文档ID :

MC-20芯片FPGA原型验证SPEC

**作 者: 张 坤**

**审 查:**

**批 准:**

**北京爱芯科技有限公司**

**SoC部 – FPGA原型验证组**

**修改记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **描述** | **日期** | **修改人** |
| V0.1 | 创建文档初稿 | 2020.06.31 | 张坤 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[MC-20芯片FPGA原型验证SPEC 1](#_Toc44522619)

[1 验证概述 4](#_Toc44522620)

[2 验证目标 4](#_Toc44522621)

[3 验证需求 4](#_Toc44522622)

[3.1.1 时钟需求 4](#_Toc44522623)

[3.1.2 接口/IP需求 5](#_Toc44522624)

[3.1.3 子卡需求 5](#_Toc44522625)

[3.1.4 子系统需求（小版本） 6](#_Toc44522626)

[3.1.5 整系统需求（大版本） 7](#_Toc44522627)

[4 验证方案 7](#_Toc44522628)

[4.1 时间安排 7](#_Toc44522629)

[4.1.1 代码移植：（3周）（7.10~7.31） 7](#_Toc44522630)

[4.1.2 建立仿真环境：（2周）（8.1~8.14） 7](#_Toc44522631)

[4.1.3 最小系统：（2周）（8.14~8月底） 8](#_Toc44522632)

[4.1.4 IP/接口集成：（2~3个月）（11月底） 8](#_Toc44522633)

[4.1.5 子系统验证：（1~2个月）（10月底） 8](#_Toc44522634)

[4.2 移植方案 8](#_Toc44522635)

[4.2.1 子卡设计： 8](#_Toc44522636)

[4.2.2 资源评估： 9](#_Toc44522637)

[4.2.3 划片方案： 9](#_Toc44522638)

[4.2.4 PIN设计： 11](#_Toc44522639)

[4.2.5 时钟方案： 11](#_Toc44522640)

[4.3 接口验证方案 11](#_Toc44522641)

[4.3.1 JTAG接口验证 11](#_Toc44522642)

[4.3.2 Spi flash接口验证 11](#_Toc44522643)

[4.3.3 UART接口验证方案 12](#_Toc44522644)

[4.3.4 I2C接口验证 12](#_Toc44522645)

[4.3.5 I2S接口验证 12](#_Toc44522646)

[4.3.6 DDR接口验证方案 12](#_Toc44522647)

[4.3.7 CSI接口验证方案 13](#_Toc44522648)

[4.3.8 DSI接口验证方案 13](#_Toc44522649)

[4.3.9 EMAC接口验证方案 14](#_Toc44522650)

[4.3.10 USB接口验证方案 14](#_Toc44522651)

[4.3.11 SDIO接口验证方案 14](#_Toc44522652)

[4.4 子系统FPGA原型方案 15](#_Toc44522653)

[4.4.1 Base\_sys最小系统验证方案 15](#_Toc44522654)

[4.4.2 ISP\_sys验证方案 16](#_Toc44522655)

[4.4.3 NPU\_sys验证方案 16](#_Toc44522656)

[4.4.4 Video\_sys验证方案 16](#_Toc44522657)

[4.4.5 DPU\_sys验证方案 16](#_Toc44522658)

[4.5 整系统级验证方案 16](#_Toc44522659)

# 验证概述

此验证文档目的是指导和规范MC20芯片的整个FPGA原型验证工作。主要对芯片的系统功能，根据芯片的系统行为、时序和功能，进行连续的、单向的、长时间的系统行为的时序功能验证，以满足芯片的系统性的交互、功能和时序要求。

# 验证目标

MC20芯片FPGA原型验证的目标包括：

* 芯片外设接口模块的功能及时序；
* 系统原型的建立，为软件bootloder及操作系统的移植跟调试提供可靠的硬件平台，加速上层软件的开发
* 验证各子系统的功能的正确性

考虑到MC20芯片架构对FPGA资源的需求、对MC40芯片验证成果及经验的继承以及为MC40芯片的验证积累经验等因素，MC20芯片的验证方案采取与MC40相同得以HAPS平台为FPGA原型的母板，在其上搭建相关的子板来实现具体的验证需求。

# 验证需求

### 时钟需求

MC20芯片SOC端时钟需求

|  |  |
| --- | --- |
| PLL | 频率 |
| CPLL | 1.196G |
| EPLL | 1.5G |
| VPLL | 1.118G |
| DPLL | 1.066G |
| NPLL | 1.6G |
| HPLL | 1.2288G |

以上为SOC端时钟频率，FPGA无法达到如此高的频率，需要一定比率降频。

### 接口/IP需求

MC20芯片软件对于各接口及IP时钟需求：

|  |  |
| --- | --- |
| IP | 描述 |
| Cpu | Cortex A7 双核 |
| Ddr | 前期使用Xlinx ddr3 ip代替，后期换成cadance ip。 |
| Rgmii | 使用xilinx sgmii ip代替，（） |
| Display | 三种输出形态：  RGB  BT1200  DSI |
| Sensor | 4路4lane D-PHY接口  （） |
| Usb2 | M31 phy，（utmi 60m） |
| Spi | 3个普通spi（SPI MST+SLV），加nand flash，nor flash |
| I2c | HS |
| I2s | Multi channel |
| Sdio | Sd casd，emmc，wifi， |
| Emmc |  |
| Jtag |  |
| Flash |  |
|  |  |

### 子卡需求

MC20芯片各接口对应子板

|  |  |
| --- | --- |
| 接口 | 子板 |
| Ddr | LPDDR4子板 |
| Sgmii | SGMII子板 |
| Usb2 | M31子板 |
| Spi | GPIO子板 |
| I2c |
| I2s | I2s子板 |
| Sdio | SD卡子板 |
| Display | MIPI-display子板M31 |
| Sensor | MIPI-sensor子板M31 |
| Emmc | Emmc子板 |

### 子系统需求（小版本）

驱动同事需求尽量多人复用haps，即fpga版本可以在更少的fpga上实现。

#### Base\_sys

base\_sys包括CPU+SRAM+DDR+JTAG+Uart

#### ROM\_sys

ROM\_sys包括CPU+SRAM+DDR+JTAG+Uart+Rom/Ram，需从ROM启动，前期可用RAM。

#### ISP\_sys

base\_sys + PreISP

#### NPU\_sys

base\_sys + NPU

#### Video\_sys

base\_sys + Video Enc + Video Dec

#### DPU\_sys

base\_sys +DPU + TDE（2D加速器）

### 整系统需求（大版本）

# 验证方案

## 时间安排



### 代码移植：（3周）（7.10~7.31）

1、确定时钟方案，重新生成时钟；

2、生成memery；

3、整理filelist，修改代码（DDR相关RTL修改），集成环境。

### 建立仿真环境：（2周）（8.1~8.14）

1、熟悉FPGA仿真环境；

2、搭建最小系统仿真环境；（DV支持，bootrom）（驱动支持，cmm文件）

3、完成串口/GPIO打印仿真。

### 最小系统：（2周）（8.14~8月底）

1、连接串口打印；

2、Xilinx ddr IP gen，仿真；

3、集成ddr IP。

### IP/接口集成：（2~3个月）（11月底）

预计IP及接口每个需要1~2周，Gmac、usb、iis、iic、sdio、flash、CSI、DSI、

### 子系统验证：（1~2个月）（10月底）

目前软件初步给出4个子系统需求，预计每个子系统1~2周

1、base\_sys + PreISP

2、base\_sys + NPU

3、base\_sys + Video Enc + Video Dec

4、base\_sys +DPU + TDE

## 移植方案

### 子卡设计：

|  |  |  |  |
| --- | --- | --- | --- |
| **interface** | **MC20** | | |
| **子卡** | **HT3接口数** | **IO数** |
| Lpddr4 | Synopsis颗粒子卡 cadence soft phy | 3 |  |
| I2S | 自研 | 1 | 8 |
| Gmac | Synopsis MGB子卡 | MGB |  |
| NAND (SPI) | FLASH\_EMMC\_IF\_V10 | 1 | 8  7  12 |
| NOR (SPI) |
| SDIO(card) |
| SDIO(wifi) |
| EMMC |
| I2C | MC40\_PIN\_MUX\_V1.0  两个对面 | 1 |  |
| UART |
| Cpu调试器 | 软件已采购 |  |  |
| USB2 | M31 phy+自研子板(转FMC) | 6 | 51 |
| MIPI(CSI)-RX | M31 phy+自研子板(转FMC) | 6 | 98 |
| MIPI(DSI)-TX | M31 phy+自研子板(转FMC) |  |  |
|  | RGB |  |  |
|  | BT1120 |  |  |
| LVDS |  |  |  |
|  |  |  |  |

### 资源评估：

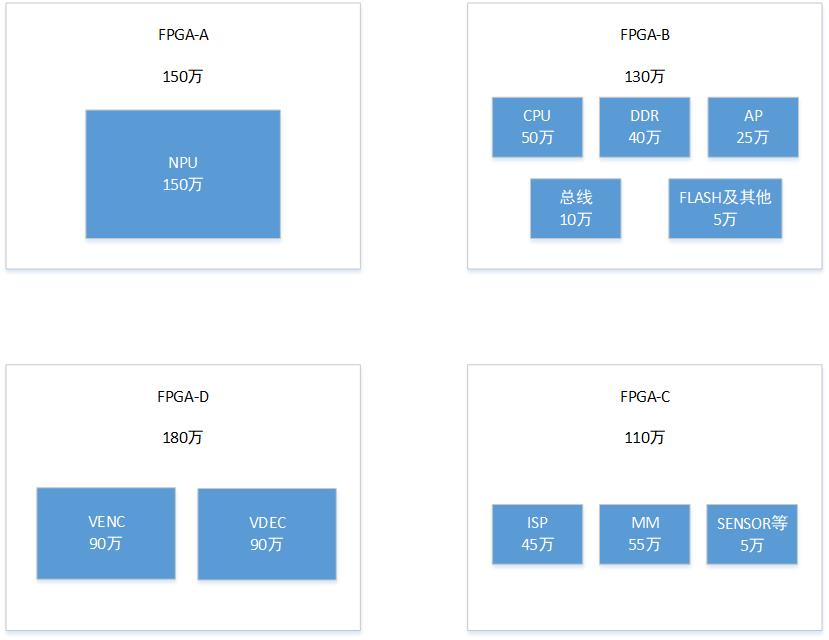
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CELL** | **MC40** | | **MC 20（预估）** | |
| **LUT** | **REG** | **LUT** | **REG** |
| cpu\_sys\_top | 968034 | 265426 | 50万 |  |
| ap\_sys | 726175 | 489599 | 25万 |  |
| ddr0\_sys | 384307 | 227998 | 40万 |  |
| ddr1\_sys | 383813 | 227394 | 0 |  |
| fabric\_FabCen | 175053 | 160811 | 10万 |  |
| npu\_sys | 2849927 | 1966597 | 150万 |  |
| venc\_sys\_top | 1412524 | 513408 | 90万 |  |
| vdec\_sys\_top | 1203984 | 592298 | 90万 |  |
| vdsp\_sys\_top | 1219758 | 399635 | 0 |  |
| isp\_sys | 586828 | 408911 | 45万 |  |
| mm\_sys\_top | 374685 | 260719 | 55万 |  |
| flash\_sys | 27497 | 29278 | 暂时无法预估，但占用资源不多 |  |
| common\_sys | 12316 | 11693 |  |
| sen\_sys | 6397 | 5880 |  |
| pin\_pad | 1733 | 1223 |  |

### 划片方案：

设计原则：将整个MC20芯片的设计映射到2个/4个FPGA芯片中，尽量明确各sys在FPGA上的映射，确定芯片两两之间信号，减少互联信号数量，通过HSTDM设计方法实现互连管脚的时分复用，HSTDM目前每个FPGA间有3个HT3接口，每个HT3接口有48个IO，单个IO频率最高可达到1.2G。设计中最紧张的资源是LUT资源，其他资源均充足，划片按LUT资源划分。

HAPS80-S104共有4颗VU440-FPGA，每颗FPGA有260万LUT，按70%使用量估算每颗FPGA可使用LUT数180万。

根据上节粗略估算，base\_sys(cpu+ddr+ap+fabric+flash)≈130万，可以考虑将base\_sys放在同一颗FPGA上实现，**前提是HT3接口充足**。基于以上，HAPS平台的MC20芯片的FPGA原型验证中设计的划片方案如图所示：



根据初步评估的资源，理论上可以将整个设计放在一台haps80-s104中，但考虑布局布线等问题，后期实现时可能需要做裁剪或其他调整。对于子系统需求，可采用以下方案：

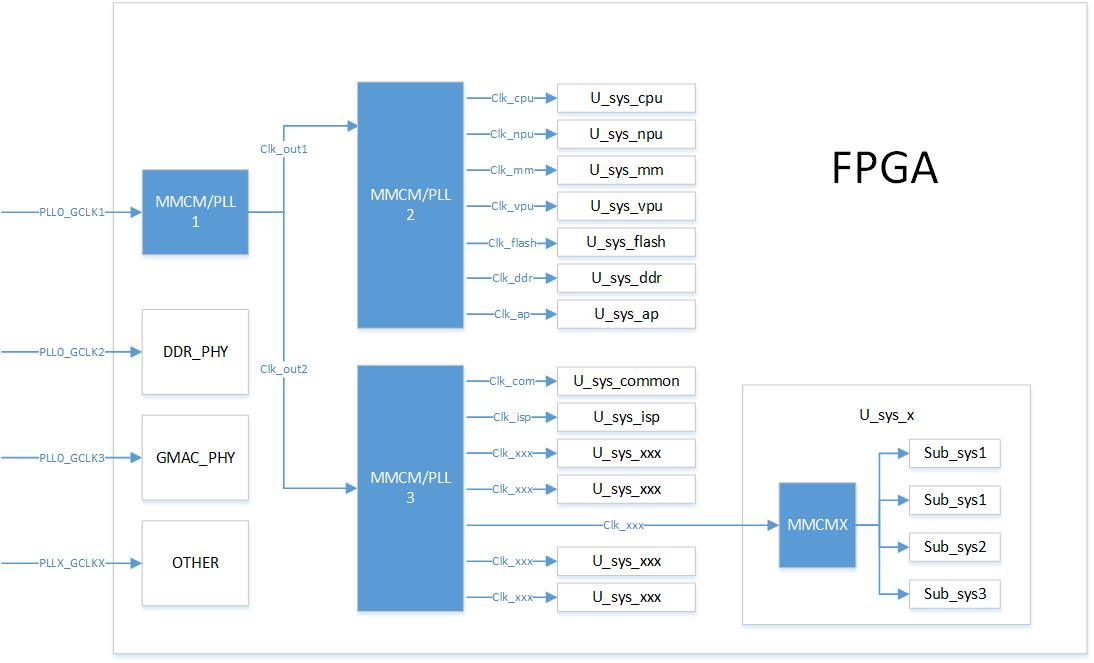
|  |  |  |
| --- | --- | --- |
| Sub sys | 包含sys | FPGA数量 |
| base\_sys | CPU+SRAM+DDR+JTAG+Uart | 单颗 |
| ROM\_sys | CPU+SRAM+DDR+JTAG+Uart+Rom/Ram | 单颗 |
| ISP\_sys | base\_sys + PreISP | 双颗 |
| NPU\_sys | base\_sys + NPU | 双颗 |
| Video\_sys | base\_sys + Video Enc + Video Dec | 双颗 |
| DPU\_sys | base\_sys +DPU + TDE | 双颗 |

* **还有一点需要注意，我们前期需要使用xilinx的ddr IP代替，而Xilinx对于我们使用的vu440芯片没有lpddr4 ip，只有ddr3和ddr4，出于成本考虑使用haps上自带的用于deep trace debug功能的ddr3代替，所以前期无法使用DTD功能.而且需要另采购DDR3子。如有特殊情况需要用到DTD功能时可以插两颗ddr3子板实现。**

### PIN设计：

设计原则：去掉IOMUX，减少可能增加delay和布线拥塞的可能。

### 时钟方案：



设计原则：时钟尽可能简单，同频尽量复用，原则上clk\_gate，clk\_mux全部去掉。

### 代码移植：

#### 分支

通过

## 接口验证方案

### JTAG接口验证

1. 测试目的

* 验证arm JTAG接口的调试功能的正确性

1. 测试步骤

* 通过调试器硬件连接到HAPS的cpu调试接口，在上位机运行debug软件连接cpu

### Spi flash接口验证

1. 测试目的

* 验证Norflash/NANDFLASH接口的正确性。
* 验证arm 从norflash boot的正确性。

1. 测试步骤
2. 通过串口/调试器向nor/nandflash地址中读写数据，判断读写正确性，单次读写ok以后多次/遍历整个地址空间读写。

### UART接口验证方案

1. 测试目的

* 验证UART RS232的功能正确性
* 验证UART RS485的功能正确性
* 验证UART速率

1. 测试步骤

* 通过U转串接口连接，硬件连接上位调试机U口及HAPS UART接口，打开上位机串口助手，配置串口信息，
* 在flash中烧写串口打印kernel，复位后看串口是否正常打印。
* 若flash未调通，则将kernal放在FPGA内部ROM中启动。
* UART子板上应分别有USB转232和USB转485通道，232换回测试，485两颗互联。

### I2C接口验证

1. 测试目的

* 验证I2C master功能的正确性

1. 测试步骤

通过sensor iic验证。

### I2S接口验证

1. 测试目的

* 验证I2S master功能的正确性

1. 测试步骤

外接IIS设备，通过编写cpu测试case访问iis设备。

### DDR接口验证方案

方案一、

1.测试目的

* 验证ASIC模式下DDR的controller+phy的功能的正确性。

方案二、前期使用此方案

1. 测试目的

* 用FPGA的DDR IP替换SOC设计中的DDR，连接DDR3子卡测试DDR功能的读写，为后续运行操作系统提供存储。

1. 测试步骤

* Xilinx ip生成，单ip仿真
* 单ip板上测试，验证ddr3子卡硬件
* 用xilinx的DDR IP替换SOC设计中的DDR
* 搭建DDR的最小调试系统，用ARM对其读写，先进行仿真验证；
* HAPS板上硬件调试，对DDR的读写操作。

### CSI接口验证方案

1. 测试目的

* 测试CSII接口功能的正确性

1. 测试步骤
2. 通过IIC读写寄存器；
3. 子卡采集图像数据；
4. 接收后存到DDR中；
5. 从DDR中取出数据分析；
6. Tx接mc40。

### DSI接口验证方案

1. 测试目的

* 测试接口功能的正确性

1. 测试步骤
2. 将图像数据加载到DDR中；
3. 从DDR取出数据发送；
4. 点亮屏幕。

### EMAC接口验证方案

1. 测试目的

* 验证EMAC的发送数据帧和接收数据帧的情况。
* 验证EMAC的DMA的中断响应情况。

1. 测试步骤

整个EMAC的FPGA验证分别为环回测试、与PC的收发测试：

1. 将EMAC配置自回环模式，然后发送固定的帧。测试是否收到数据。
2. 例化两个EMAC接口，分别对应EMAC子卡两个网口，子卡网口用网线互联，然后发送固定的帧。测试是否收到数据。
3. 通过千兆网PHY子卡将EMAC与PC端相连。连接编写ARM程序，控制EMAC初始化，然后发送固定的帧。只接在调试串口上ping一个任意网址。

### USB接口验证方案

1. 测试目的

* 主要测试集成的USB controller部分是否配置正确，能否和PHY配合，正确完成USB通信功能；

1. 测试步骤
2. 硬件部分使用FPGA通过HT3接口和USB PHY芯片互联；并通过USB接口和上位机互联；
3. 软件部分，如果软件能够正确枚举出串口，并且能够正确通信，那么即表明USB controller工作正常；

### SDIO接口验证方案

1. 测试目的

* 测试SDIO master控制器的正确性；
* 测试SDIO slave的正确性；
* 测试能否正确读写访问SD卡；
* SDIO-WIFI功能通过SD卡来验证
* 测试能否正确读写访问emmc卡。

1. 测试步骤
2. 硬件部分使用FPGA通过HT3接口外接SDIO子卡；
3. 软件部分通过对SD的扇区进行读写访问；
4. 测试结果，使用串口打印抓取SD卡返回的响应验证SD-HOST与SD卡通信的正确性。
5. 环回测试验证slave的正确性或连接mc40验证。

### GPIO接口验证方案

1. 测试目的
2. 测试usb gpio正确性；
3. 测试gmac gpio的正确性；

验证方案：

Output通过子板引出，连接示波器测试；

input，与ouput连接回环测试。

## 子系统FPGA原型方案

### Base\_sys最小系统验证方案

#### 验证方案

base\_sys包括CPU+SRAM+DDR+JTAG+Uart。

由于fpga版本ddr到货时间未定，前期采用xilinx ip代替。Xilixn ip不支持lpddr4，只支持ddr4、ddr3，每台haps上自带一个ddr3子板，所以选用xilinx ddr3 mig ip，外接axi 4转1桥及异步桥。



#### 验证步骤

1. 生成Xilinx DDR3 MIG ip，单ip仿真；
2. 单IP集成JTAG debug IP，综合实现生成bit，板上通过测试，验证ddr3子卡硬件

3、用xilinx的DDR IP替换SOC设计中的DDR

4、搭建DDR的最小调试系统，用ARM对其读写，先进行仿真验证；

5、HAPS板上硬件调试，对DDR的读写操作。

### ISP\_sys验证方案

base\_sys + PreISP

### NPU\_sys验证方案

base\_sys + NPU

### Video\_sys验证方案

base\_sys + Video Enc + Video Dec

### DPU\_sys验证方案

base\_sys +DPU + TDE

## 整系统级验证方案