

iCore 开发板 FSMC 读写 FPGA 说明 V0

一、 ARM 程序介绍

1. 写函数

```
#define fpga_write(offset, data) *[(volatile unsigned short int*)(CSO_BASE +  
(offset << 17))] = data
```

其中：

- offset 为空间偏移量，范围为 0~7 (8 个空间)；
- data 为数据，16 位整数，范围为 0~65535；
- CSO_BASE 为 ARM 连接 FPGA cs0 的基地址，具体内容详见原理图。

若向空间 2 写入 1234，则调用方法为：

```
fpga_write(2,1234);
```

2. 读函数

```
#define fpga_read(offset) *[(volatile unsigned short int*)(CSO_BASE +  
(offset << 17))]
```

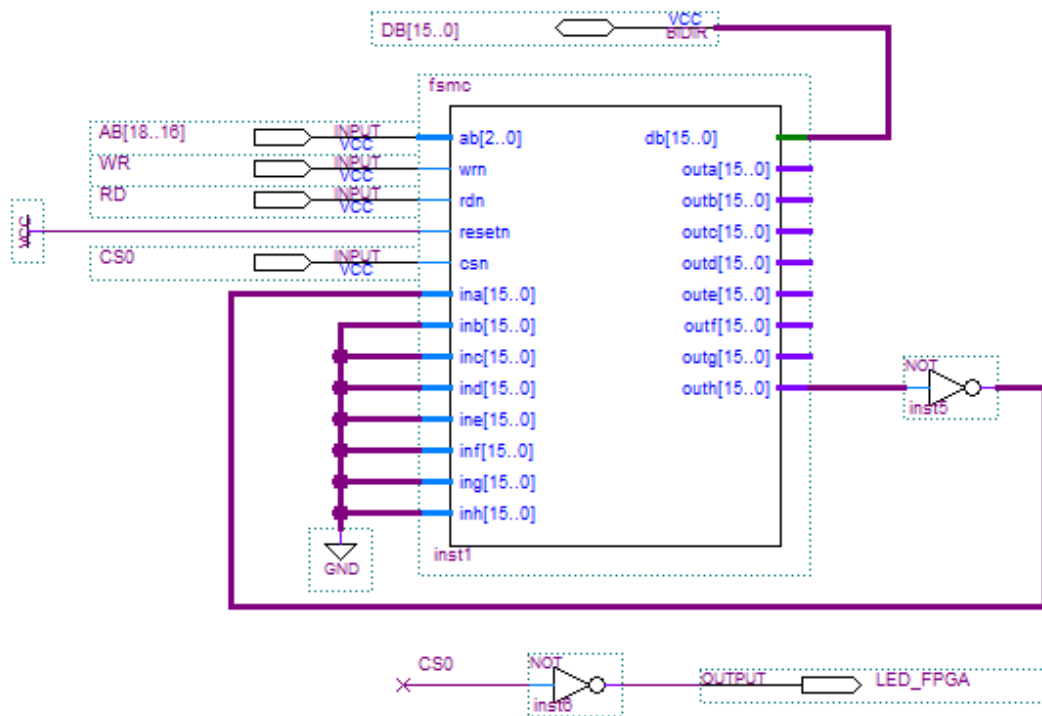
其中：

- offset 为空间偏移量，范围为 0~7 (共 8 个空间)；
- 范围值为 16 位证书，范围为 0~65535；
- CSO_BASE 为 ARM 连接 FPGAcso 的基地址，具体内容详见原理图。

若读取空间 2 的值，则调用方法为：

```
unsigned short int read_data;  
  
read_data=fpga_read(2);
```

二、FPGA 程序介绍



图示为 FSMC 代码生成的顶层模块连接示意图，其中：

- DB[15..0] AB[18..16] WR RD CS0 连接到 ARM 上；
- LED_FPGA 用做通信指示；
- ina[15..0] ~ inh[15..0] 为读取数据输入，共包含 8 个空间，每个空间 16 位；
- outa[15..0]~outh[15..0]为写入空间寄存器，共 8 个空间，每个空间 16 位；

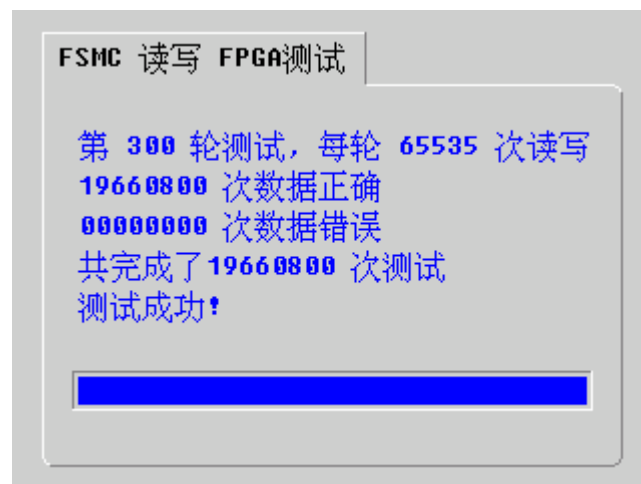
具体实现原理请参考 fsmc.v 程序。

三、 功能测试

如上图所示，FPGA 内把 outh[15..0] 输出的数据，取非后连接到 ina[15..0]，ARM 程序会写入空间 7（也就是 outh[15..0]），然后读取空间 0（也就是 ina[15..0]），然后做数据取非校验，判断成功与否。每轮测试 65536 次，共进行 300 轮测试。核心代码如下：

```
for (i = 0; i < 65536; i++) {  
    fpga_write(7, i);  
  
    dat = fpga_read(0);  
  
    if ((dat + i) != 65535) {  
        errors++;  
    }else success++;  
}
```

测试时，FPGA LED 会一直闪烁，直到测试成功，若 iCore 连接到扩展板上，则会出现以下显示界面：



E.V.Studio

QQ 群：

[Group A] 204255896 (500 人，满)

[Group B] 165201798 (500 人，满)

[Group C] 215053598 (200 人，满)

[Group D] 215054675 (200 人高级群)

[Group E] 215055211 (200 人，满)

[Group F] 78538605 (200 人高级群)

[Group G] 158560047 (500 人，满)

淘宝：<http://i-board.taobao.com>

博客：<http://XiaomaGee.cnblogs.com>

论坛：<http://www.heijin.org>

Copyright ©2010-2013 E.V. Stdio All Rights Reserved.