**AXI LCD控制器设计**

========================

|| 控制接口：AXI-Lite ||

|| 数据接口：AXI-Full ||

========================

**控制过程：**

处理器配置好目标图像的开始和结束地址后，装在图像只需要发送一个装载命令，判断是否装载完成通过读取状态寄存器是否等于装载完成字，如果是的话则**软件Disassert掉**。

**//Configuration**

Processor: slv\_reg\_endAdrr=0x1000\_0000;

Processor: slv\_reg\_startAddr=0x1000\_FFFF;

**//Control Process**

Processor: slv\_reg\_command=Load\_PIC;

Processor: while(slv\_reg\_status！=SLOAD\_PIC\_DONE);

Processor: slv\_reg\_command=DisassertStatus;

**//Control Process End**

**系统框图**



**设计核心规范**

1. CPU通过四个寄存器进行与AXI\_LCD控制器通信，CPU发送的命令必须由命令状态机（Command Finite State Machine, CFSM\_state）进行译码转发给AXI\_Full状态机和LCD\_Core状态机(LCDC\_state)，同样地，AXI\_Full和LCD\_Core的状态必须由命令状态机进行编码，供CPU查询。
2. 设备中不做大容量的片上缓存，仅仅只缓存一个Burst（因为AXI\_Full的读取速度比LCD\_Core的速度快很多倍，所以必须缓存）。
3. 命令规范-附表。

**命令状态机(CFSM\_State)状态**

|  |  |  |
| --- | --- | --- |
| 状态 | 内涵 | 命令 |
| CFSM\_IDLE | 等待命令 |  |
| CFSM\_DISASSERT | 重写状态字 | DISASSERT |
| CFSM\_INIT\_AXIF | 初始化AXI-FULL总线 | INIT\_AXIF |
| CFSM\_INIT\_AXIF\_W | 等待AXI-FULL总线初始化完成 |  |
| CFSM\_INIT\_LCD | 初始化LCD | INIT\_LCD |
| CFSM\_INIT\_LCD\_W | 等待LCD初始化完成 |  |
| CFSM\_LOAD\_PIC | LCD装载图像 | LOAD\_PIC |
| CFSM\_LOAD\_PIC\_W | 等待LCD装载图像完成 |  |

**命名规范**

|  |  |  |
| --- | --- | --- |
| Input(AXI\_Lite.v) | Status | Status from AXI\_Full and LCD\_Core |
| (AXI\_Lite.v) | Slv\_reg\_Status\_w | CPU write status register |
| (AXI\_Lite.v) | Slv\_reg\_status\_r | CPU read status register |
| Output(AXI\_Lite.v) | Cmd | Command to AXI\_Full and LCD\_Core |
| (AXI\_Lite.v) | Slv\_reg\_cmd\_w | CPU write command register |
| (AXI\_Lite.v) |  |  |