

视频解码芯片 SAA7113 的初始化与控制

作者：郑州威科姆电子科技有限公司 张晓健

摘要：本文首先简要介绍了视频解码芯片 SAA7113 的特点与应用，然后介绍了初始化时的寄存器配置，用 51 单片机控制 7113 的方法，最后给出了初始化 7113 的汇编程序及控制方法。

关键词：视频解码 SAA7113 I2C 总线 初始化

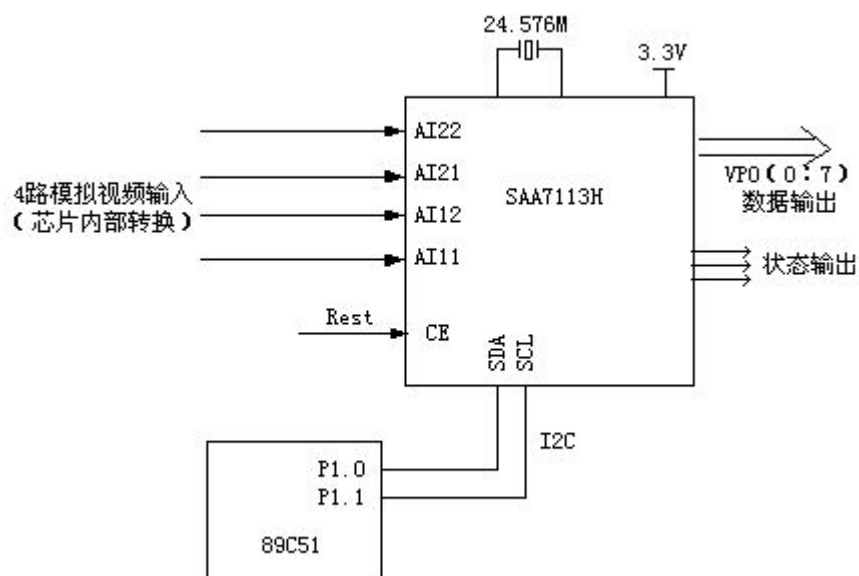
引言：SAA7113 是飞利浦公司视频解码系列芯片的一种，非常具有代表性，在很多视频产品如电视卡、MPEG2、MPEG4 中都有应用，熟悉了 7113 的原理后，对其它系列芯片 SAA7114、7115、7118 就会很容易理解。SAA7113 的主要作用是把输入的模拟视频信号解码成标准的“VPO”数字信号，相当于一种“A/D”器件。7113 兼容全球各种视频标准，在我国应用时必须根据我国的视频标准来配置内部的寄存器，即初始化，否则 7113 就不能按要求输出，可以说对 7113 进行研发的主要工作就是如何初始化。对 7113 初始化需要通过 I2C 总线进行，本文给出用 51 单片机控制的例子。

1. SAA7113 的基本原理与应用

SAA7113 是一种视频解码芯片，它可以输入 4 路模拟视频信号，通过内部寄存器的不同配置可以对 4 路输入进行转换，输入可以为 4 路 CVBS 或 2 路 S 视频 (Y/C) 信号，输出 8 位“VPO”总线，为标准的 ITU 656、YUV 4: 2: 2 格式。

7113 兼容 PAL、NTSC、SECAM 多种制式，可以自动检测场频适用的 50 或 60Hz，可以在 PAL、NTSC 之间自动切换。7113 内部具有一系列寄存器，可以配置为不同的参数，对色度、亮度等的控制都是通过对相应寄存器改写不同的值，寄存器的读写需要通过 I2C 总线进行。

7113 的模拟与数字部分均采用+3.3V 供电，数字 I/O 接口可兼容+5V，正常工作时功耗 0.4W，空闲时为 0.07W。7113 需外接 24.576MHz 晶体，内部具有锁相环 (LLC)，可输出 27MHz 的系统时钟。芯片具有上电自动复位功能，另有外部复位管脚 (CE)，低电平复位，复位以后输出总线变为三态，待复位信号变高后自动恢复，时钟丢失、电源电压降低都会引起芯片的自动复位。7113 为 QFP44 封装。7113 的典型应用如下图所示。



2. SAA7113 的寄存器简要介绍

SAA7113 的地址从 00H 开始，其中 14H、18H~1EH、20H~3FH、63H~FFH 均为保留地址，没有用到，00H、1FH、60H~62H 为只读寄存器，只有以下寄存器可以读写：01H~05H（前端输入通道部分），06H~13H、15H~17H（解码部分），40H~60H（常规分离数据部分）。

下面列表对 7113 中的寄存器进行简要说明，其中默认值为芯片复位后的寄存器默认值，设置值为可以适用于我国 PAL 制式的设置参数，这些参数只供参考，详细信息请参考 7113 数据手册，有些参数如亮度等可以根据用户的需要适当更改。

| 地址 | 寄存器功能 | 默认值 | 设置值 | 配置功能简述 |
|--------|------------|-----|-----|--|
| 00H | 版本号 | 只读 | | |
| 01H | 水平曝光延迟 | 08H | 08H | 推荐值 |
| 02H | 模拟输入控制 1 | C0H | C0H | 选模式 0, 输入通道选择 A111, 输入复合视频信号, 帧后更新关, 放大器及抗锯齿滤波器启动。 |
| 03H | 模拟输入控制 2 | 33H | 33H | 自动曝光通过模式 0-3 控制, 并且启用, 帧包控制关, 长垂直空白, 正常相位。 |
| 04H | 模拟输入控制 3 | 00H | 00H | 静态曝光控制通道 1 取值, 约-3dB。 |
| 05H | 模拟输入控制 4 | 00H | 00H | 静态曝光控制通道 2 取值, 约-3dB。 |
| 06H | 水平同步开始 | E9H | EBH | 对应不同的延迟时间。 |
| 07H | 水平同步停止 | 0DH | EDH | 对应不同的延迟时间。 |
| 08H | 同步控制 | 98H | B8H | 垂直同步设为正常模式, 水平 PLL 关, 水平时间为快锁模式, 选 50Hz, 625 线, 自动场频检测。 |
| 09H | 亮度控制 | 01H | 01H | 光圈: 0.25, 水平更新每线一次, 亮度处理工作, 中心频率 4.1MHz, 前置滤波器不用, CVBS 模式。 |
| 0AH | 亮度控制 | 80H | 80H | 取值 128 (中间值, CCIR 标准) (范围: 0~255) |
| 0BH | 对比度 | 47H | 47H | 取值 1.109 (CCIR 标准) (范围: -2~+2) |
| 0CH | 色度-饱和度 | 40H | 42H | 取值 1.0 (CCIR 标准) (范围: -2~+2) |
| 0DH | 色相控制 | 00H | 01H | 取值 0 (范围: -180~+178) |
| 0EH | 色度控制 | 01H | 01H | 正常带宽 (800KHz), FCTC 正常速度, 按快滤波器工作, PALB CHIN 制式, 不能消除 DTO。 |
| 0FH | 色度获取控制 | 2AH | 0FH | 自动色度控制。 |
| 10H | 格式/延迟补偿 | 00H | 00H | 亮度延迟取值 0, URLN 长度 286, 标准 ITU656 格式。 |
| 11H | 输出控制 1 | 0CH | 0CH | 彩色输出自动控制, VPO 输出不受控制, RTS0, RTS1, RTCO 工作, VPO 可以输出。 |
| 12H | 输出控制 2 | 01H | A7H | RTS0、RTS1 的输出信号选择。 |
| 13H | 输出控制 3 | 00H | 00H | 模拟输出信号的控制。 |
| 14H | 保留 | 00H | 00H | |
| 15H | VGATE 开始 | 00H | 00H | VGATE 的起始脉冲取值。 |
| 16H | VGATE 停止 | 00H | 00H | VGATE 的终止脉冲取值。 |
| 17H | VGATE 高位控制 | 00H | 00H | 配合 15H、16H 使用。 |
| 18-1EH | 保留 | 00H | 00H | |
| 1FH | 解码器状态 | 只读 | | |
| 20-3FH | 保留 | 00H | 00H | |
| 40H | 分离器控制 1 | 02H | 02H | 分离器时钟选择 13.5MHz, 隔行自动搜索, 允许一加代码错误, 场频适合千 50 Hz。 |
| 41-57H | 滤波控制寄存器 | FFH | FFH | 默认值 |
| 58H | 可编程加偏置 | 00H | 00H | 默认值 |
| 59H | 分离器的水平偏置 | 54H | 54H | 推荐值 |
| 5AH | 分离器的垂直偏置 | 07H | 07H | 适合千 50Hz, 625 行。 |
| 5BH | 场偏移, 5A 高位 | 83H | 83H | 默认值 |
| 5C-5DH | 保留 | 00H | 00H | |
| 5EH | 分离器数据鉴别码 | 00H | 00H | 默认值 |
| 5FH | 保留 | 00H | 00H | |
| 60H | 分离器状态 1 | 只读 | | |
| 61H | 分离器状态 2 | 只读 | | |
| 62H | | 只读 | | |

3. SAA7113 寄存器的配置方法

SAA7113 的寄存器配置通过 I2C 总线来进行，遵从 I2C 总线协议，下面从读、写两个方面来说明操作的格式：

对 7113 寄存器的“写”操作：

| | | | | | | | |
|---|-----------------|-------|------------|-------|------|-------|---|
| S | Slave address W | ACK-S | Subaddress | ACK-S | Data | ACK-S | P |
|---|-----------------|-------|------------|-------|------|-------|---|

对 7113 寄存器的“读”操作：

| | | | | | |
|----|-----------------|-------|------------|-------|---|
| S | Slave address W | ACK-S | Subaddress | ACK-S | |
| Sr | Slave address R | ACK-S | Data | ACK-m | P |

说明：S：起始位，条件是 SCL 高电平时 SDA 有下降沿；

Sr：重复设一个起始位

Slave address W：7113 芯片地址+写标志，0100 1010 = 4AH，若 RTS0 通过 3.3K 电阻接地，则为 48H；

Slave address R：7113 芯片地址+写标志，0100 1011 = 4BH，若 RTS0 通过 3.3K 电阻接地，则为 49H；

ACK-S：7113 产生的回应信号；

ACK-m：主机产生的回应信号；

Subaddress：寄存器地址；

P：停止位，条件是 SCL 高电平时 SDA 有上升沿；

对多个寄存器操作时，寄存器地址有自动加 1 功能。

4. 用 51 单片机对 7113 初始化和控制

SAA7113 的初始化就是对寄存器配置合适的参数，使其能够有符合要求的输出。寄存器配置通过 I2C 总线来进行，很多可以控制 I2C 总线的器件都可以作为主器件对 7113 进行初始化，这里介绍用 51 单片机初始化 7113 的例子。

51 单片机和 7113 的硬件连接非常简单，只要把单片机的两个 I/O 口（如 P1.0、P1.0）直接和 7113 的 SCL、SDA 管脚相连，再加上上拉电阻即可。

用单片机初始化 7113 的主要任务是程序的编写，首先要熟悉 I2C 总线协议，根据 I2C 总线的原理写出启动、停止、应答信号等的子程序，由子程序再写出发送、接收 1 个字节的程序，然后根据 7113 的寄

寄存器操作格式写出读写寄存器的程序，最后根据以上的子程序写出初始化 7113 的程序段。

对 7113 的控制一般是改变色度、亮度等指标以及输出管脚的输出信号，这可以通过修改相应寄存器的值来完成，程序上写出“读写命令”即可。

下面以程序段的形式给出初始化 SAA7113 以及读写寄存器的具体例子，以供参考。

```
SDA          BIT   P1.0
```

```
SCL          BIT   P1.1
```

```
I2C_ERROR    BIT   00H    ; I2C 总线数据传输出错标志
```

```
DeviceaddressW EQU   4AH    ; 7113 器件地址+写
```

```
DeviceaddressR EQU   4BH    ; 7113 器件地址+读
```

```
Subaddress    EQU   4DH    ; 7113 寄存器地址字节在单片机中的存放地址
```

```
DATA_I2C      EQU   50H    ; 设置写入或读出数据在单片机中的存放地址
```

```
; *****启动*****
```

```
I2C_START:   SETB   SDA
```

```
                NOP
```

```
                SETB  SCL
```

```
                NOP
```

```
                CLR   SDA
```

```
                NOP
```

```
                CLR   SCL
```

```
                RET
```

```
; *****停止*****
```

```
I2C_STOP:    CLR    SDA
```

```
                NOP
```

```
                SETB  SCL
```

```
                NOP
```

```
                SETB  SDA
```

```
                NOP
```

```
                RET
```

*****送应答位*****

SEND_ACK: CLR SCL

NOP

CLR SDA

NOP

SETB SCL

NOP

NOP

CLR SCL

NOP

SETB SDA

RET

*****送非应答位*****

SEND_NOACK: SETB SDA

NOP

SETB SCL

NOP

NOP

CLR SCL

NOP

RET

*****检查应答位*****

CHECK_ACK: NOP

CLR SCL

NOP

SETB SDA

NOP

SETB SCL

```

    NOP

    NOP

    MOV    C, SDA

    MOV    I2C_ERROR, C

    CLR    SCL

    NOP

    RET

;*****发送 1 字节数据，待送数据在 A 中*****

I2C_SEND_1BYTE:

    MOV    R0, #8

SEND100: RLC    A

    MOV    SDA, C

    NOP

    SETB   SCL

    NOP

    NOP

    CLR    SCL

    DJNZ   R0, SEND100

    RET

;*****接收 1 字节数据，接收数据放在 A 中*****

I2C_RECEIVE_1BYTE:

    MOV    R0, #8

RCV100:  SETB   SDA

    NOP

    SETB   SCL

    NOP

    NOP

    NOP
```

```
MOV C, SDA

CLR SCL

RLC A

DJNZ R0, RECV100

RET

; *****通过 I2C 总线向某一寄存器写入一个字节数据*****

I2C_WRITE: ACALL I2C_START ; 发启动信号

MOV A, # DeviceaddressW ; 调 7113 地址+写

ACALL I2C_SEND_1BYTE ; 发送 7113 地址及“写”命令

ACALL CHECK_ACK ; 检查 7113 的应答信号

JNB I2C_ERROR, WR200 ; 应答正确, 继续

WR100: ACALL I2C_STOP ; 应答不对, 返回

RET

WR200: MOV A, Subaddress ; 调寄存器地址

ACALL I2C_SEND_1BYTE ; 发送寄存器地址

ACALL CHECK_ACK ; 检查 7113 的应答信号

JB I2C_ERROR, WR100 ; 应答不对, 返回

MOV A, DATA_I2C ; 调准备写入的数据

ACALL I2C_SEND_1BYTE ; 发送数据字节

ACALL CHECK_ACK

JB I2C_ERROR, WR100

ACALL I2C_STOP ; 发停止信号

RET

; *****通过 I2C 总线读出某一寄存器的数据*****

I2C_READ: ACALL I2C_START

MOV A, # DeviceaddressW ; 调 7113 地址, 写入

ACALL I2C_SEND_1BYTE

ACALL CHECK_ACK
```



```
JNB    I2C_ERROR, RD200

RD100:  ACALL I2C_STOP

        RET

RD200:  MOV  A, Subaddress      ; 调要读的寄存器地址

        ACALL I2C_SEND_1BYTE   ; 发送寄存器地址字节

        ACALL CHECK_ACK

JB      I2C_ERROR, RD100

ACALL I2C_START      ; 重发起动信号

MOV  A, # DeviceaddressR    ; 调 7113 地址, 读

        ACALL I2C_SEND_1BYTE

        ACALL CHECK_ACK

JB      I2C_ERROR, RD100

ACALL I2C_RECEIVE_1BYTE ; 接收读出的数据

MOV  DATA_I2C, A      ; 读出数据转存

ACALL SEND_NOACK      ; 发送非应答位

ACALL I2C_STOP        ; 停止

        RET

; *****初始化 7113, 配置各寄存器*****

INIT_SAA7113:  MOV DPTR, #SAA7113_Subaddress

               MOV R7, #28

INIT100:      MOV A, #0

               MOVC A, @A+DPTR

               MOV Subaddress, A    ; 调寄存器地址

               MOV A, #28

               MOVC A, @A+DPTR

               MOV DATA_I2C, A      ; 调寄存器配置数据

               INC DPTR

               ACALL I2C_WRITE      ; 配置 1 个寄存器
```

```
JB I2C_ERROR, INIT200

DJNZ R7, INIT100

INIT200:    RET

; *****SAA7113 寄存器初始化配置数据*****

SAA7113_Subaddress:

DB 01H,02H,03H,04H,05H,06H,07H,08H,09H,0AH,0BH,0CH,0DH,0EH

DB 0FH,10H,11H,12H,13H,15H,16H,17H,40H,58H,59H,5AH,5BH,5EH

; 共 28 个

I2C_REG_VALUE_AI11:

DB 08H,0C0H,33H,00H,00H,0EBH,0E0H,0B8H,01H,7EH,46H,43H,01H,01H

DB 0FH,00H,0CH,0A7H,00H,00H,00H,00H,02H,00H,54H,07H,80H,00H

; *****对 SAA7113 某一寄存器的改写与读出*****

WRITE_READ:    MOV  Subaddress, #0AH ; 设寄存器地址为 0AH

                MOV  DATA_I2C, #88H ; 改寄存器的值为 88H

                ACALL I2C_WRITE ; 改写

                ACALL I2C_READ ; 读出
```

结语

SAA7113 在很多产品中都可以应用，但其初始化与控制的原理都一样，本文中的程序段经过实际应用可以保证 7113 正常工作，其寄存器设置参数与控制方法可以被借鉴或直接应用。

参考文献

1. 陈露晨主编. 计算机通信接口技术. 成都: 电子科技大学出版社, 1999
2. 张洪润主编. 单片机应用技术教程. 北京: 清华大学出版社, 1997
3. SAA7113H Product specification / Data sheet . PHILIPS, 1999