



SiGma Micro  
IC Solution Designing

# 规格说明书

---

**SGB60721**

**72 Seconds Voice Synthesizer IC**

版本 1.1

希格玛保留修改此文件而不预先通知的权利。

## 目 录

1	概述 .....	3
2	特性 .....	3
3	管脚说明 .....	4
4	PAD 图 .....	4
5	额定值范围 .....	5
6	电气特性 .....	6
7	功能描述 .....	7
7.1	只读存储器(ROM) .....	7
7.2	随机存取存储器(RAM) .....	7
7.3	I/O .....	8
7.4	算术逻辑单元(ALU) .....	8
7.5	内部定时器 .....	9
7.6	看门狗定时器 .....	9
7.7	语音合成器 .....	9
7.8	其它指令 .....	10
7.9	指令集 .....	10
8	汇编程序格式 .....	11
9	应用电路图 .....	12
10	修改记录 .....	13

## 1 概述

SGB60721 是一种 72 秒语音合成芯片，包含 12 个 I/O 脚和内置微控制器。用户可以通过对 MCU 的编程来实现应用。

## 2 特性

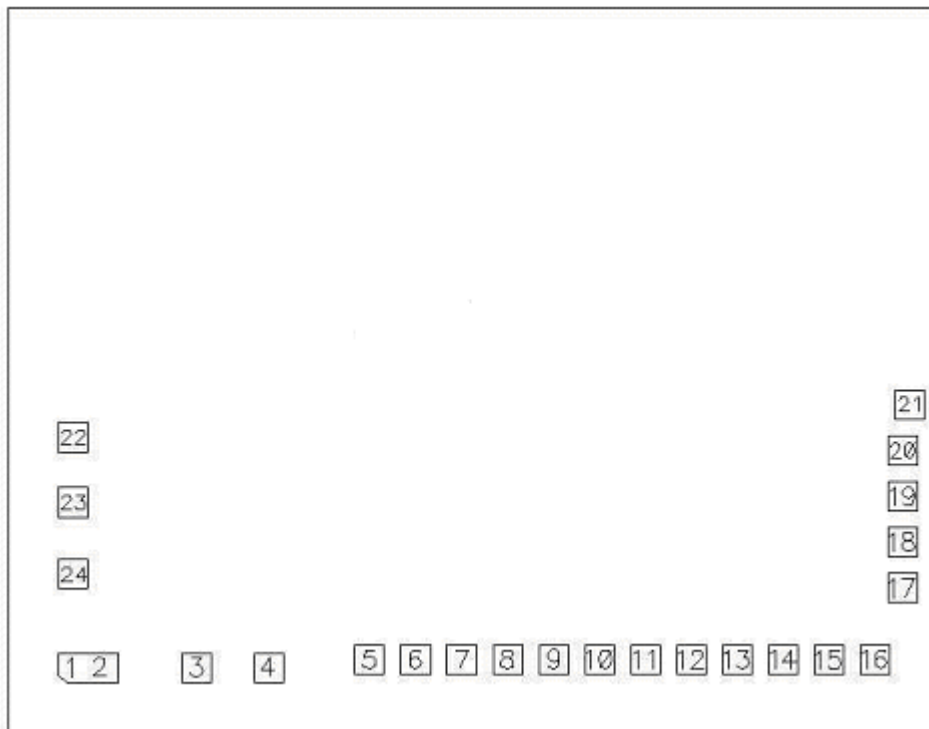
- 单一电源供电：2.4V – 5.5V
- 内置微控制器
- 12 个 I/O 脚，其中 1 个可用于 38KHz 载波输出
- 与系统振荡器配合的看门狗定时器 (WDT)，可编程控制
- 2 种时钟发生器—RC 振荡器和晶体振荡器
- 外部复位
- 32\*4 bits RAM
- 16K ROM
- 高品质语音合成器
- 放音频率 1.5k ~ 27kHz
- 运用 BDNC 的噪音修正运算法则尽可能的消除噪音
- PWM 模式直接驱动扬声器

### 3 管脚说明

管脚名称	I/O	功能描述
P20~P23	I/O	I/O 端口 2.
P30~P33	I/O	I/O 端口 3.
P40~P43	I/O	I/O 端口 4.
OSC_XIN	I	振荡电阻连接端/晶振输入端
XOUT	O	晶振输出
TEST	I	测试脚（内部下拉，高电平有效）
EXT_RES	I	外部复位端（内部下拉，高电平有效）
PWM1	O	PWM1 输出
PWM2	O	PWM2 输出
VDD (3)*	POW	电源
GND (3)*	POW	地

注意：SGB60721 中 VDD 和 GND 各有 3 个

### 4 PAD 图



衬底面积：3000  $\mu\text{m}$   $\times$  2500  $\mu\text{m}$  (衬底接地)

管脚号码	管脚名称	SGB60721		管脚号码	管脚名称	SGB60721	
		X	Y			X	Y
1	VDD	177.50	179.95	13	P4<3>	1982.90	201.70
2	VDD	257.50	179.95	14	P4<2>	2107.90	201.70
3	PWM1	512.90	179.95	15	P4<1>	2232.90	201.70
4	GND	708.65	179.95	16	P4<0>	2357.90	201.70
5	P2<3>	982.90	201.70	17	TEST	2432.65	397.30
6	P2<2>	1107.90	201.70	18	RES_EXT	2432.65	522.30
7	P2<1>	1232.90	201.70	19	OSC_XIN	2432.65	647.30
8	P2<0>	1357.90	201.70	20	XOUT	2432.65	772.30
9	P3<3>	1482.90	201.70	21	GND	2450.80	897.30
10	P3<2>	1607.90	201.70	22	VDD	177.50	808.50
11	P3<1>	1732.90	201.70	23	GND	177.50	631.10
12	P3<0>	1857.90	201.70	24	PWM2	177.50	435.35

## 5 额定值范围

项目	参数	最小值	最大值	单位
电压	$V_{DD}$	-0.3	5.0	V
输入电压	$V_{IN}$	$V_{SS}-0.3$	$V_{DD}+0.3$	V
工作环境温度	$T_{OP}$	-20.0	70.0	°C
存储温度	$T_{STG}$	-55.0	125.0	°C

## 6 电气特性

项目	参数	最小值	典型值	最大值	单位	测试条件
工作电压	$V_{DD}$	2.4	3.0	5.5	V	
静态电流	$I_{SBY}$	-	-	1.0	$\mu A$	$V_{DD}=3V$ 无负载
工作电流	$I_{OP}$	-	-	1200	$\mu A$	$V_{DD}=3V$ , 无负载
P2, P3, P4 输入电流	$I_i$	-	7	-	$\mu A$	$V_{DD}=3V$
P2, P3, P4 驱动电流 (高电平)	$I_{OD}$	1.5	2	-	mA	$V_{DD}=3V, V_O=2.4V$
P2, P3, P4 驱动电流 (低电平)	$I_{OS}$	2.0	3	-	mA	$V_{DD}=3V, V_O=0.4V$
PWM 驱动电流	$I_{pwm}$	-	180	-	mA	$V_{DD}=3V, R_L=8$
振荡电阻	$R_{osc}$	-	80	-	K	$V_{DD}=3V$
振荡频率	$F_{OSC}$	-	4	-	MHz	$V_{DD}=3V$

## 7 功能描述

### 7.1 只读存储器(ROM)

SGB60721 的 ROM 大小、结构如下所示：

程序 (最大 16k* 10)	内部 ROM 大小为 2E000 (hex) * 10 位
语音片段	
语音片段	
...	

有 2 个指针用于 ROM 寻址：

程序计数器 (14 位)

SGB60721 有一个程序计数器和 3 个程序计数器堆栈 (STK1, STK2, STK3)。

程序计数器用于指向下一条被执行的指令。

程序计数器堆栈用于存储返回的程序地址。执行指令之后，除非执行了跳转指令或条件跳转指令成立，否则程序计数器的值增加 1，指向下一条指令。

以下的指令会改变程序计数器：

JMP label：跳转到子程序

CALL1 label：执行子程序并且将返回地址存储于 STK1

RET1：将 STK1 的内容恢复给程序计数器

CALL2 label：执行子程序并且将返回地址存储于 STK2

RET2：将 STK2 的内容恢复给程序计数器

CALL3 label：执行子程序并且将返回地址存储于 STK3

RET3：将 STK3 的内容恢复给程序计数器

CJP label：如果内部播放标记位“PF”设置为 1，则跳转到分支程序

CJC label：如果内部计数器标记位“CF”设置为 1，则跳转到分支程序

CAJE #data label：如果寄存器 A 的值等于 data (data 在 0000b ~1111b 范围)，则跳转到子程序

CAJE M, label：如果寄存器 A 的值等于存储器的值 (M 在 m0~m15 范围内)，则跳转到子程序

语音地址计数器(SAC)

SGB60721 有一个语音地址计数器，用于指向下一个放音数据的 ROM 地址。

### 7.2 随机存取存储器(RAM)

SGB60721 有 32 个 RAM，被分为 2 个 bank，可以通过指令 BANK0 和 BANK1 选择 (如下所示)：

BANK0：选择第一个 bank 的 RAM，并把第 1 ~ 16 的 RAM 用符号 M0 ~ M15 标识。

BANK1：选择第二个 bank 的 RAM，并把第 17 ~ 32 的 RAM 用符号 M0 ~ M15 标识。

BANK0	m0-m15
BANK1	m0-m15

立即数可以通过指令被写入到 RAM：

MOV M, #data：传送立即数 data 到 M (M 在 m0~m15 范围内，data 在 0000b ~1111b 范围内)。

### 7.3 I/O

SGB60721 有 3 个 4 位 I/O 端口。相关指令的描述如下：

MOV A, P2：从 I/O 端口 2 (管脚：P20 到 P23) 读取数据到寄存器 A

MOV A, P3：从 I/O 端口 3 (管脚：P30 到 P33) 读取数据到寄存器 A

MOV A, P4：从 I/O 端口 4 (管脚：P40 到 P43) 读取数据到寄存器 A

MOV P2, A：将寄存器 A 的数据写到 I/O 端口 2 (管脚：P20 到 P23)

MOV P3, A：将寄存器 A 的数据写到 I/O 端口 3 (管脚：P30 到 P33)

MOV P4, A：将寄存器 A 的数据写到 I/O 端口 4 (管脚：P40 到 P43)

对每一个 I/O 端口，都有 2 个可选择的下拉电阻：

Case0：小的下拉电阻，低电平驱动电流大

Case1：大的下拉电阻，低电平驱动电流小

这 2 种情况可以通过状态寄存器 P2C、P3C、P4C 来选择。

如果 P2C/P3C/P4C 设置为 1，端口选择 case1；反之，则选择 case0。相关指令如下：

MOV P2C A：将寄存器 A 的数据写到寄存器 P2C。

MOV P3C A：将寄存器 A 的数据写到寄存器 P3C。

MOV P4C A：将寄存器 A 的数据写到寄存器 P4C。

### 7.4 算术逻辑单元(ALU)

SGB60721 内置 1 个 4 位 ALU，一些简单的算术和逻辑功能都是通过一个 4 位寄存器 A 来实现。

INCA：寄存器 A 的值增加 1

AND A, M：寄存器 A 的值和存储器的值执行“与”运算，然后将结果存到寄存器 A，M 在 m0 ~m15 范围内。

OR A, M：寄存器 A 的值和存储器的值执行“或”运算，然后将结果存到寄存器 A，M 在 m0 ~m15 范围内。

XOR A, M：寄存器 A 的值和存储器的值执行“异或”运算，然后将结果存到寄存器 A，M 在 m0 ~m15 范围内。



MOV A #N : 传送立即数到寄存器 A , 立即数在 0000b~1111b 范围内。

MOV A, M : 传送存储器 M 的值到寄存器 A , M 在 m0 ~m15 范围内。

MOV M, A : 传送寄存器 A 的值到存储器 M , M 在 m0 ~m15 范围内。

## 7.5 内部定时器

SGB60721 有一个 8 位定时器 , 每 1 个指令周期 , 定时器增加一次。当 8 位定时器从 11111111 计数到 00000000 , 计数标记位 “ CF ” 被设置为 1 , 直到执行指令 CLRC , CLRCF 后 “ CF ” 才自动复位。

CLRC : “ CF ” 自动复位设置为 0 , 且 8-bit 定时器为 00000000。

CLRCF : “ CF ” 自动复位设置为 0 , 但 8-bit 定时器继续计数。

## 7.6 看门狗定时器

当看门狗使能 , 指令 WDT\_CLR 每 65536 个指令周期内必须被执行。让看门狗知道程序仍然正确的运行。如果每 65536 个指令周期 , 软件不能把看门狗定时器清零 , 看门狗将重新启动系统。

WDT\_CLR : WDT 清零

## 7.7 语音合成器

SGB60721 提供一个内置语音合成器。当命令 “ play sec\_num ” 被执行 , 指定的语音段起始地址被读取到语音地址计数器(SAC) , 且播放标记位被置为 1。PF 将保持值为 1 直到 :

1. 语音段播放结束
2. 执行指令 “ STOP ”
3. 进入睡眠模式 ( 执行指令 “ END ” )

SAC 的计算速度由寄存器 RH、 RL 决定。RH 是 2-bit 寄存器 , RL 是 4-bit 寄存器 , 它们可以合并为 6-bit RATE 寄存器 :

语音播放速度 (SAC 的计算速度) 如下 :

语音播放速度=1/((56-RATE)\*12 us)。

例如 : 播放速度= 8k , 每个播放样品的持续时间约为 125us , RATE= 46 (RH=0010B, RL=1110) , 因为 (56-46)\*12 = 120us。

相关指令描述如下 :

PLAY sec\_num : 如果 sec\_num 是一个语音文件 , 且播放标记 “ PF ” 位置 1 , 则读取起始地址到 SAC。

STOP : 标记位 PF 清零 , 且强制当前语音播放停止。

MOV RH, A : 从寄存器 A 传送数据到寄存器 RH。

MOV RL, A : 从寄存器 A 传送数据到寄存器 RL。

## 7.8 其它指令

NOP : 空操作

END : 进入睡眠模式

## 7.9 指令集

1 个指令周期等于 48 个振荡周期。

指令语句	指令周期	字节
JMP label	2	2
CALL1 label	2	2
RET1	1	1
CALL2 label	2	2
RET2	1	1
CALL3 label	2	2
RET3	1	1
CJP label	2	2
CJC label	2	2
PLAY sec_num	2	2
BANK0	1	1
BANK1	1	1
STOP	1	1
WDTC	1	1
CLRC	1	1
CLRCF	1	1
INCA	1	1
CAJE #n, label	3	3
CAJE m, label	3	3
MOV p2, a	1	1
MOV p3, a	1	1
MOV p4, a	1	1
MOV p2C, a	1	1
MOV p3C, a	1	1
MOV p4C, a	1	1
MOV rl, a	1	1
MOV rh, a	1	1

MOV m, a	1	1
MOV m, #n	1	1
MOV a, p2	1	1
MOV a, p3	1	1
MOV a, p4	1	1
MOV a, m	1	1
MOV a, #n	1	1
AND a, m	1	1
XOR a, m	1	1
OR a, m	1	1
NOP	1	1
END	1	1

## 8 汇编程序格式

程序格式：

```

;Comment

NTEMP EQU M0                ;EQU DEFINE

.....

Program                      ; Assemble program will begin here
Start:
    .....
    .....
    end
Wake_up:
    .....
    .....
  
```

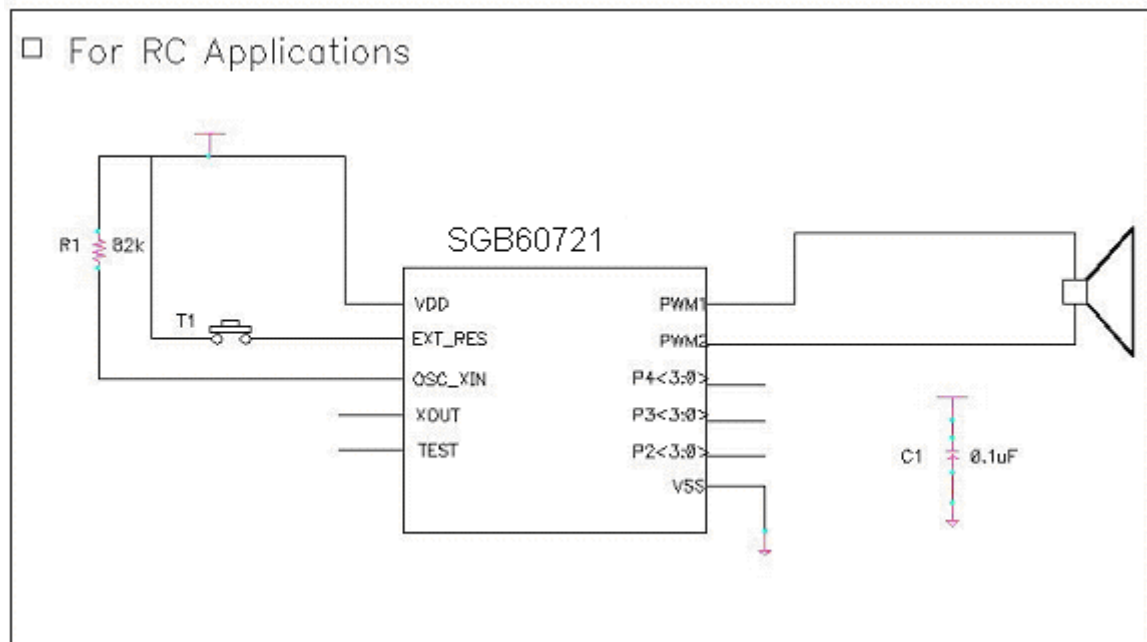
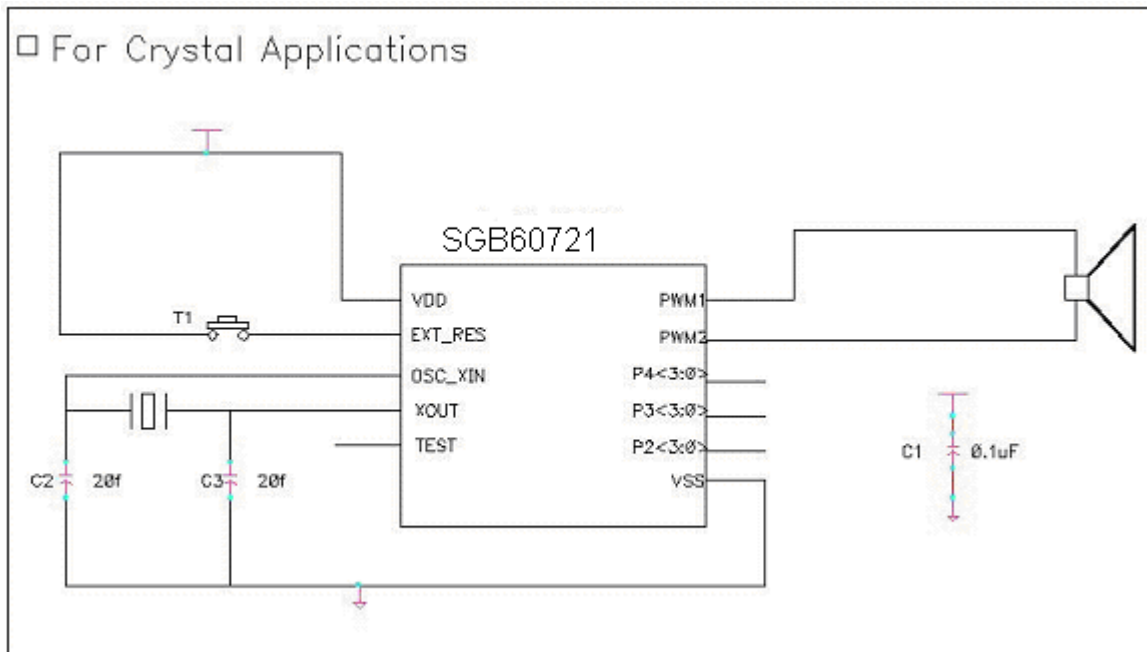
汇编程序文件：

格式声明“ Program ”，说明程序文件的格式是汇编程序的格式。关键字“ Program ”后面是应用程序，在程序里必须存在“ Start ”和“ Wake\_up ”2个标号。当上电（电池）时，程序由标号“ Start ”开始。睡眠模式期间，如果任何 I/O 脚（端口 2、端口 3、端口 4）状态的改变都具有唤醒作用，程序将由标号“ Wake\_up ”开始执行。

文件系统：

文件类型	Sub_name	描述
汇编程序	Name.asm	用户的应用汇编程序
语音文件	Name.wav	语音数据文件

## 9 应用电路图



## 10 修改记录

版本	更新时间	修改内容	修改人
V0.9.1	2005-1-6	Draft	Wangcy
V0.9.2	2005-1-11	Add port4	Wangcy
V0.9.3	2005.3.24	ADD PAD LOCATION	
V0.9.4	2005.4.21	ADD APPLICATION CURCUIT	WANGCY
V1.0	2005-4-21	Modify the electrical characteristics	沈继铭
V1.1	2005-5-11	Modify the electrical characteristics	沈继铭

版本	北京确认人	确认时间	深圳确认人	确认时间
V1.0			楚振全	2005-4-21
V1.1			楚振全	2005-5-11