

基于 MATLAB GUI 界面的 MCU 串口实时绘图设计

李华志 尹俊峰 王亚飞

中国地质大学（武汉）自动化专业 073082 班 湖北 武汉 430074

摘要: 介绍了基于MATLAB的GUI界面与单片机的串口实时通信并绘图的实现方法。通信系统主要由PC机、RS232C电平转换器、51单片机以及通信程序构成，单片机通信程序用C语言设计，PC机通信程序则是用MATLAB的GUI界面的事件和回调函数来实现。

关键词: MATLAB GUI 51单片机 串行通信 绘图

1 引言

在控制工程中，使用大量的数据采集系统，该类系统通常是微处理器(单片机)系统,其处理、分析数据的能力相对较低，一般是将这些数据传送到PC机，由相应的应用软件系统进行处理、分析和形成图表、打印，因此，两类系统必须要实现通信。MATLAB以其强大的科学计算与可视化功能、简单易用、开放式可扩展环境，特别是诸多的面向不同应用领域的工具箱支持，使其在信号处理、自动控制等许多领域获得广泛的应用，成为计算机辅助设计和分析、算法研究和应用开发的基本工具和首选平台^[1,2]。GUI为图形化界面，应用简单，不需要对MATLAB理解就可以应用^[3]。本文讨论应用MATLAB的GUI界面实现PC机与51单片机串行通信并实时绘制51单片机发送数据的图形,对于实时观察控制系统的运行状态和调节PID系数有着重要的应用。

2 总体设计原理

PC 机配置的RS232标准串行接口COM和单片机的串口连接,使用三线制(收、发、地)实现数据传递,用MAX232作为单片机与PC机间的电平转化芯片。串行通信总体设计原理图如图1所示。

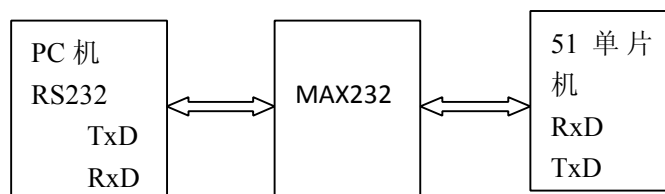


图 1 总体设计原理图

3 MATLAB 的 GUI 串行通信设计

GUIDE 是一个专用于GUI程序设计的向导设计器。而GUI 是由各种图形对象,如图形窗口、图轴、菜单、按钮、文本框等构建的用户界面,是人机交互的有效工具和方法,通过GUIDE可以很方便地设计出各种符合要求的图形用户界面。用户可以根据界面提示完成整个工程,却不必去了解工程内部是如何工作的。GUI 设计既可以基本的MATLAB 程序设计为主,也可以鼠标为主利用GUIDE工具进行设计,也可综合以上两种方法进行设计。

3.1GUI 界面的整体布局设计

通过 MATLAB 的 GUI 界面设计程序界面和功能,程序包括程序运行起始时间、实时时间显示、目标值(Goal Value)、实时数据显示、COM 口选择、波特率选择、打开串口(Open Serial)、停止绘图(Stop)、继续绘图(Go On)、关闭绘图窗口并删除串口设备(Clear)、保存数据(Save)、关闭串口(Off Serial)、关闭绘图串口并关闭 MATLAB(Quit)。如

图 2 所示。

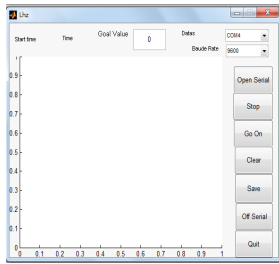


图 2 设计的 GUI 界面

3.2 MATLAB 对串口的控制

MATLAB 对串口的控制操作过程如下^[4,5]:

- ①首先要创建串行口设备对象，这里用到的主要函数是 `serial()`；
`s=serial('COM1');`%创建 COM1 为串口对象并用 s 标识。
- ②定义串口设备对象的通信属性，主要有波特率、异步串行帧格式（包括起始位、数据位、校验位、停止位等）和输出输入缓冲的大小等；
- ③用函数 `fopen()` 实现设备对象与硬件设备的软连接（相对于硬件设备的连接），这样就可以进行数据的读写操作；
`fopen(s)` %打开 s 标识的串口；
- ④进行 MATLAB 串行读写操作，主要用到的函数有 `fwrite()`、`fread()`等；
- ⑤关闭和删除设备对象，用到的是 `fclose()`、`delete()`和 `clear()`等函数。

3.3 各个控件的回调函数

(1)Open Serial控件的回调函数

```
function Open_Serial_Callback(hObject, eventdata, handles)
global s;%定义全局变量s,用于标识串口
global datas;%定义全局变量datas,用于存储串口发送来的数据
global out;%定义全局变量out,用于标识是否关闭串口，out=1不关闭串口，out=0关闭串口
global rate;%定义全局变量rate,用于表示串口通信的波特率
global COM;%定义全局变量COM，用于标识选取的COM口
out=1;
%通过Select_COM下拉条控件改变全局变量COM的值，根据COM的值选取打开的COM口
if(COM==1)
s=serial('COM1');
elseif(COM==2)
s=serial('COM2');
elseif(COM==3)
s=serial('COM3');
elseif(COM==4)
s=serial('COM4');
end
%通过BaudRate下拉条改变全局变量rate的值，根据rate的值选取串口所对应的波特率
if(rate==1)
set(s,'BaudRate',19200);
```

```

elseif(rate==2)
set(s,'BaudRate',9600);
    end
fopen(s);%打开串口
k=1;
datas=zeros([1,100]);%将全局变量datas赋初始值为1*100的零矩阵
a=num2str(datestr(now,'HH:MM:SS'));%获得实时时间并转换为字符型存在a变量中
set(handles.Start_time,'String',a);%将字符型a显示在静态存储Start中
guidata(hObject, handles); %更新结构体
while(out==1)
    if s.BytesAvailable %查询是否接收到数据，接收到数据s.BytesAvailable=1
datas(1,k)=fread(s,1);%以二进制读取串口s中的数据并存储在datas中
        end
        plot(datas);%绘制数组datas的图形
        d = get(handles.Value,'String');%获取Value动态存储区中输入的值，即目标值
        y=str2num(d);%将字符型d转换为数字型，并存储在y中
        e=size(datas);%定义e为与datas形同形式的数组
        i=1;
        while(i<=k) %将e数组赋值全部为y,即为目标值
            e(1,i)=y;
            i=i+1;
        end
        hold on %保持接收到的数据绘制的图，并绘制目标值的图形
        plot(e,'-')
        hold off
b=num2str(datestr(now,'HH:MM:SS'));%实时更新显示时间
set(handles.Time,'String',b);
guidata(hObject, handles); %更新结构体
        c = num2str(datas(1,k)); %将实时数据存储在c中
        set(handles.Datas,'String',c);%在静态存储区Datas中显示串口实时数据
        guidata(hObject, handles); % 更新结构体
        k=k+1;
    end %这里结束时关闭了串口设备out不等于1
fclose(s);%关闭串口设备
delete(s);%删除串口连接
clear s;
(2)停止显示（Stop）回调函数
function Stop_Callback(hObject, eventdata, handles)
uiwait %等待
(3)继续显示（Go On）回调函数
function Go_On_Callback(hObject, eventdata, handles)
uiresume %继续显示
(4)关闭绘图窗口并删除串口设备（Clear）回调函数
function Clear_Callback(hObject, eventdata, handles)

```

```

clear
close all %关闭GUI界面窗口
clc
(5)保存数据（Save）回调函数
function Save_Callback(hObject, eventdata, handles)
global datas;%声明全局变量datas
dlmwrite('a.txt',datas,'\t');%将datas中数据以txt格式存储在a.txt中
(6)关闭串口（Off Serial）回调函数
function Off_Serial_Callback(hObject, eventdata, handles)
global out;%声明全局变量out
out=0;%out=0为退出绘图循环，关闭并删除串口设备
(7)目标值（Goal Value）回调函数 主要用于显示真实值
function Value_Callback(hObject, eventdata, handles)
input = str2num(get(hObject,'String'));
if (isempty(input))
    set(hObject,'String','0')
end
guidata(hObject, handles);
(8) COM口选择回调函数
function Select_COM_Callback(hObject, eventdata, handles)
global COM;%声明全局变量COM
COM=1;%COM赋初始值为1
rate=1;%rate赋初始值为1
val=get(hObject,'value');%将下拉条value的值赋给val
%通过val值选择COM的值
switch val
    case 1
        COM=1;
    case 2
        COM=2;
    case 3
        COM=3;
    case 4
        COM=4;
end
(9)波特率选择回调函数
function Baude_Callback(hObject, eventdata, handles)
global rate;%声明全局变量rate
val=get(hObject,'value');%将下拉条value值赋给val
%通过val的值选择rate 的值
switch val
    case 1
        rate=1;
    case 2

```

```

        rate=2;
end

```

4 单片机串口程序

4.1 串口初始化程序

初始化串口波特率为 19200，方式 1，10 位异步发送，禁止接收^[6,7]。

```

void InitI_series(void)
{
    TMOD=0X20; //波特率 19200
    TH1=0XFD; //Fd
    TL1=0XFD; //FD
    SCON=0X40; //方式 1,10 位异步发送，禁止接收
    PCON=0X80; //波特率加倍
    TR1=1;      //开定时器 1
}

```

4.2 单片机向 PC 机发送数据函数

通过数组 buf_pc 发送数据，len 为发送数据的长度。

```

void Send_to_PC( uchar *buf_pc,uchar len )
{
    unsigned int i=0,j=0;
    while( i<len )    //判断字符是否发送完
    {
        SBUF=*(buf_pc+i);
        while(TI==0); //判断中断
        TI=0;
        i++;
    }
}

```

5 总结

由 MCU 构成的控制系统具有价格低廉、编程简单、工作可靠等诸多优点，在控制领域有着重要的应用。MATLAB 的 GUI 为图形用户界面，具有应用简单，不需要对程序有很深的了解就可以应用的特点，可以大大的减轻用户的工作量。基于 MATLAB 的 GUI 界面的 MCU 串口实时绘图的设计，可以通过 MCU 的串口与 PC 机实时通信并绘制图形并可以在图形中标识期望的目标值直线，对于观察控制系统的工作过程和调节 PID 参数有着重要的应用。

参考文献

- [1] 吴晓燕，张双选. MATLAB 在自动控制中的应用[M]. 西安：西安电子科技大学出版社，2006.9.
- [2] 李国勇，谢克明，杨丽娟. 计算机仿真技术与 CAD：基于 MATLAB 的控制系统[M]. 北京：电子工业出版社（第 2 版），2008.
- [3] 应亮. Matlab 的图形用户界面的应用[J]. 电脑知识与技术，2005，（11）： 75-76.
- [4] 张道明，郝继飞. 基于 MATLAB 的 MCU 串行通信[J]. 网络与通信，2004，（5）： 1-3.
- [5] 兰红莉，罗文广. 基于 MATLAB 的 PC 机与单片机串行通信实现[J]. 计算机应用与软件，2006,23（6）： 74-76.
- [6] 汪文，陈林. 单片机原理与应用[M]. 武汉：华中科技大学出版社，2008. 75-101.
- [7] 张义和. 例说 51 单片机(C 语言版. 第 3 版)[M]. 北京：人民邮电出版社，2010. 208-237.
