## 实验九 DMA模块实验

直接存储器存取（Direct Memory Access，DMA）是一种数据传输方式，该方式可以使数据不经过CPU直接在存储器与I/O设备之间、不同存储器之间进行传输。这样的好处是传输速度快，且不占用CPU的时间。DMA是所有现代微控制器的重要特色，它实现存储器与不同速度外设硬件之间进行数据传输，而不需要依于 CPU过多介入。DMA传输对于高效能嵌入式系统和网络是很重要的，通常只有数据流量较大的外设才需要有支持DMA能力，例如视频、音频和网络等接口。

### 一、实验目的

（1）了解DMA模块基本概念和基本工作原理。

（2）理解和掌握微控制器的DMA模块的基本编程方法。

（3）进一步深入理解MCU和C#串口通信的编程方法。

### 二、实验准备

（1）硬件部分：PC机或笔记本电脑，一台；开发套件或实验箱一套。

（2）软件部分：集成开发环境、VS2012开发平台及其高版本软件。

（3）在认真学习教材相应章节基础上，仔细阅读下面的实验要求，做好实验准备。

### 三、实验要求

#### 1．验证性实验

验证样例程序（ch12-DMA），主要功能是实现通过开发板上串口输出10字节字符串，MCU存放由DMA传输来的数据的数组将为传输值，同时通过DMA回发至PC机。

实验步骤如下：

##### 1）拷贝样例工程并重命名

拷贝“\04-Soft\ CH09-DMA\DMA”工程到自己的工作文件夹，改为自己确定的工程名，建议尾端增加“-190501”字样，表示日期，避免混乱。

##### 2）导入工程、编译、下载到GEC中

（1）导入工程。运行“..\05-Tool\AHL-GEC-IDE\AHL-GEC-IDE.exe”文件，这一步打开了集成开发环境AHL-GEC-IDE。接着单击“”→“” →导入你拷贝到自己文件夹并重新命名的工程。

（2）编译工程。在打开工程，并显示文件内容前提下，可编译工程。单击“”→“”，则开始编译。

（3）下载并运行。步骤一，硬件连接。用USB-TTL线（Micro口）连接GEC底板上的“MicroUSB”串口与电脑的USB口。步骤二，软件连接。单击“”→“”，将进入更新窗体界面。点击“”查找到目标GEC，则提示“成功连接……”。步骤三，下载机器码。点击“”按钮导入被编译工程目录下Debug中的.hex文件，然后单击“”按钮，等待程序自动更新完成。下载之后，程序自动运行。

##### 3）观察实验现象

（1）将“TTL-USB串口线”的“USB端口”接PC机的USB口，串口线的串口接开发板上UART0（3根，RX接蓝线，TX接白线，GND接黑线）；

|  |  |
| --- | --- |
| 图1 底板 | 图2 TTL-USB串口线 |

（2）正确连接串口后，打开串口调试工具或ch06-UART文件夹中的“C#串口测试程序”进行串口通信测试；

（3）在串口输出“Start DMA transmission”信息后表示DMA已启动，在输入框中输出数据长度为10的字符串并点击发送。DMA将接收到的数据后回传至PC端，如图3所示。

|  |
| --- |
| 图3 DMA信号传输 |

（4）分析理解main.c程序和中断服务例程isr.c。

##### main.c

主函数首先初始化Uart与DMA。

|  |
| --- |
| //（1.6）用户外设模块初始化  dma\_init(DMA\_CH0);  uart\_init(UARTA,9600); |

并且将串口使能为空闲中断。

|  |
| --- |
| //（1.7）使能模块中断  uart\_IdleIrq\_enable(UARTA);//开串口空闲中断 |

在主循环中判断DMA完成接收标志，若有数据则进行回发。

|  |
| --- |
| for(;;)  {  ……  //主循环中检测完成接收标志进行数据回发  if(gDmaRecFlag)  {  uart\_sendN(UARTA,25,gDmaSendBuf);  //使用DMA发送内存中接收到的数据进行验证  Delay\_ms(5);  dma\_send(DMA\_CH0,(uint\_32)gDmaRecBuf,10);//通过DMA发送  gDmaRecFlag=0;//标志位清0  }  } |

##### isr.c

UARTA空闲中断在接收到结束符并且没有信号传入时进入中断，在空闲中断处理程序中发起一次DMA传输请求（完成上一次的DMA传输后重启），并且清中断标志位使主函数发送该帧数据。

|  |
| --- |
| //======================================================================  //程序名称：UARTA空闲中断处理程序  //触发条件：UARTA接收到结束符并且没有信号传入。  //======================================================================  void UARTA\_Handler(void)  {  uint\_8 ch;  uint\_8 flag;  DISABLE\_INTERRUPTS; //关总中断  //------------------------------------------------------------------  Delay\_ms(10);  dma\_read(DMA\_CH0,(uint\_32)gDmaRecBuf,10);//请求DMA进行传输数据  gDmaRecFlag=1;//标志位置位  uart\_IdleIrq\_clear(UARTA);//清中断标志位  //------------------------------------------------------------------  ENABLE\_INTERRUPTS; //开总中断  } |

#### 2．设计性实验

（1）复制样例程序（UART），利用该程序框架实现串口的DMA编程方法，通过串口调试工具或ch06-UART文件夹中的“C#串口测试程序”进行串口通信测试传输数据。

请在实验报告中给出MCU端程序main.c和isr.c流程图及程序语句。

（2）复制样例程序（SPI），利用该程序框架实现SPI的DMA编程方法，采用芯片的不同SPI模块间进行通讯测试，通过串口调试工具或ch06-UART文件夹中的“C#串口测试程序”进行测试数据的发送和接收。

请在实验报告中给出MCU端程序main.c和isr.c流程图及程序语句。

（3）复制样例程序（AD），利用该程序框架实现AD数据采集的DMA编程方法，通过串口调试工具或ch06-UART文件夹中的“C#串口测试程序”进行测试数据的发送和接收。

请在实验报告中给出MCU端程序main.c和isr.c流程图及程序语句。

#### 3．进阶实验★

（1）复制样例程序（ch09-Flash）中（Flash），利用该程序框架，采用DMA编程方法实现：通过C#程序打开一副图片（比如自己的一寸电子照片），通过串口（也可以采用其他接口）将图片数据发送至MCU并保存,lash中；通过C#程序可以将MCU的Flash中保存的图片读取并显示。C#界面设计如如图1所示。

请在实验报告中给出MCU端程序main.c和isr.c流程图及程序语句和C#方主要程序段。



图1 C#界面设计

### 四、注意事项

（1）实验前应该首先确认所使用硬件设备的正常与否。MCU最小系统的测试、小灯测试、UART的测试方法如附录C所述，否则，将难以确认问题所在。

（2）送电实验前，先将所编写的程序编译后，下载至MCU中，接着将所需部件连接至开发板，然后送电。注意：不要带电操作，特别是**接串口线，一定不要带电接**。

（3）MCU程序中使用的串口号要和实际开发板上所接的串口要一致，并注意 “TTL-USB串口线”的“USB端口”接PC机的USB口，串口线的串口接开发板上的串口（3根，RX接蓝线，TX接白线，GND接黑线）。

### 五、实验思考题

（1）本书微控制器芯片的哪些模块具备DMA编程方法，请说出他们的使用方法？

（2）一般什么情况需要采用DMA编程方法？