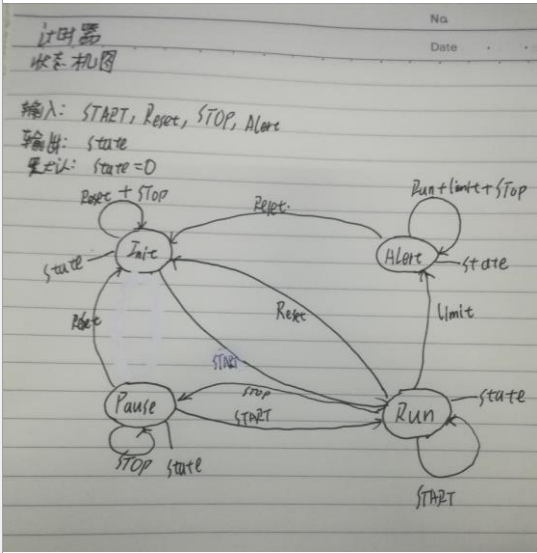


数字逻辑实验三实验报告

实验名称	短跑计时器设计与实现			课程名	数字逻辑
姓 名		学 号		班 级	
实验地点		实验日期	2018. 5. 20	组长电话	
指导老师	黄永刚	同组其他成员		Email	.

一、实验内容（含实验原理介绍）：

一共分为四个模块，第一个模块是分频器，即将系统的频率变为 100Hz，以便于后面的计时。第二个模块是实现对状态的控制，即输入 start、stop、reset 后要转换为对应的状态。第三个模块是实现计数的功能，即每 10 毫秒在末尾加 1，后面 3 位都是满 9 进 1，倒数第四位是满 6 进 1，第四个模块是实现数码管的显示。最后用一个主模块将这四个模块综合起来。



控制器模块的状态机图

状态	状态码	转换条件	下一个状态	下一状态码	非0输出
Init	1000	reset+stop	Init	1000	
		start	Run	0001	state
Pause	0100	reset	Init	1000	
		stop	Pause	0100	
		start	Run	0001	state
Alert	0010	reset	Init	1000	
		stop+run+limit	Alert	0010	
Run	0001	reset	Init	1000	
		stop	Pause	0100	
		limit	Alert	0010	
		start	Run	0001	state

控制器的状态表

二、实验目的：

实现一个最大计时时限为 1 分 59 秒 99 的计数器，并实现 start、end、reset 功能，当时间大于一分钟时，代表分钟的 LED0 亮，当时间达到 2 分钟时，计时停止，LED7 亮，表示超过最大计时时限。

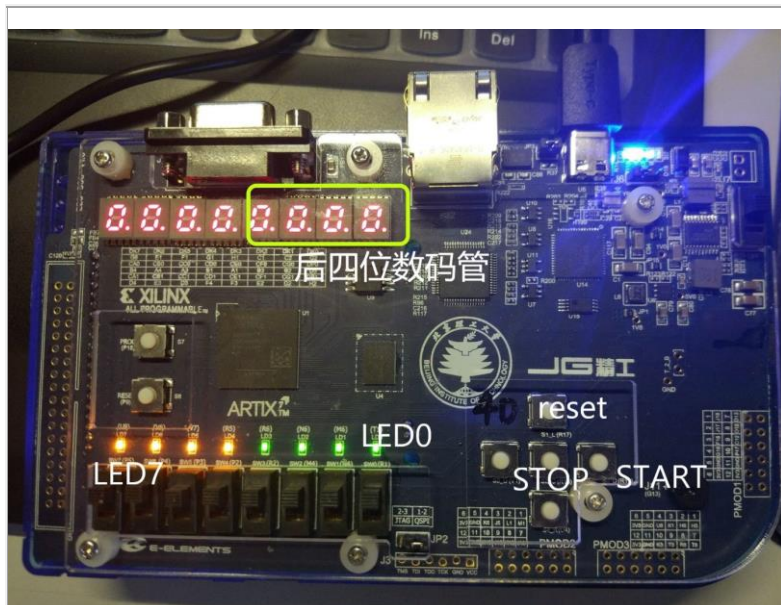
三、涉及实验的相关情况介绍（包含使用软件或实验设备、设计思路等情况）：

我们的设计思路是自底向上的，先实现了分频、状态控制、计时、数码管显示四个模块的功能，然后通过主模块将其综合在一起从而实现整体的计时器的功能。整个实验过程中使用 vivado 软件和 EES-338 口袋计算机进行开发。在实验中主要使用了 EES-338 口袋计算机上的后四位数码管用来显示秒和毫秒位，用了 LED0 表示是否达到一分钟，当时间达到一分钟，LED0 亮起。用 LED7 作为警示灯，当时间达到两分钟及以上时，警示灯亮起。然后用了三个按钮分别用来表示 start、stop、reset。

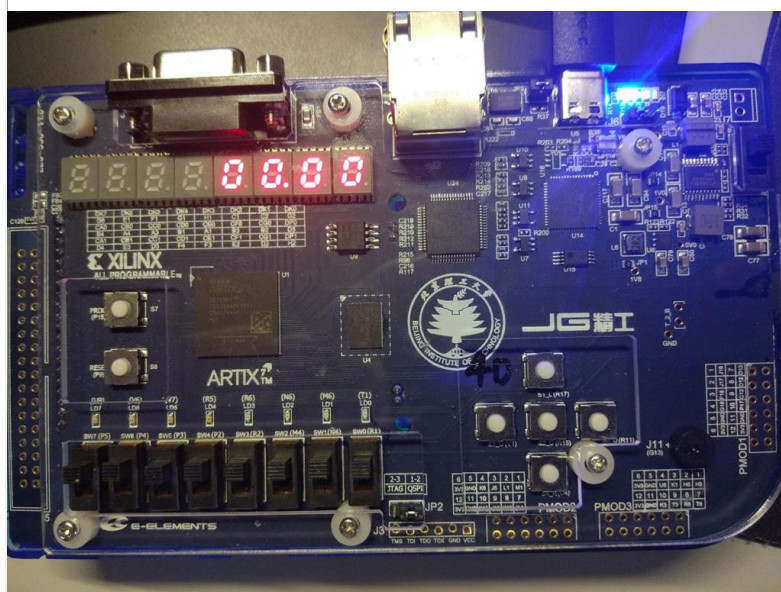
四、实验步骤和测试方法

我们首先确定了模块的划分，即一个主模块综合四个分模块的设计思路，然后设计了控制器模块的状态机图和状态表，通过控制器模块的状态机图写出来控制器模块的代码。然后通过分析各个模块的功能完成了各个模块的代码，在 vivado 上生成文件之后就在 EES-338 口袋计算机上进行检测。

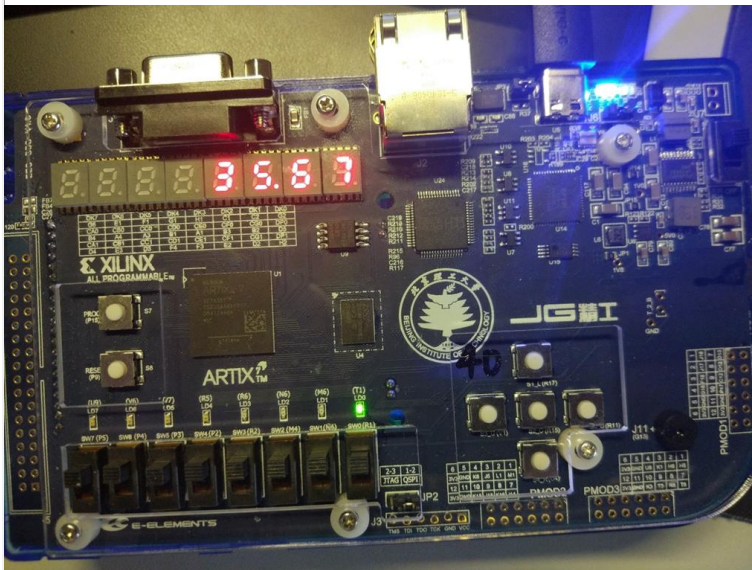
五、实验结果（含程序、数据记录及分析和实验总结等，可附页）：



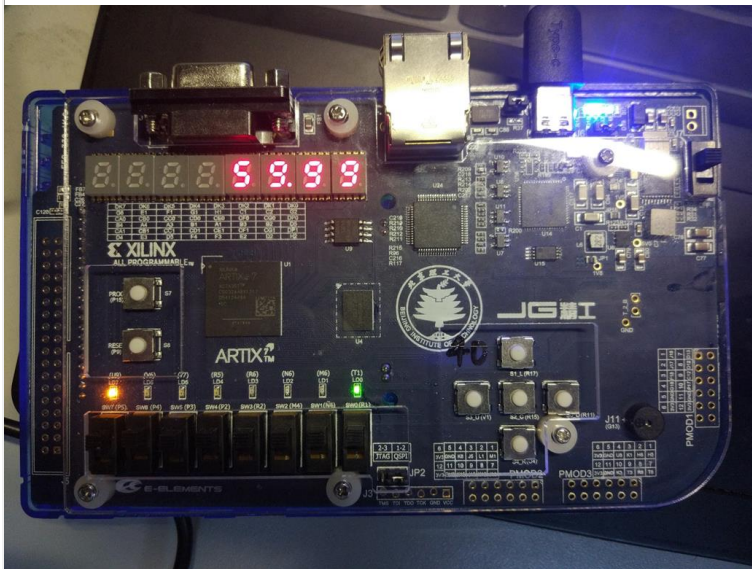
配置的管脚



最初状态以及复位后的状态



计时超过一分钟时 LED0 亮起



计时超过两分钟时 LED7 亮起

六、实验分工、实验过程中问题的处理、讨论和建议，收获和体会

在这个实验过程中，我们遇到的主要问题一共有两个，第一个是开始频率调的过低，导致数码管在显示时总是闪烁，后来我们的解决办法是将频率调高为原来的 50000 分之一，并在显示时隐藏最后一位毫秒位，也就是显示的最后一位还是 10 毫秒位，这样由于人眼无法分辨过高的频率，所以数码管的显示看上去就不会是闪烁的。第二个问题是刚开始没有设置报警灯，而只是计时达到两分钟及以上时，计时停止在 1 分 59

秒 99 的状态。这样当时间位 1 分 59 秒 99 时，我们就无法确定是超时了还是刚好在 1 分 59 秒 99 停止。后来增加了 LED7 作为报警灯使得超时的标志更加明显。在整个实验完成之后，我们主要有两点心得，第一是团队合作很重要，一个工作一个人可能要两天才能完成，但两个或者几个人一起较大的提高工作效率，每个人的知识都有所区别，合作能互补，故而能提升效率。第二是规划很重要，特别在一个项目比较复杂的时候，首先要构思出这个项目的总体结构，例如我们在做这个实验时，如果首先便将报警器计入项目中，可能效率会更高一点，而不至于后面来加。