通过串口读取并写入内存

一.流程说明

1.串口控制器初始化（已在preloader中初始化）

2.屏蔽串口所有类型中断

3.使能fifo

4.将接受串口读取数据的内存清零

5.进入读取串口循环，每个循环将fifo中的数据全部读到内存中

6.判断读取是否结束，结束后打印接受到的字节数

二.测试代码示例

以下代码是一个用串口测试工具测试的代码片段，包含步骤2到6，串口工具发送的数据全部接收到，没有丢失。

```
/* Receive Buffer Register,Transmit Holding Register,Divisor Latch Low */
#define RBR_THR_DLL 0xFFC02000

/* This register enables/disables receive and transmit interrupts and also
controls the most-significant 8-bits of the baud rate divisor */
#define IER_DLH 0xFFC02004

/* Returns interrupt identification and FIFO enable/disable when read */
#define IIR 0xFFC02008

/* Controls FIFO Operations when written. */
#define FCR 0xFFC02008

/* Formats serial data */
#define LCR 0xFFC0200C

/* Reports status of transmit and receive. */
#define LSR 0xFFC02014

/* indicates the number of data entries in the transmit FIFO. */
#define TFL 0xFFC02080

/* Indicates the number of data entries in the receive FIFO. */
#define RFL 0xFFC02084

/* 从串口接收到的数据要写到的内存地址 */
```

```
#define YESHEN_MEM_ADDR 0x30000000

#define rYESHEN_MEM_ADDR(offset) (*(volatile unsigned char*)(YESHEN_MEM_ADDR + offset))


/* 将fifo中的数据写入地址0x30000000开始连续的内存中 */
static void readintobuf_yeshen(int cnt)
{
    int i;
    unsigned char ch;
    static unsigned char *p = YESHEN_MEM_ADDR;
    for(i=0; i<cnt; i++,p++){
        ch = readb(RBR_THR_DLL);
        writeb(ch, p);
    }
}

/* 步骤2到6的测试函数 */
void uart_mem_test(void)
{
    /* for yeshen test uart,a fifo, muti size watermark and no interrupt version */
    int val, cnt = 0, max = 0, num = 0;

    writel(0x0, IER_DLH);           //屏蔽所有串口中断

    writel(0x1, FCR);       //使能fifo

    if((readl(IIR) & 0xC0) == 0xC0)     //表示fifo确实已使能
        printf("\nfifo enable\n");

    memset(YESHEN_MEM_ADDR,0,YESHEN_MEM_SIZE);  //将接收串口数据的内存清零
    printf("begin reading uart\n");
    for (;;) {
        val = readl(RFL);               //读取fifo中当前有多少字节数据
        if(val > max){                  //实时维护fifo最大值
            max = val;
            printf("max data of fifo now:%d\n",max);
        }

        if(val > 0){
            readintobuf_yeshen(val);    //读取当前fifo中的所有数据到指定内存
            num += val;
            cnt = 0;
        }else{
            cnt += 1;
        }

        if(cnt > 0xFFFFFF){         //接收结束，打印所收到的数据字节数以便和串口调试工具比较
            printf("there is no data from uart\n");
            printf("the total number recevied from uart are %d bytes\n",num);
            rYESHEN_MEM_ADDR(10) = 0;
            puts(YESHEN_MEM_ADDR);  //打印前10个字节
            hang_yeshen();
```

```
        }
    }
}
```