* 1. 2018-1-8 T4080+PCIe调试和QT环境搭建
     1. T4080+PCIe调试

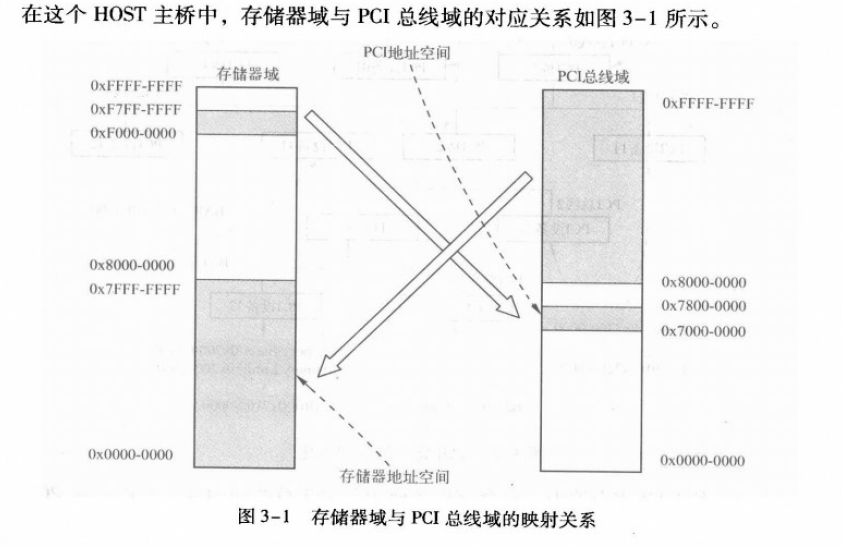
PCIe的驱动是OK的，经过反复查看和确认，两个PPC的PCI总线地址映射并不是在底层驱动里面完成，而是在其他地方配置。（这里花了很长时间看代码和资料，主要是对PCIe不是很熟悉）

目前，FPGA烧写有PCIe的代码后，PPC0（主）可以扫到FPGA，并能访问到其配置空间，后续工作就是等硬件和CPLD把复位调试OK，复位后通过PPC0对FPGA的PCIe Bar空间进行地址分配。

总结一下：PCIe总线的比较重要的思想就是地址映射，具体映射方式，如下：

**存储器域A-----PCIe控制器（转换）-----PCI总线地址-----对端的PCIe控制器（转换）------存储器域B**

如图所示：



* + 1. QT环境搭建

Qt环境搭建的实质是编译出qmake以及相关的QT库。（但是，花了一晚上才搭建好，艹）

* 1. 2018-1-9 Linux+QT

熟悉了QT的环境配置，以及基本的使用。

* + 1. qDebug无输出

影响qDebug输出的主要是一个宏定义：QT\_NO\_DEBUG\_OUTPUT，该宏定义可通过一下方法设置：

方法1：直接在main函数之前加入宏定义。注意，这种方法，好像无法保证全局所有类中的qDebug被屏蔽。最好还是用第二种或第三种方法。

方法2：

在Qt工程文件.pro里面添加

DEFINES += QT\_NO\_WARNING\_OUTPUT\ QT\_NO\_DEBUG\_OUTPUT

然后clear项目重新编译即可屏蔽qDebug()的输出.

而影响QT应用程序qDebug输出的，除了编译选项外，QT应用程序本身对输出的重定向也会导致qDebug无输出，比如：

./Test1 >/dev/null 2>/dev/null

>/dev/null 将Test1的输出重定向到null设备，将2（标准错误）也重定向到null，换言之，所有Test1中的qDebug都无法输出到终端，包括标准错误信息，因为它们都被重定向到null设备，而null设备是个空设备，任何写入它的数据都会被丢弃。

* + 1. QT静态插件（实现ARM上的QT自定义键盘）

<https://wenku.baidu.com/view/79a5aa645a8102d277a22f1f.html>

熟悉了一下QT的编程，也熟悉了一下QT的插件编写，对要使用什么类不是很熟悉。（痛苦ing）