基于 NiosII 与 PC 数据通信的源码实现

Author：Itspy 独舞

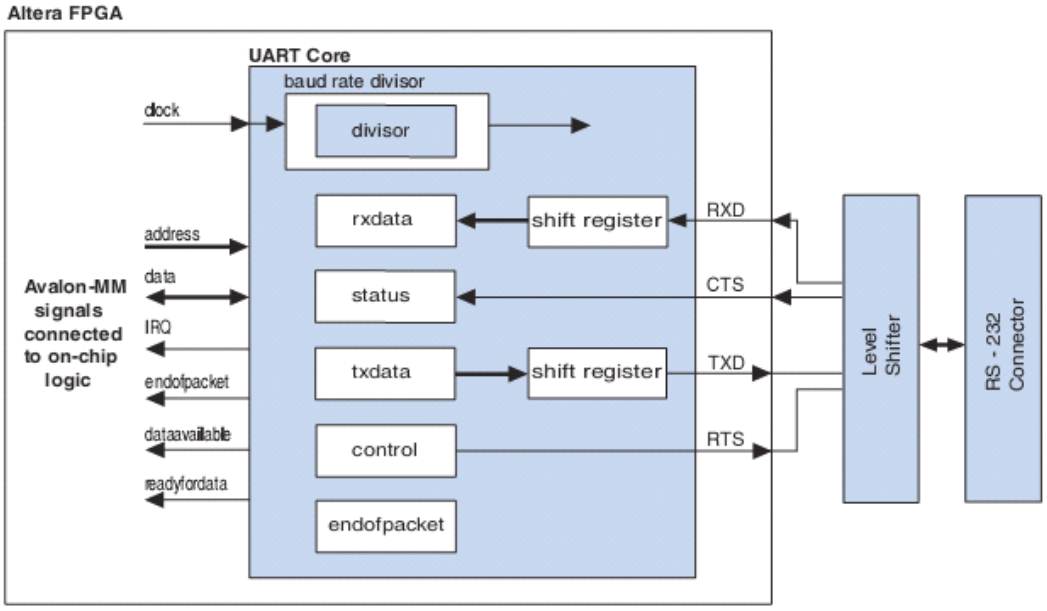
利用 Altera 公司的 UART core 即可和 PC 进行数据通信。关于 UART core ，Altera 公司有关的资料是这样描述的：

The universal asynchronous receiver/transmitter core with Avalon® interface (UART core) implements a method to communicate serial character streams between an embedded system on an Altera® FPGA and an external device. The core implements the RS-232 protocol timing, and provides adjustable baud rate, parity, stop and data bits, and optional RTS/CTS flow control signals. The feature set is configurable, allowing designers to implement just the necessary functionality for a given system.

大概翻译如下（若有不对，谢谢指正）：
通用异步 收/发 核通过与 Avalon 总线接口为基于 Altera FPGA 芯片和外部设备所组成的嵌入式系统提供了一个实现字符流式串行通信的方法。该核遵行 RS-232 协议时序，且提供程序可设置的波特率，奇偶校验，停止位，数据位，还有 RTS/CTS 流控制信号。这些功能设置是可配置的，从而允许设计者根据自己所设计的系统选择所需要的功能。

下面是 UART 的结构图：

Figure 9–1. Block Diagram of the UART Core in a Typical System



因为 PC 中文件是基于字节存储的，这样通过 UART core 与 PC 进行文件的传输与通信，如传输 ASCII 文档、图像数据等。

源码实现：

```
//  此程序基于 NiosII 传送二进制文件或 ASCII 文件
//  上位机测试软件为  Serial Port Monitor
```

```
//   Create by itspy                                                    qq：275885635

#include "stdio.h"
#include "string.h"
#include "system.h"
#include "sys/alt_irq.h"
#include "altera_avalon_uart_regs.h"
#define   Freqc       75000000    //Cpu clock
unsigned long Index=0;
unsigned char Mem[800*600] ; //Data array for an image (800*600)
static void Uart_ISR( void *context,alt_u32 id);
void InitDev();
int main ()
{
    InitDev();
    while(1) printf("%lu,%d\n",Index;Mem[Index-1]);    // Display the data in IDE
    return 0;
}

static void Uart_ISR(void *context,alt_u32 id)
{
  unsigned char tmp;
  context=context;
  tmp=IORD_ALTERA_AVALON_UART_RXDATA(UART_BASE);    //Receive data
  Mem[Index++]=tmp;                 //Store the receive data
  //IOWR_ALTERA_AVALON_UART_TXDATA(UART_BASE,tmp);    //Transmit data
  IOWR_ALTERA_AVALON_UART_STATUS(UART_BASE,0); //Clear STATUS register
  IOWR_ALTERA_AVALON_UART_CONTROL(UART_BASE,ALTERA_AVALON_UART_
  CONTROL_RRDY_MSK);    //Enable interrupt for read ready
}
void InitDev()
{
  float Dvs=0.0;
  Dvs=IORD_ALTERA_AVALON_UART_DIVISOR(UART_BASE);
  Dvs=(Freqc/9600)+0.5;         // divisor=int(clock frequency)/baud rate +0.5
  IOWR_ALTERA_AVALON_UART_DIVISOR(UART_BASE,Dvs); //Set baud rate:9600bps
  IOWR_ALTERA_AVALON_UART_STATUS(UART_BASE,0); //Clear STATUS register
  IOWR_ALTERA_AVALON_UART_CONTROL(UART_BASE,ALTERA_AVALON_UART_
  CONTROL_RRDY_MSK);    //Enable interrupt for read ready
  alt_irq_register(UART_IRQ,NULL,Uart_ISR);    //Register interrupt routine
}
```

Friday, March 28, 2008