UART.v

ScriptMem.v

DemoTop.v

这三个都是project本身提供的

UART.v就是UART负责UART传输协议的

DemoTop.v就是顶层模块

ScriptMem.v就是负责脚本的读取和加载

DivideClock.v这个模块用于时钟分频

我把100MHZ的时钟转换成了UART所需要时钟,1000HZ时钟和1HZ时钟

(后面两个暂时没用上)

TravelerOperateMachine.v

TravlerTargetMachine.v

GameStateChange.v

这三个模块的功能都是一样的,就是根据FPGA的按钮开关输入产生输出的8bit数据

生成的3个数据都在顶层DemoTop.v模块中用三个8bit寄存器变量保存

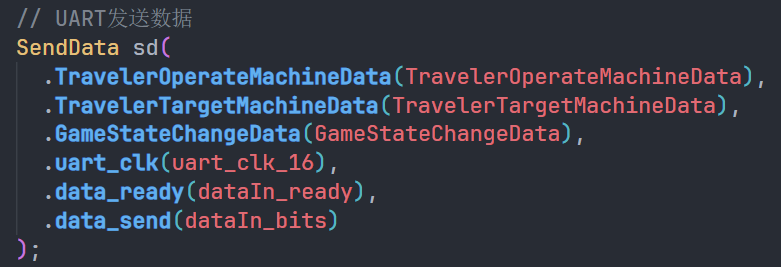
***wire [7:0] GameStateChangeData;        // 游戏状态***

***wire [7:0] TravelerOperateMachineData; // 机器操作***

***wire [7:0] TravelerTargetMachineData;  // 机器选择***

SendData.v这个模块是用来向客户端发送数据的模块

他把上面的三个寄存器变量以循环赋值给UART发送的数据包里



dataIn\_bits是要通过UART发送的数据

dataIn\_ready(应该是不需要了)

把上面三个变量循环赋值给dataIn\_bits(每个uart时钟周期切换一次)

在TravelerOperateMachine.v ,TravlerTargetMachine.v这两个模块中还做了一个防抖处理

就是保持按下按钮1ms左右才认为你是按下了按钮,不然就当作轻微扰动直接忽略



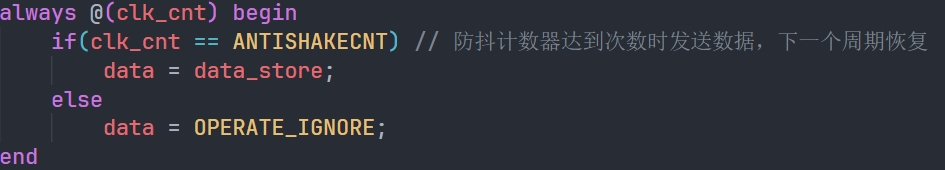
clk\_cnt是一个计数器,在每个时钟周期自增,如果达到设定的值(也就是到达了1ms)认为你成功按下了按钮.

prev\_buttons 保存上一次按下的按钮

buttons 保存现在按下的按钮

如果在一个时钟周期内不一样,就将计数器清零,prev\_buttons = buttons

如果一样就将计数器自增



对于按钮而言,如果计数器到达了设定好的值,就改变按钮输出的数据,在下一个时钟周期还原