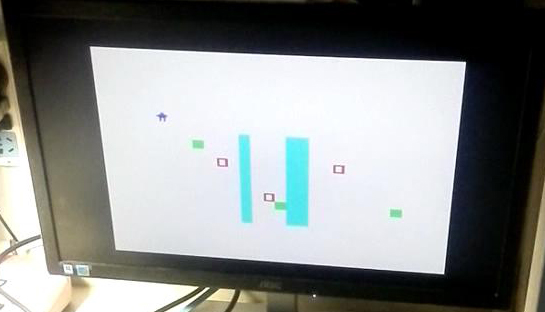
**基于nexys2板设计的“推箱子”**

**游戏介绍**

**主要功能：**

1. 设置一个小人，可以利用nexys2板提供的4个按钮实现其上下左右移动。
2. 设置3个箱子，可以由小人推动箱子自由移动。
3. 设置2面简单的障碍墙，小人或箱子一但碰到墙则不能再向墙方向移动。
4. 当有箱子被推到指定的3个目标中任意一个时，数码管及led灯分别显示目前已经到达目的地的箱子数目。
5. 当所有箱子均被推到指定点时则获得胜利，显示“you win”。

效果图：



**原理及代码实现：**

在800\*600的屏幕上选出一个600\*500的区域，选中的区域为白色，其他的颜色为黑色，箱子可以在白色的可移动区域中移动。不能在黑色区域中移动。输入的时钟信号即采用了nexys2自带的50M Hz信号。

主要的原理及代码实现可以分为以下几个部分

1. 边框模块：变量biankuang控制VGA显示
2. 按钮模块：BTN\_EAST BTN\_WEST BTN\_SOUTH BTN\_NORTH分别为按钮4个输入位，对应的move\_left move\_right move\_up move\_down表示让小人向4个方向移动标志，1位移动，0不移动。为去除按键的抖动及控制每秒的移动次数，设置s\_counter3\_flg，每0.2s产生一个标志，只有在其产生标志时，移动输入才有效。
3. 箱子模块：设置三个20\*20像素的箱子，箱子设为绿色，一开始箱子在初始位置不动，当标志与箱子接触时，标志可以推动箱子向前移动，当箱子碰到墙或者边框时，则箱子不能再向前移动，而且此时标志也不可以在相同方向进行移动。当箱子到达目的点的时候，则箱子不能再移动，当三个箱子都到达目的点的时候，则游戏结束，出现结束画面。三个箱子可以任意到达三个目标点。标志推动箱子移动的过程的实现分为四个方向，当某方向的标志和箱子相接触，并且标志继续向着此方向移动，则箱子也朝着该方向移动，箱子每次向前移动20个像素，但当箱子在某方向上碰到障碍物或者边框的时候，则箱子不能够再在此方向上移动。
4. 小人模块：设置一个20\*20像素的标志，标志设为蓝色，标志可以随着按键的不同上下左右自由移动，每次向前移动20个像素，当标志接触到障碍物或者边框时，则标志不能移动。当标志接触到箱子且箱子和标志移动的方向上没有障碍物时，则标志和箱子一起向该方向移动。当标志和箱子移动的方向上碰到障碍物或边框时，则箱子和标志则不能在这个方向上移动。标志的移动主要是通过标志的相关坐标在该方向像素的加减实现的，由于每次标志移动20个像素，所以每次相应坐标加减的像素大小也为20个像素。用01数组将标志描画成一个小人，再命令数组为1的部分显示为蓝色，为0的部分为白色，然后就可以在原来20\*20像素的大小的方块上显示出一个小人的图案。
5. 目标点模块：设置三个20\*20像素的目标点，目标点为红色框，目标点为不动点。当绿色的箱子被蓝色的小人推到红色的目标点时，则箱子即被固定住，不能再被推动。当三个箱子都被推到目的点时，游戏结束，出现结束画面。三个箱子可以任意到达三个目标点的其中之一，已有箱子的目标点不能再推入箱子。
6. 障碍物模块：为模拟真实的推箱子游戏情况，在可移动区域内增加两堵矩形的障碍物墙，该障碍物的作用是可以阻挡小人和箱子在此方向的前进。当小人和箱子碰到障碍物时，则不能再移动。由于标志的移动是通过相应方向上的像素的加减实现的，所以此时可以设计一个判定，当标志在某方向碰到障碍物时，则相应的像素不再进行加减。当标志是推动箱子向前移动时，当箱子在某方向上碰到障碍物时，则箱子和标志的相关像素不再进行相应的加减。另外箱子也不能进入障碍物所划定的区域。
7. Led灯及数码管显示模块：从目标点标志处读取目前已经到达目标点的箱子个数，led灯以二进制形式显示目前已完成的箱子数，7段数码管则以数字形式直接显示已到达目标的箱子个数。
8. 结束画面：当三个箱子都被推到目的点时，就会出现一个结束画面，结束画面为在黑色的屏幕上出现一个绿色的“you win”。“you win”的画法与之前画小人和目标点的方法是一样的，只是先设计一个标志，开始时此标志置1，显示游戏画面，当读取到游戏结束的信号时（三个箱子都被推到目标点时），此标志置0，此时显示的是结束画面。
9. VGA显示信号接口：取用8bit颜色输出的其中3bit，即为简单的red blue green来完成输出要求。通过VGA显示对前面所写的小人，箱子，障碍墙，目标点等进行着色，其中小人为蓝色，箱子为绿色，障碍墙为淡蓝色，目标点为红色，结束画面字母为绿色，游戏区域为白色等。

**操作说明：**

利用4个按钮可以实现小人的上下左右移动及推动箱子的功能，拨动开关switch0可以实现游戏的复位（重新开始）。

**设计总结及收获感悟：**

这是自己第一次单独使用Verilog语言设计比较大的程序，所以所花的时间都比较多。最开始是在按键的部分遇到了比较大的麻烦，主要是按键不太稳定导致按一下键会有箱子跑出屏幕的情况，进行按键去抖动后还是不能解决问题。通过分析程序，发现该问题是由于时钟触发信号频率过高，导致在一次按键的过程中形成多次触发，最终使箱子跑出屏幕。

分析出上述原因后，为了防止在一次按键过程中形成多次触发，于是设计了一个时钟计时器，使其每隔0.2s产生一个标志，用它和按键一起控制小人的移动，使小人只有在按键按下时和标志为1这两个条件同时满足的条件下才能移动，这样就比较好的解决了在一次按键的情况下多次触发的问题。

其后就是解决小人和箱子碰到边框以及障碍物后不能再移动的问题，一开始我是设计一个标志位，当箱子和小人碰到墙后则让标志位置1，从而使相应方向的坐标像素不再加减，但这样做无法使标志再置0，从而导致小人永久不能在此方向移动，于是我就决定跳过标志，直接通过小人是否与相对应的墙壁碰撞来决定相应坐标进行加减，这样虽然逻辑判断方面比较困难，代码量也大大增加，但还是较好地解决了问题。

这里的逻辑判断主要分为三个方面，一个是小人碰到相对应的障碍物则不能再继续移动，第二是小人推动箱子碰到相对应的障碍物时，小人不能再移动，第三个是箱子不能移动到障碍物相对应的区域。

通过这次的设计实验，我加深了对VGA显示的相关过程的理解，熟悉了Verilog语言的相关编程的运用，也提高了自己在分析程序和排除错误方面的能力。