号的干扰，称为光电探测器的噪声。

Us（t）表示信号

U (t ) 表示传输或变换后得到的信号。

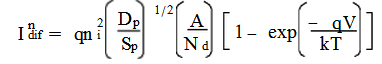
噪声信号：Un（t）=U(t)-Us（t）

B、响应度R

响应度R（和量子效率η）是描述器件光电转换能力的物理量，其大小为光电转换器（又称光检测器）的平均输出电流Ip与光电转换器（又称光检测器）的平均输入功率Po的比值，即输出电信号电流大小与输入光信号功率大小之比。

　　用公式表示为：R=Ip/Po，单位为A/W。PIN管在光功率Po的照射下，产生的光电流为Ip=(e\*Po\*η)/(h\*f)。式中e为电子电荷；h为普朗克常数；f为入射光频率；η为量子效率，其数值总是小于1。因此又可把响应度的公式写为：R=η/(h\*f/e)。响应度与器件材料、光波长有关。

C、暗电流



式中:ni为本征载流子浓度,Dp为i区中空穴扩散系数,Sp为i区中空穴的寿命,Nd为i区的掺杂浓度,A是耗尽层与p区和i区的接触面积,V为探测器所加偏压。

D、上升时间



式中,S(f)为频率是f时的响应度;S0为频率是零时的响应度

2. I/V变换与滤波放大电路

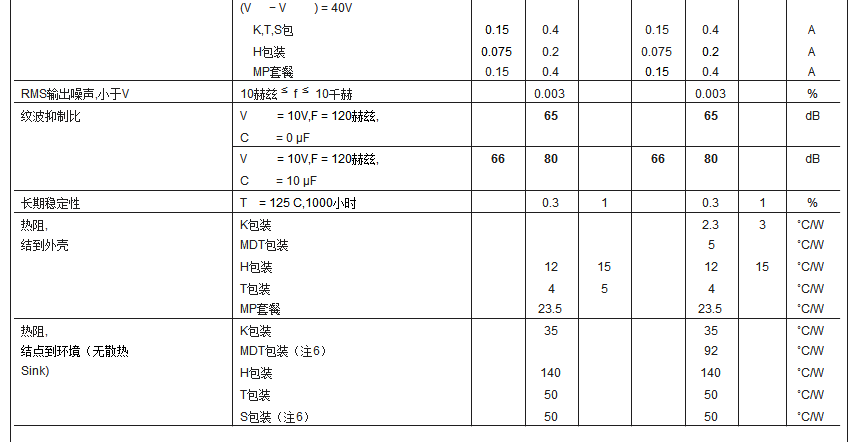
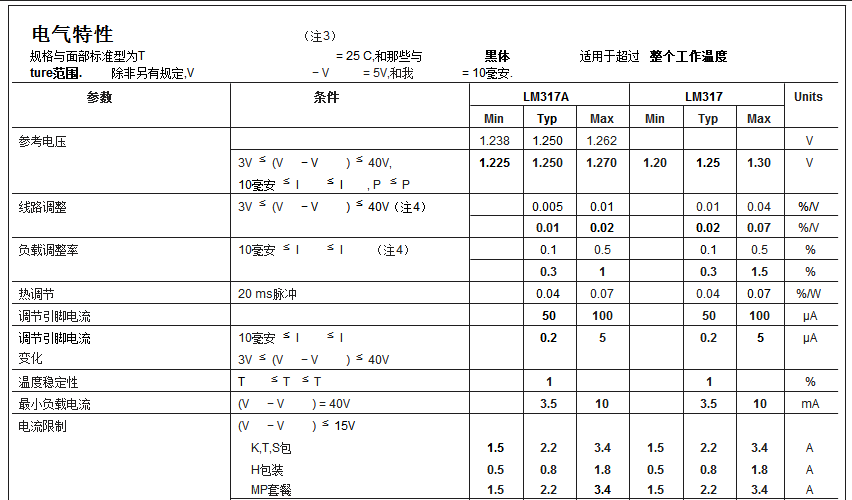
（1）LM741的相关参数[2]

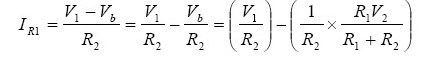
表2

（2）涉及公式

A、 下图是能够将两个输入电压V1、V2之间的电压差值放大的电路，同样是利用将输出电压引回反相输入端的负反馈电路，可以得到R1/R2倍的放大倍率。因为此电路之b点电位是由V2决定，所以a点与b点会有相同的输入电压，则



通过反馈电阻R1上的电流为



故通过电阻R1所产生的压降为



因为输出电压Vo是VR1与Vb之和，故整理得输出电压为

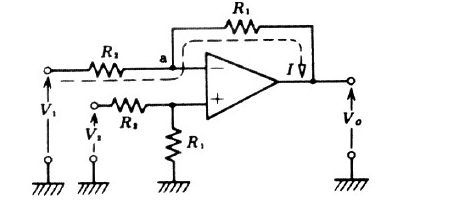
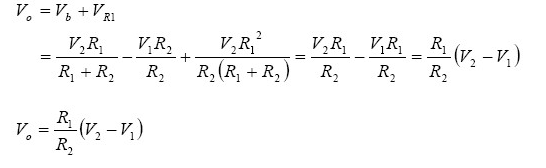


图7

## 元器件选择等

光纤接口选择普通FC/SC/ST等接口[2]如图8所示，无需复杂转换 。GaAs材料比较接近近红外。所以该光功率计采用InGaAs-PIN光电二极管、其波长响应范围是1.3~1.6 μm（如图9所示），刚好满足我们测量近红外的光功率的需求。放大部分使用的是LM741运算放大器LM741是一种高速度，高精度，高响应度的深度负反馈放大器价格低廉，适合于工作需要如图10所示、ADC0808模数转换器和STC89C516RD+高速单片机。

图8

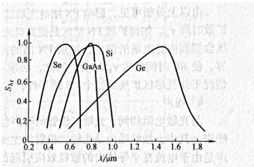
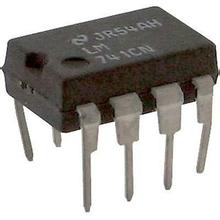


图9

图10

## 完整的系统原理电路图

系统完整的电路图如图11所示。

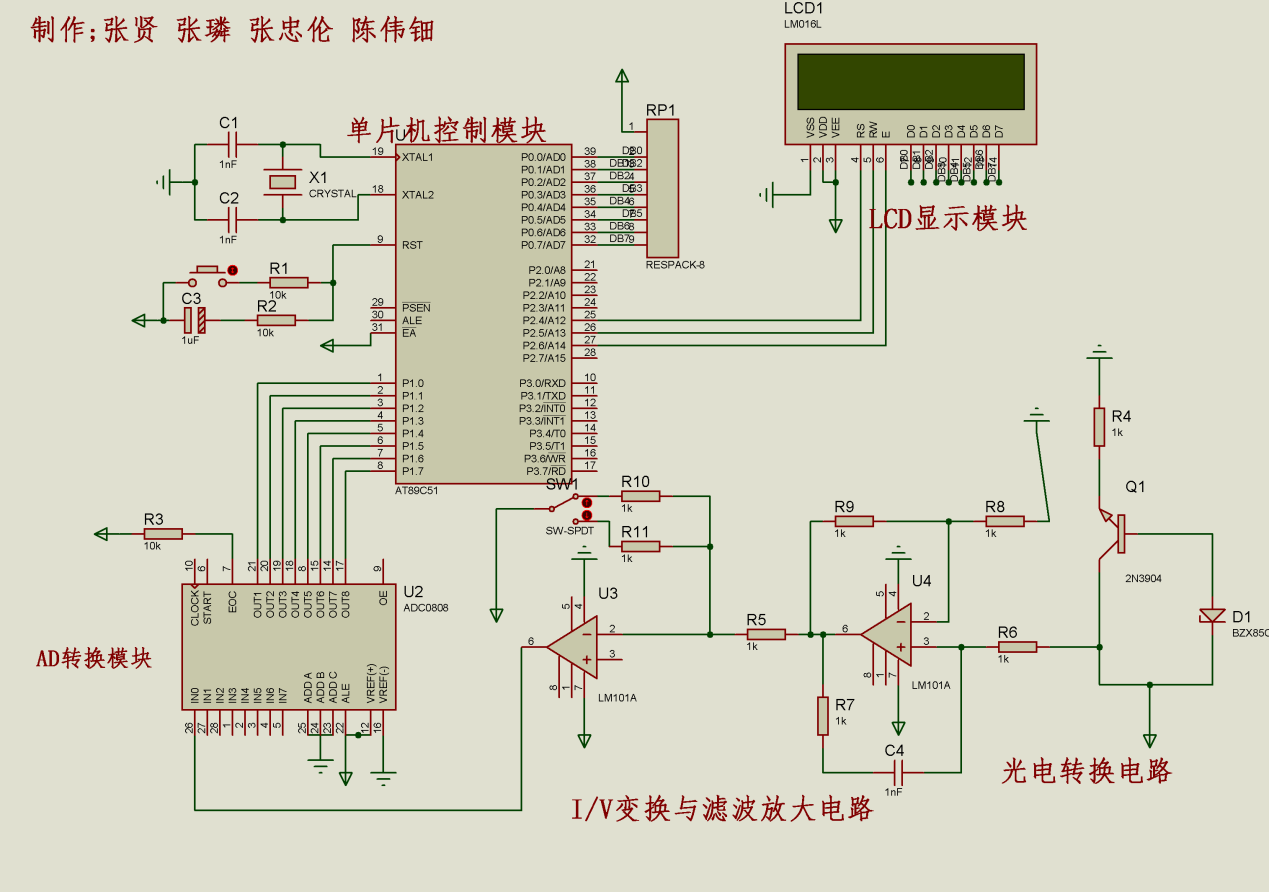


图11

## 所需的元器件清单

FC/SC/ST光纤接口。InGaAs-PIN光电二极管，三极管，LM741运算放大器、ADC0808模数转换器和STC89C516RD+高速单片机单片机,LCD模块，AT89C51，电解电容，晶振，陶瓷电容，排阻，ADC0808，LN741,多路开关，开关，按钮，电阻，电线等。

|  |  |  |
| --- | --- | --- |
| 元件名称 | 数量 | 数值 |
| FC/SC/ST光纤接口 | 1 | - |
| InGaAs-PIN | 1 | - |
| LM741运算放大器 | 1 | - |
| ADC0808模数转换器 | 1 | - |
| STC89C516RD | 1 | - |
| LCD模块 | 1 | - |
| 电解电容 | 1 | 1uF |
| 晶振 | 1 | 12MHz |
| 陶瓷电容 | 2 | 22pf |
| 排阻 | 1 | 220 |
| 多路开关 | 1 | - |
| 开关 | 若干 | - |
| 按钮 | 若干 | - |
| 电阻 | 若干 | 1k,220,10k |
| 电线 | 若干 | - |

## 仿真调试方案与步骤

1. 打开Proteus 8软件。点击菜单栏里的“Debug”,选择”Start VSM Debugging”.
2. 弹出一个”Digtial Oscilloscope”,再点击菜单栏里的“Debug”,选择“Run Simulation”,即开始模拟仿真。仿真结果如图

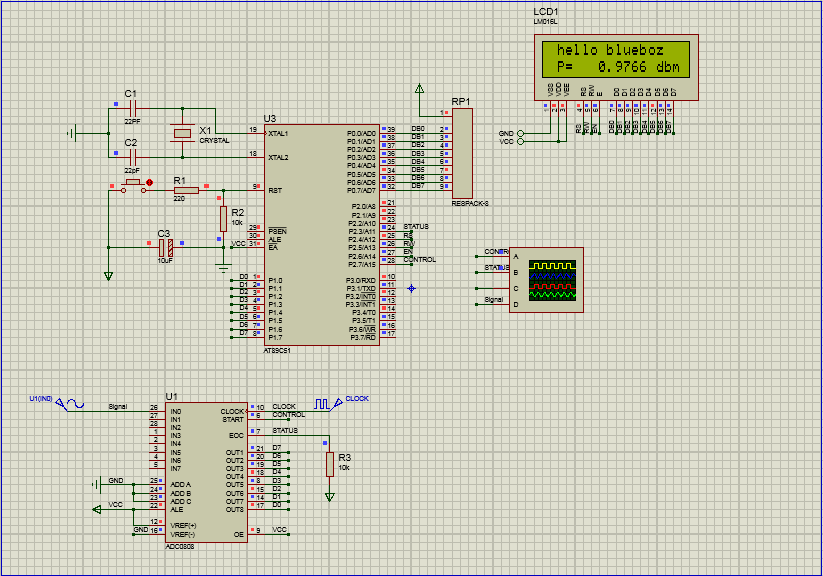


图13

因为光电模块不便仿真，而且误差较大，我们使用普通的正弦电压模拟经过放大的光电信号。首先从ADC模块开始，图中所示的信号源为正弦信号源，其电压值输入ADC的IN0，然后ADC将其模拟的电压值转化为数字量，从D0--D7输出,连接到单片机P1--P7接口，单片机读取数字量，转化为电压值。通过控制LCD模块显示（此时电压值改为dBm），所得绿线即为正弦信号源。

## 相应测量表格与测量结论

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 光电流（μA） | 放大后电压（V） | 实测数据（μW） | 参考功率计（μW） | 实测数据（dBm） | 参考功率计（dBm） |
| 1 | 0.081 | 55.81 | 55.23 | -12.51 | -12.57 |
| 1 | 0.146 | 111.73 | 97.16 | -9.53 | -10.11 |
| 2 | 0.211 | 153.64 | 140.5 | -8.14 | -8.47 |
| 3 | 0.279 | 195.51 | 182.3 | -7.09 | -7.45 |
| 3 | 0.327 | 237.43 | 210.1 | -6.21 | -6.72 |
| 4 | 0.361 | 265.31 | 240.2 | -5.78 | -6.17 |
| 4 | 0.414 | 293.28 | 278.9 | -5.31 | -5.55 |

表3

表3所列是1550nm时的测量结果。该测量结果与实际结果基本相同，光电流和功率的关系与设计初期所测量的数据相符。

## 完整的系统印刷电路板布线设计图

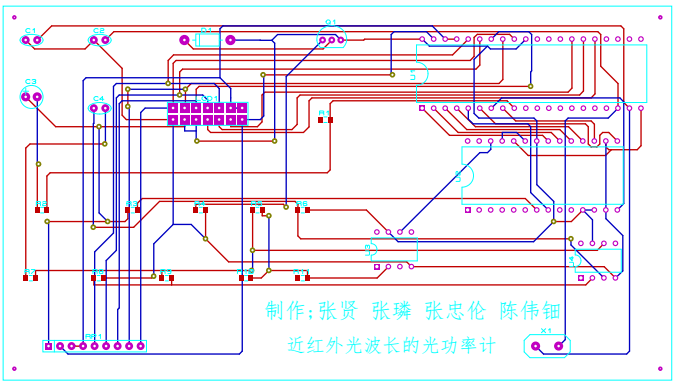


图14

## 程序设计部分

|  |
| --- |
| #include<reg52.h> //包含头文件，一般情况不需要改动，头文件包含特殊功能寄存器的定义  #include<intrins.h>  #include <stdio.h>  sbit RS = P2^4; //定义端口  sbit RW = P2^5;  sbit EN = P2^6;  sbit CTL= P2^7; //控制端口  sbit STATUS=P2^3;  unsigned char readData;  unsigned char DisTempData[16];  #define RS\_CLR RS=0  #define RS\_SET RS=1  #define RW\_CLR RW=0  #define RW\_SET RW=1  #define EN\_CLR EN=0  #define EN\_SET EN=1  #define DataPort P0  /\*------------------------------------------------  uS延时函数，含有输入参数 unsigned char t，无返回值  unsigned char 是定义无符号字符变量，其值的范围是  0~255 这里使用晶振12M，精确延时请使用汇编,大致延时  长度如下 T=tx2+5 uS  ------------------------------------------------\*/  void DelayUs2x(unsigned char t)  {  while(--t);  }  /\*------------------------------------------------  mS延时函数，含有输入参数 unsigned char t，无返回值  unsigned char 是定义无符号字符变量，其值的范围是  0~255 这里使用晶振12M，精确延时请使用汇编  ------------------------------------------------\*/  void DelayMs(unsigned char t)  {    while(t--)  {  //大致延时1mS  DelayUs2x(245);  DelayUs2x(245);  }  }  /\*------------------------------------------------  判忙函数  ------------------------------------------------\*/  bit LCD\_Check\_Busy(void)  {  DataPort= 0xFF;  RS\_CLR;  RW\_SET;  EN\_CLR;  \_nop\_();  EN\_SET;  return (bit)(DataPort & 0x80);  }  /\*------------------------------------------------  写入命令函数  ------------------------------------------------\*/  void LCD\_Write\_Com(unsigned char com)  {  // while(LCD\_Check\_Busy()); //忙则等待  DelayMs(5);  RS\_CLR;  RW\_CLR;  EN\_SET;  DataPort= com;  \_nop\_();  EN\_CLR;  }  /\*------------------------------------------------  写入数据函数  ------------------------------------------------\*/  void LCD\_Write\_Data(unsigned char Data)  {  //while(LCD\_Check\_Busy()); //忙则等待  DelayMs(5);  RS\_SET;  RW\_CLR;  EN\_SET;  DataPort= Data;  \_nop\_();  EN\_CLR;  }  /\*------------------------------------------------  清屏函数  ------------------------------------------------\*/  void LCD\_Clear(void)  {  LCD\_Write\_Com(0x01);  DelayMs(5);  }  /\*------------------------------------------------  写入字符串函数  ------------------------------------------------\*/  void LCD\_Write\_String(unsigned char x,unsigned char y,unsigned char \*s)  {  if (y == 0)  {  LCD\_Write\_Com(0x80 + x); //表示第一行  }  else  {  LCD\_Write\_Com(0xC0 + x); //表示第二行  }  while (\*s)  {  LCD\_Write\_Data( \*s);  s ++;  }  }  /\*------------------------------------------------  写入字符函数  ------------------------------------------------\*/  void LCD\_Write\_Char(unsigned char x,unsigned char y,unsigned char Data)  {  if (y == 0)  {  LCD\_Write\_Com(0x80 + x);  }  else  {  LCD\_Write\_Com(0xC0 + x);  }  LCD\_Write\_Data( Data);  }  /\*------------------------------------------------  初始化函数  ------------------------------------------------\*/  void LCD\_Init(void)  {  LCD\_Write\_Com(0x38); /\*显示模式设置\*/  DelayMs(5);  LCD\_Write\_Com(0x38);  DelayMs(5);  LCD\_Write\_Com(0x38);  DelayMs(5);  LCD\_Write\_Com(0x38);  LCD\_Write\_Com(0x08); /\*显示关闭\*/  LCD\_Write\_Com(0x01); /\*显示清屏\*/  LCD\_Write\_Com(0x06); /\*显示光标移动设置\*/  DelayMs(5);  LCD\_Write\_Com(0x0C); /\*显示开及光标设置\*/  }  float s;  /\*------------------------------------------------  主函数  ------------------------------------------------\*/  void main(void)  {  LCD\_Init();  LCD\_Clear();//清屏  CTL=1;  DelayMs(2000);  CTL=0;  while (1)  {  LCD\_Write\_String(1,0,"hello blueboz");  DelayMs(2000);  LCD\_Write\_String(1,1,DisTempData);  //STATUS=1;  //如果没有读取到STATUS为1，不让执行  while(STATUS!=1);  //如果读取到STATUS为1执行读取P1数据，  if(STATUS==1){  readData=P1;  P3=readData;  s=readData/256.0\*5.0;  sprintf(DisTempData,"P=%9.4f dbm ",s);//打印输出结果  CTL=1;  DelayMs(2000);  CTL=0;  DelayMs(2000);  }  }  } |

参考文献

[1] 耿涛，赵光，陶志勇.基于STC89C516单片机的数字光功率计的设计.电子元器件应用.2008,10(11):8-10

[2] 百度百科.百度百科.百度中国.2015

[3] 徐波.一种通用光功率计的实现原理.测试技术卷.2006,(5):3-7