**设计名称：电子超速抓拍器**

**组号：7号**

**学生1姓名：余尔聪**

**实现内容：摄像头（管脚约束、VGA显示），边沿检测的调试与综合，测速状态机的编写与调试**

**学生2姓名：江陆翔**

**实现内容：sobel算子边沿检测，测速状态机的初始代码以及debug，报告撰写**

**学生3姓名：赵俊童**

**实现内容：SD卡的存入与读取，SD卡提取图像的VGA显示，SD卡与摄像头的代码合并**

**一、设计的功能和指标**

可以对监控路段的车辆进行实时的监控，并且自动抓拍超速车辆，照片存入后台数据库，可以随时调看。摄像头采集的图像实时显示于屏幕，同时显示实时的灰度图与图像的边沿检测。

**二、设计基本原理**

1. 摄像头：

实现摄像头的管脚约束，将采集到的图像通过VGA加以显示。显示分为四块，分别为原图显示、二值化显示，预留SD卡图像读取显示、阈值可调整的边沿检测。

2. 图像边沿检测：

采取sobel算子进行图像的边沿检测，生成边沿检测图像。

3. SD卡的读取与写入：

通过SD卡的存储，弥补实验板内部bram空间不足的问题。

4. 测速模块：

实现测速器的功能以及抓拍功能。

**三、设计步骤**

1. 工程的模块化分工，将总工程分解成摄像头、边沿检测和SD卡三个模块

2. 分块完成，分别实现单独的摄像头采集、VGA显示、边沿检测、SD卡的读取与存储等功能

3. VGA显示的分区实现，将摄像头采集的原始图像、二值化处理后的图像（灰度图）以及边沿检测的图像同时展现于显示屏

4. 编写状态机模块，实现对于超速车辆的检测及抓拍等功能

5. 将SD卡编入总工程，实现对于图像信息的存储以及对历史图像的调看等功能

6. 优化处理

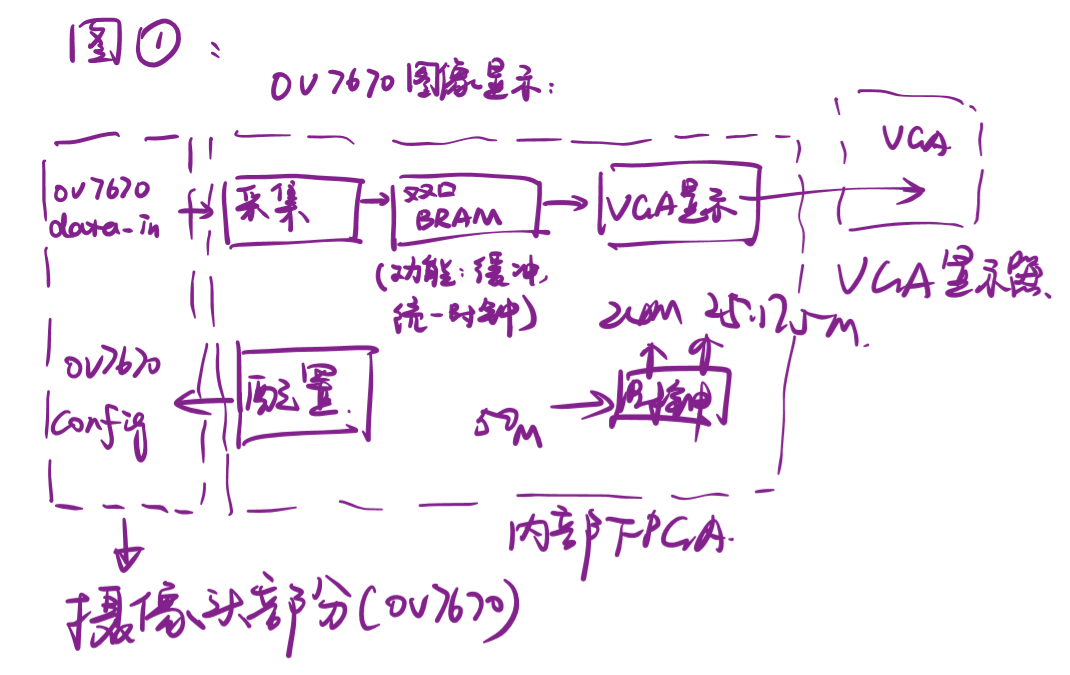
**四、详细方案**

**1. 摄像头**

摄像头是我们项目的重要也是唯一输入设备，我们的设计采用了教学统一使用的OV7670摄像头，将采集到的视频通过FPGA板子上的12位VGA视频接口输出。对于OV7670摄像头芯片的特点总结：OV7670芯片可输出色彩深度为16位（RGB565）的640\*480的图像，每次传输的视频由8位并行输出，故而将两个连续的8位数据合成一个16位的像素点。图像的刷新速度为30FPS，从而得到24M左右的像素点时钟。HREF，VSYN和PCLK三根信号线控制数据从OV7670摄像头到FPGA实验板的写入。

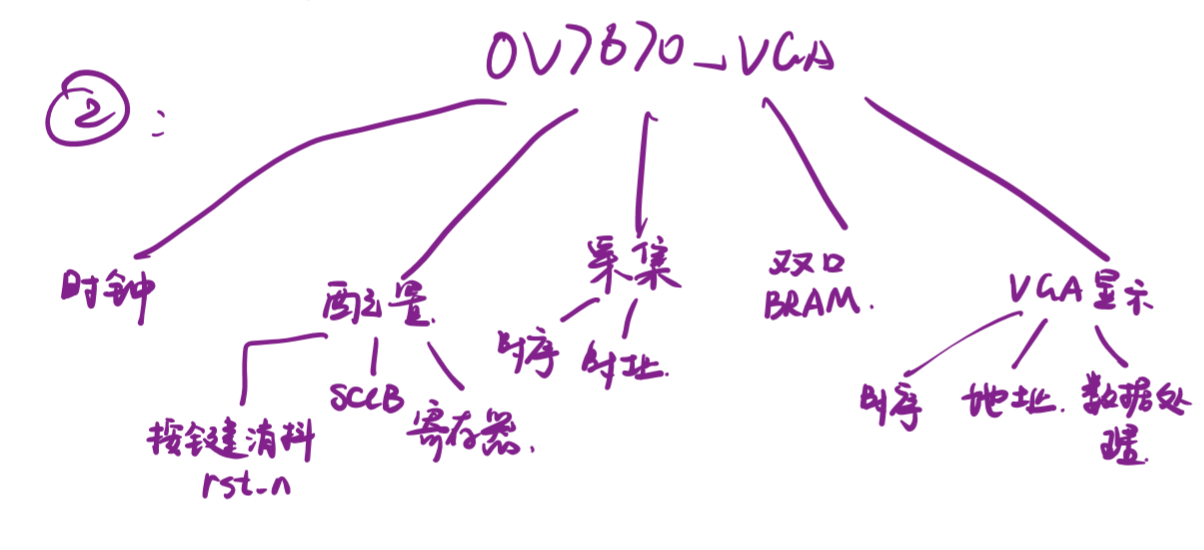
为了实现VGA显示，我们先要将数据保存到一个双口RAM，摄像头作为写端口输入数据，VGA作为读端口读取数据，两者各自管理自己的地址（读地址/写地址），相互独立。故而摄像头的设计以双口RAM为界，可以分为两个模块：由OV7670采集数据的写入端和VGA控制器产生地址并读取显示的读出端。

首先是由摄像头OV7670采集数据的写入端，详细划分见下图。



由上文的芯片特点可知，OV7670的时钟周期为24M，而FPGA板子晶振的时钟周期是50M，所以需要一个时钟转换模块。（PS：在实际设计中我们发现很多模块都需要不同的时钟周期，比如VGA显示的时钟周期为25.175M，因此时钟转换模块在后面都需要，不再明确指出）此模块的具体流程如下图所示。对于输入数据，由OV7670进行采集，接着传输到双口BRAM 用以缓冲从而统一时钟，再传递到VGA的显示端。（更改摄像头配置使得图像上下、左右镜像，以符合常识认知。）

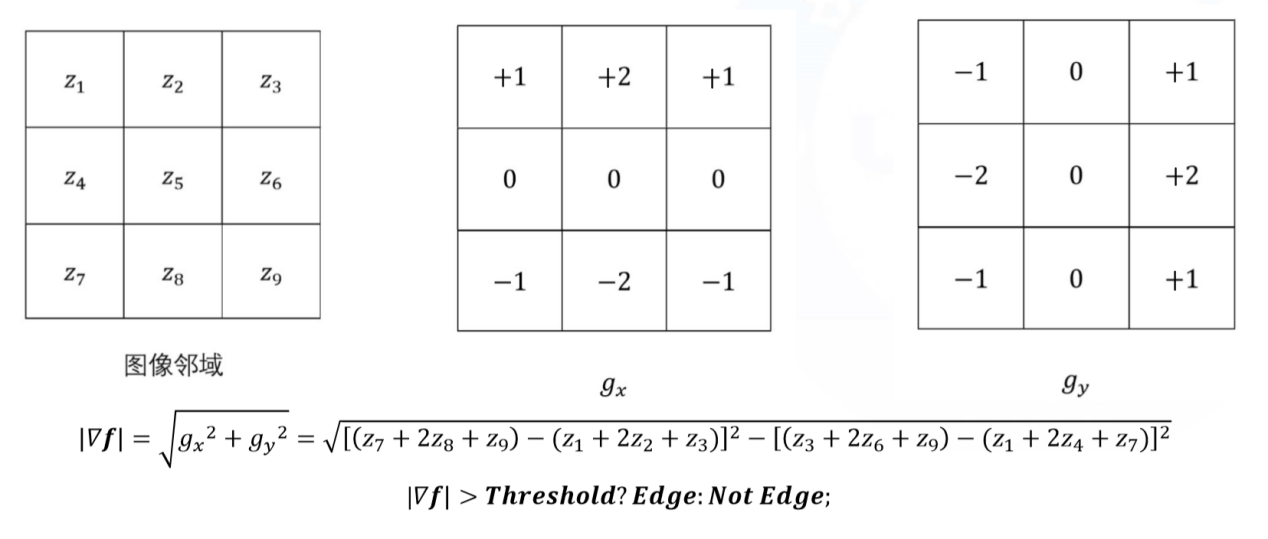
另一个模块是VGA显示模块，即输出端。OV7670\_VGA模块是我们主要实现的模块，可以细分成时钟、配置、采集、双口BRAM和VGA显示五个子模块，对于这些子模块又可作细分，从而从底层任务完成代码的编写。详见下图。



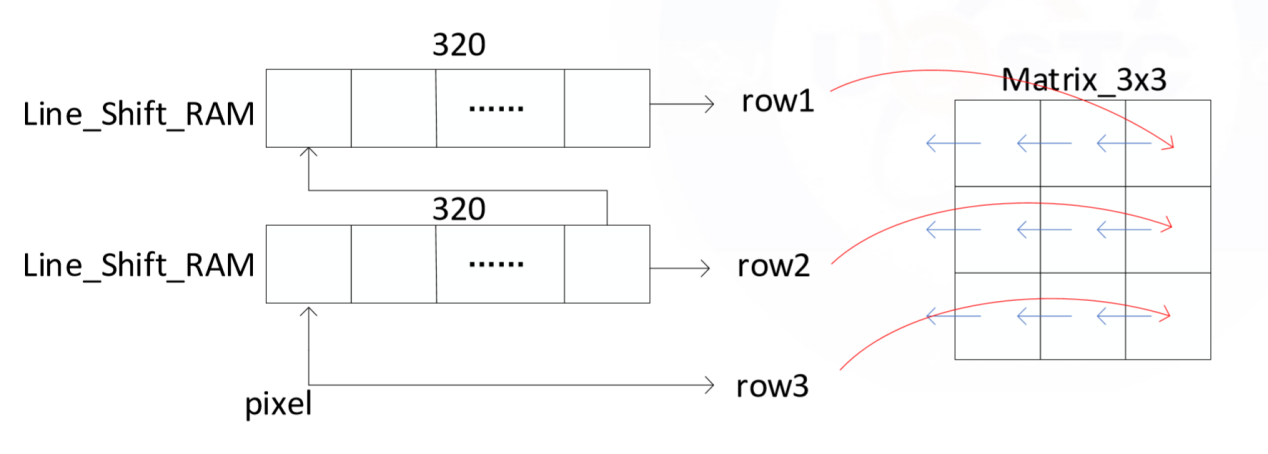
**2. 图像边沿检测**

图像边沿检测是我们工程任务实现的核心，是对于车辆位置检测的前提条件。对于本此设计中的图像边沿检测，我们采用了sobel算子实现，下面将展示sobel算子的算法和我们为了融入总工程而加以修改的部分。

首先是sobel算子边沿检测的原理。一阶导数数字上可以近似为差分，sobel边沿检测使用一个3×3的领域的行和列之间的离散差来计算梯度（即水平和数值两个方向上的梯度），其中，每行或者每列的中心像素用2来加权，以提供平滑效果。将水平和竖直方向上加权后的数值再求几何平均值得到最终的结果，将此结果与提前设好的阈值比较大小，当其超过阈值时便将此像素点视为边沿点。



接着是sobel边沿检测算法的代码实现。如下图所示，我们创建了两个深度为8的320位的移位寄存器（IP核）和一个3×3的邻域矩阵（移位方向如图箭头所示）。因为边沿检测的图像显示是1/4分屏显示，即320×240的屏幕显示，所以两个移位寄存器各自存取了一行像素的数据。对于读入图像的第一行和第二行的扫描，图像的数据信息逐次存入row1与row2，此时并未进行有效的边沿检测运算。从图像的第三行开始（第三列开始是有效计算），两个移位寄存器（row1&row2）和实时输入的数据（row3）构成一列的三个像素点，与前面六个像素点共同于邻域矩阵中，再根据sobel算子的原理判断是否为边沿。

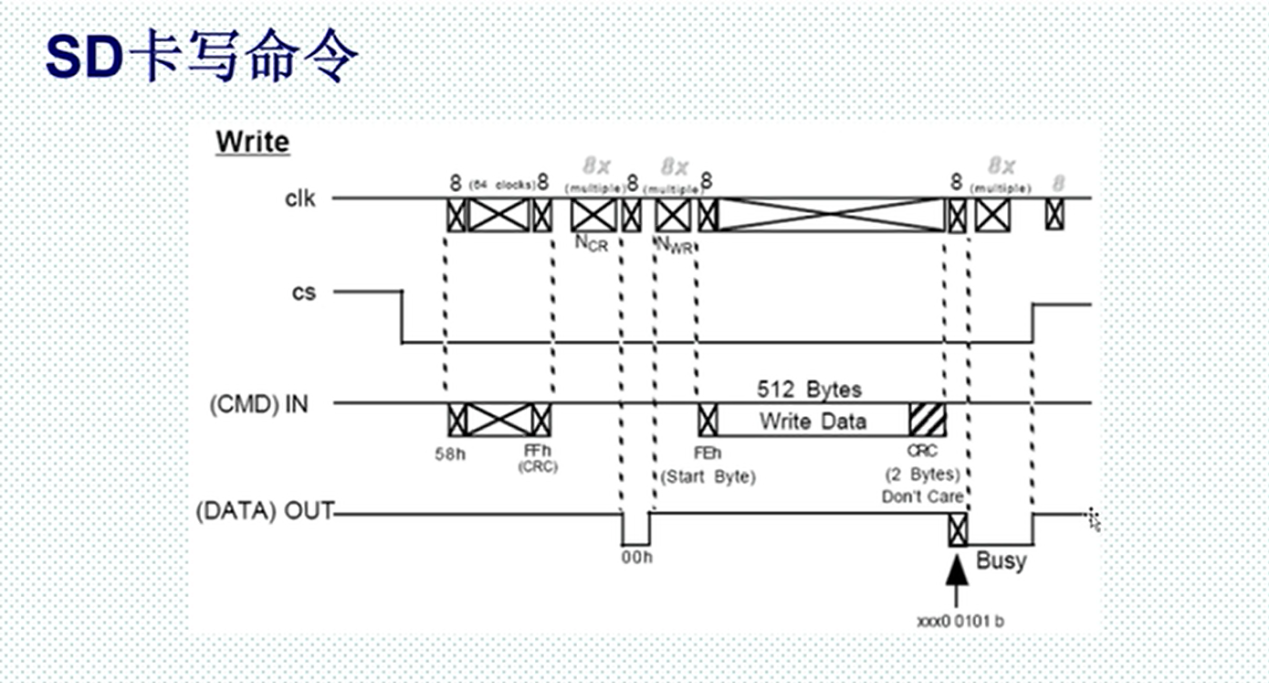


在代码的编写中，我们碰到了诸多未曾想到的问题。例如图像显示重影，后经深入学习讨论我们发现程序调用移位寄存器进行工作时（数据写入/数据移位）并不是在一个时钟周期内完成，有些操作在一个周期内只能进行一次，所以sobel内的工作不能在一个时钟周期内完成，需要相应的时钟延迟。通过调试实际程序，我们将延迟定为5个时钟周期时效果最好，故而我们用了5个16位的寄存器变量以实现时钟延迟功能。对于sobel边沿检测中的×2运算，我们运用了左移一位的操作；对于负数部分的运算，我们先在正数状态下进行运算，后与正数部分相比较，用大的数字减小的数字从而达到负数部分的实际效果（平方后符号不重要）。最后的几何平均值的运算，我们亦采用了专门的IP核。

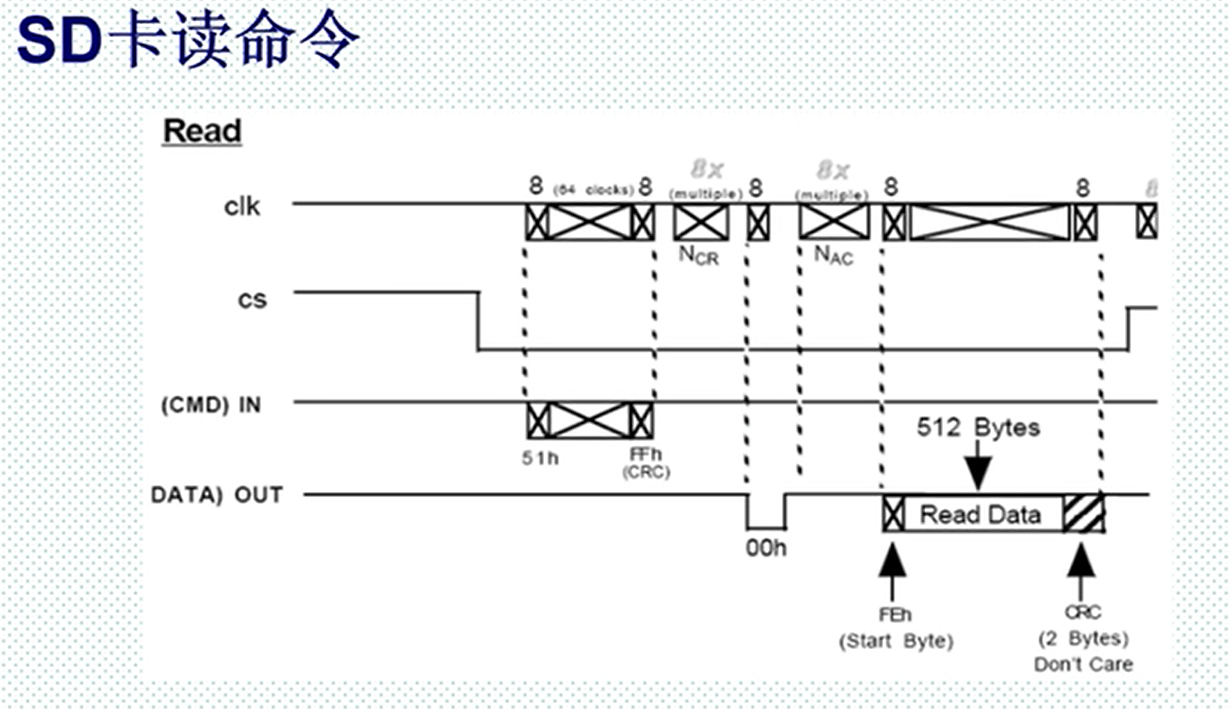
**3. SD卡的存储与读取**

由于工程设计需要，实验板自带的bram难以存储较多的图像信息，故而我们选择在工程中加入SD存储卡部分。对于SD卡的操作，主要分为数据的写入和数据的读取两个部分。

首先是数据的写入，即SD卡写命令，时序逻辑图如下。首先是对于CS变量的拉低，作为开始的信号。接着发送SD卡的写命令使能，即IN信号所示的58h，后面跟着的是SD卡写扇区的地址，校验位FFh作为地址的结束标志。在写使能命令发送完后，因为SD卡此时并不一定可以实现写入数据的操作（例如当前时钟下的SD卡正在进行读取操作等），所以需要等接收到SD卡的正确反馈后（00h）才能对SD卡进行相应的操作。在接收到SD卡的反馈后，等待8个时钟，就可以发送有效的数据了。首先需要发送数据的数据头（FEh，可视作起始的字节），接着发送512个字节的有效数据，最后是两个字节的校验位CRC。完成数据的写入后，SD卡会拉低miso引脚，这个引脚一直处于低电平，表示SD卡一直处于忙状态，不可以对SD卡进行其他的操作。等待SD卡操作完成后，拉高miso引脚，然后可以拉高CS信号，表示一个写操作的完成。等待8个时钟周期后，就可以对SD卡进行新的读写操作了。



对于SD卡的数据读取模块，时序逻辑图如下。类似于写命令的操作，首先是CS的拉低作为读命令的开始信号。接着读入SD卡的读取地址（51h开始，校验位FFh结束），然后等待SD卡返回有效数据作为读使能信号的反馈，表示SD卡已经准备好读操作。00h的返回说明SD卡响应了读命令，接着等待接收SD卡的数据（起始字节FEh，512个字节的有效数据，2个字节的校验位）。同样的，完成有效数据的传输后等待8个时钟周期，就可以将CS信号拉高，结束当前的读操作，再次等待8个时钟周期后，就可以进行下一次的读写操作了。

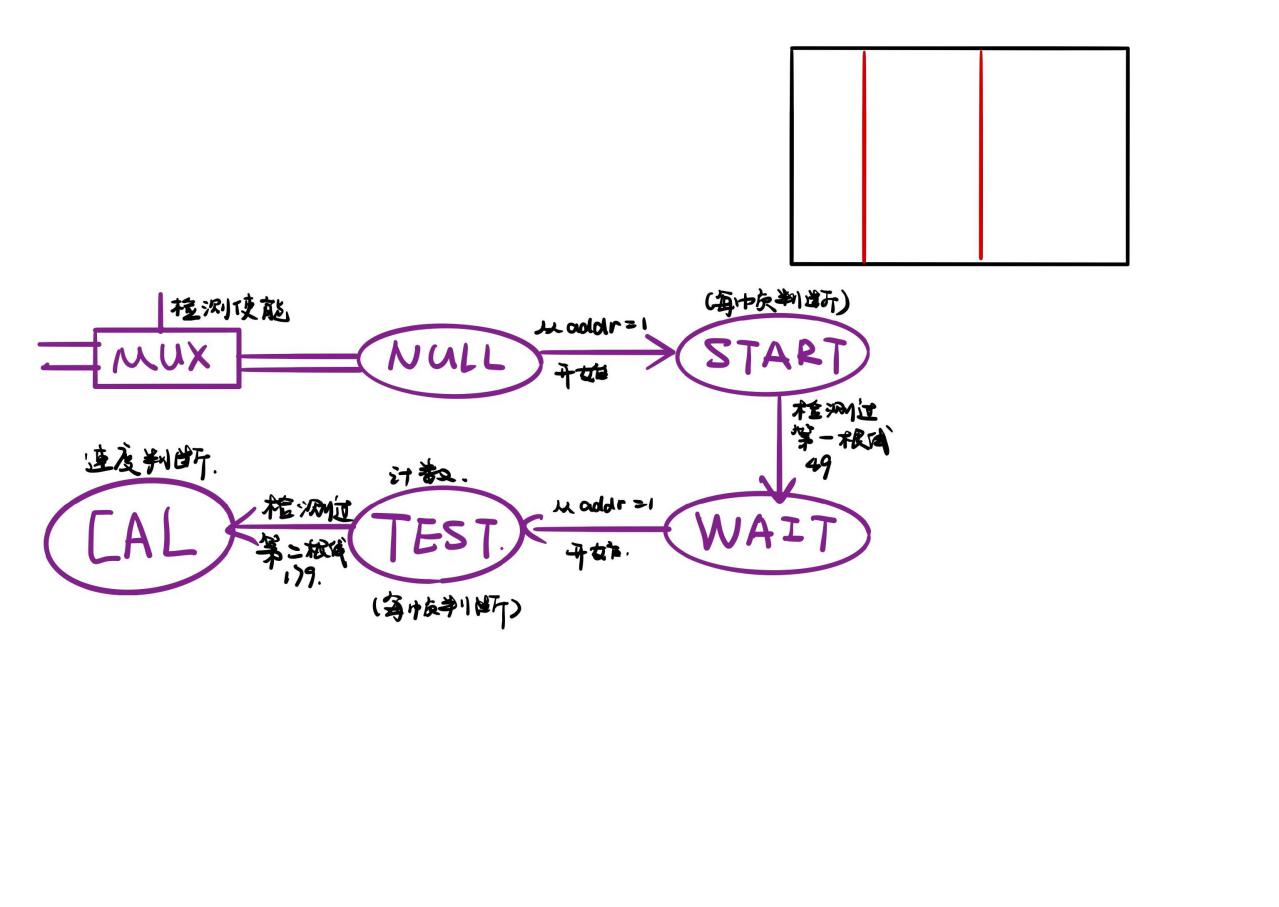


因为VGA的像素点刷新频率高于SD卡的读取频率，所以设置了一个BRAM介于VGA和SD卡之间用于数据的缓冲。具体数据流向是从摄像头图像存储BRAM读取到SD卡，从SD卡读出到缓冲BRAM，再以四位灰度图显示到VGA。写入和读取分别用“使能—读取—跳出”三步状态机来实现。最后用一个照片数量计数器实现多张照片不同地址存贮。

**4. 测速模块**

测速模块是我们设计工程实现的主要功能模块，本质上是一个状态机，对于摄像头测速在不同状态下的转换。测速模块一共分为5个状态，分别是未工作态（NULL）、工作起始态（START）、等待态（WAIT）、测速态（TEST）和计算态（CAL）。

首先有一个开关控制测速功能的工作与否（例如一个拨码开关），当测速器不工作时，边沿检测依旧正常显示，只是状态机不会对检测到的数据进行反馈操作。打开开关后，测速器开始工作，状态机进入NULL态。当检测到有车辆进入监控区域时，状态机进入START态，标志测速模块开始有效工作。当摄像头检测到车辆进入测速区时（到达第一条预设的测速线），状态机变成WAIT态。WAIT态是等待摄像头有效输入（监控区域内有车辆）的状态，与NULL态类似，由于车辆已经进入测速区，故而WAIT在下一时钟周期一定能接收到摄像头的有效输入，所以状态机会很快地进入TEST态。在TEST态，对车辆在经过测速区域的时间计数，当车辆驶离测速区时（到达第二条预设的测速线），状态机进入CAL态进行计算，如果检测到车辆超速，摄像头就会拍照，并且把照片保留显示于屏幕。如果车辆没有超速，直接返回NULL态。拍照完成后，状态机也返回NULL态，等待下一次的有效输入。



**五、设计代码**

代码很长，详见附件工程

**六、仿真（如有）和实现结果**

**七、 改进和发展：**

**1. 实际环境中：**

计算变化差值而非单纯读取边沿检测数据；

摄像头性能：距离、感光、清晰度；

摄像头安装位置，长度和速度计算。

**2. 界面优化：**

VGA上显示速度和超速情况；

存储照片时同时记录拍照日期和车速；

电脑端SD卡图片可视化。

**3. 测速限制：**

结合实际考虑溢出问题。

**八、心得体会及问题**

测速器只能有效检测一辆汽车的行驶状态，如果在对一辆汽车测速的过程中，有其他车辆经过，可能会影响原车的测速过程。同时，由于摄像头的限制，实际路况的光照会对采集的图像的清晰度造成较大影响，从而导致对车辆的测速的精准度有待提高。我们认为这个问题很大程度可以通过改良摄像头和改善边沿检测方法得到解决，例如在sobel算子横向和纵向检测的基础上加入两个斜线方向的测量，或者多种算子并用以检测车辆状态。