



## MSK010B 使用资料

(适用于 MSK010B V2.1 模块)

### 目 录

1、产品特征.....	2
2、应用范围.....	2
3、引脚图示.....	2
4、引脚描述.....	2
5、控制模式.....	3
5.1、按键触发模式.....	3
5.1.1、脉冲可重复触发.....	3
5.1.2、脉冲不可重复触发.....	3
5.1.3、电平保持可循环.....	3
5.1.4、电平保持不循环.....	4
5.1.5、播放/暂停.....	4
5.1.6、下一曲不循环.....	4
5.1.7、上一曲不循环.....	5
5.1.8、下一曲可循环.....	5
5.1.9、上一曲可循环.....	5
5.1.10、停止.....	5
5.2、一线串口触发模式.....	6
5.2.1、I/O 口分配表.....	6
5.2.2、命令码描述.....	6
5.2.3、控制时序.....	6
5.2.4、一线串口控制程序范例.....	7
6、典型应用电路.....	11
6.1、按键控制 PWM 输出应用电路.....	11
6.2、按键控制 DAC 输出应用电路.....	11
6.3、一线串口 PWM 输出应用电路.....	11
6.4、一线串口 DAC 输出应用电路.....	11
7、封装尺寸图.....	12
8、历史版本记录.....	12

### 1、产品特征

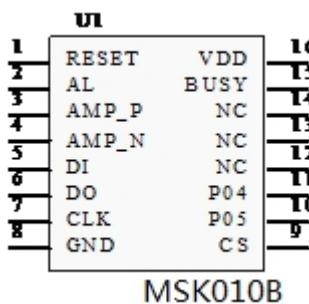
- 支持 PCM、ADPCM、WAV 音频格式播放，音质优美；
- 语音制作非常方便，用户可以直接通过上位机，可由 mp3、wmv 等格式转换得到；

- 支持外挂最大 64M SPI Flash 作为存储语音的载体，语音长度可达 2000 多秒；
- 最多可支持 3328 段语音；
- 用下载器通过 USB 连接电脑，用上位机软件直接更新语音数据；
- 支持两种控制模式：按键模式，一线串口模式；
- 支持进行 3327 段语音的播放；
- 工作电压 DC2.8~3.6V。

## 2、应用范围

本产品稳定性好，音质出色，适合应用在汽车电子，高级玩具，游戏机，家用电器，安防系统，喊话器等场合。

## 3、引脚图示



仅适用于 MSK010B V2.1 模块

## 4、引脚描述

封装引脚	引脚标号	简述	功能描述
1	/RESET	/RESET	复位脚，低电平保持 5ms 有效
2	DAC	DAC	DAC 音频输出
3	PWM+	PWM+	PWM+音频输出，可跟 PWM-驱动扬声器
4	PWM-	PWM-	PWM-音频输出，可跟 PWM+驱动扬声器
5	DI	SPI-FLASH_DI	连接到 SPI-FLASH_DI，下载数据用
6	DO	SPI-FLASH_DO	连接到 SPI-FLASH_DO，下载数据用
7	CLK	SPI-FLASH_CLK	连接到 SPI-FLASH_CLK，下载数据用
8	GND	GND	地线
9	CS	SPI-FLASH_CS	连接到 SPI-FLASH_CS，下载数据用
10	P05	I/O	按键 K2/一线串口 DATA
11	P04	I/O	按键 K1
12	NC	空	空
13	NC	空	空
14	NC	空	空
15	BUSY	BUSY	忙信号输出
16	VCC	VCC	电源正极

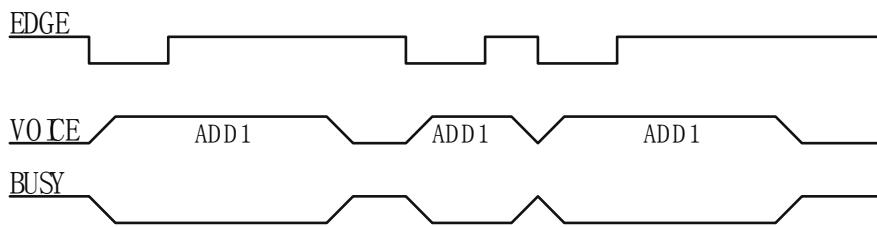
## 5、控制模式

MSK010B 具有按键控制模式，一线串口控制模式。

## 5.1、按键触发模式

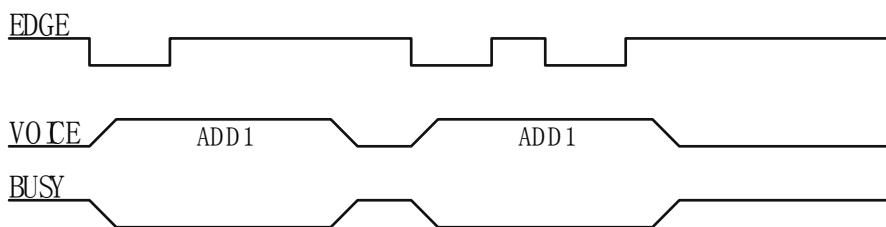
所定义的管脚可以直接触发芯片放音，即每一个管脚可控制播放一段语音。每个管脚的触发方式可单独设置。按键控制模式的防抖动时间为 40ms。按键触发模组包括脉冲可重复触发、脉冲不可重复触发、电平保持循环、电平保持不循环、语音开关播放、下一曲、上一曲、下一曲循环、上一曲循环。详细请见下时序图

### 5.1.1、脉冲可重复触发



备注：负脉冲触发。当 I/O 口检测到有下降沿时（如，该 I/O 口对地短路一下），触发播放语音。在语音播放期间，再检测到下降沿，芯片会打断正在播放的语音，重新播放。只要有下降沿信号，就重新播放。

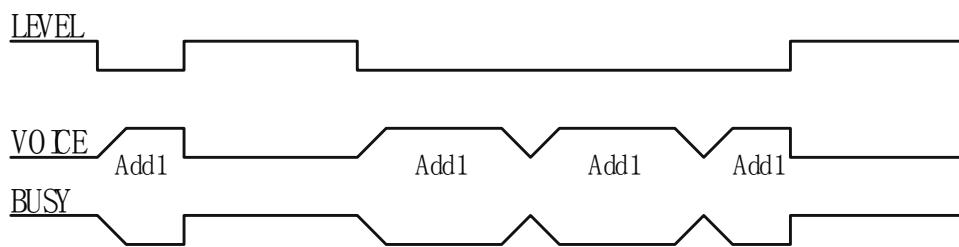
### 5.1.2、脉冲不可重复触发



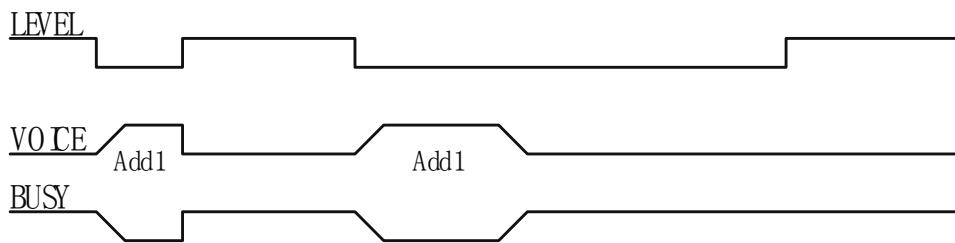
备注：负脉冲触发。当 I/O 口检测到有下降沿时（如，该 I/O 口对地短路一下），触发播放语音。在语音播放期间，再检测到下降沿时，芯片不动作。直到语音结束后，检测到的下降沿才有效。

### 5.1.3、电平保持可循环

备注：当 I/O 口为低电平时，保持播放，高电平则停止。当第一遍结束后，还保持低电平，则继续重新播放，直到转变为高电平才停止。只要是低电平，则有声音；高电平，则没声音。

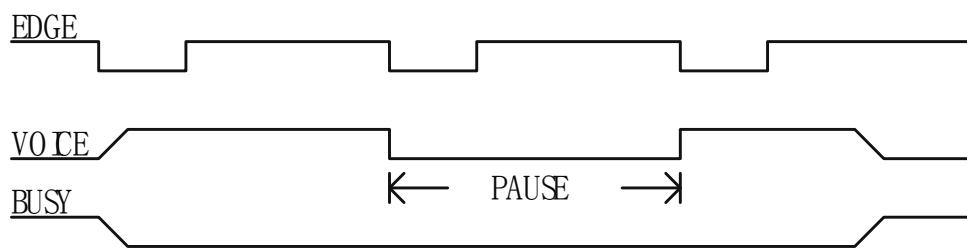


## 5.1.4、电平保持不循环



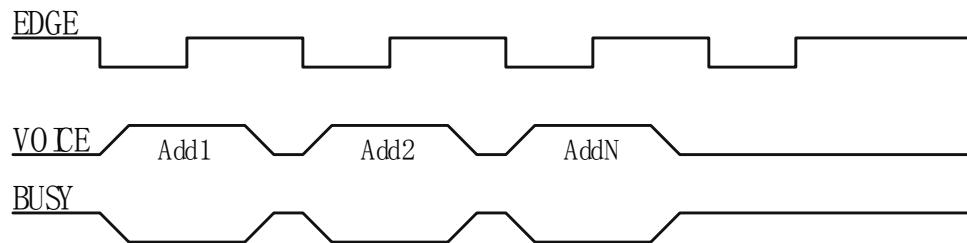
备注：低电平触发。当I/O口为低电平时，保持播放，高电平则停止。当第一遍播放结束后，还保持低电平，也不会继续播放，触发后只播放一次就结束。如果需要重新播放，则需要让I/O口处于高电平，再拉为低电平，而后保持低电平即可。

## 5.1.5、播放/暂停



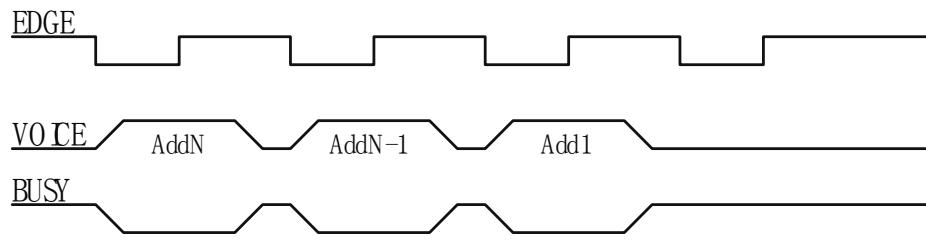
备注：负脉冲触发。负脉冲开始播放，下一个负脉冲结束。不管声音是处于播放还是停止状态，都遵照这个规则。

## 5.1.6、下一曲不循环



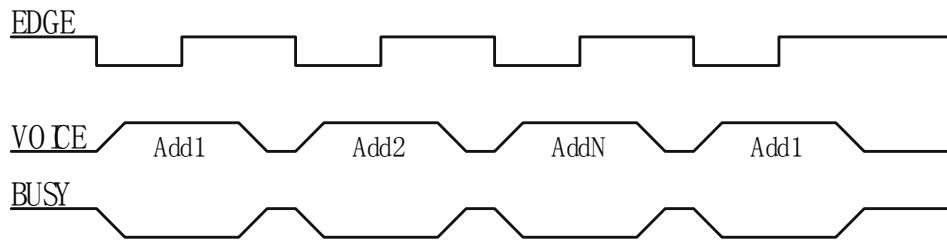
备注：负脉冲触发。用一个按键触发播放声音。一个负脉冲触发播放一段，下一个负脉冲播放下一段，播放完最后一段，则不会再有声音。重复操作，只能播放到最后一段声音。

## 5.1.7、上一曲不循环



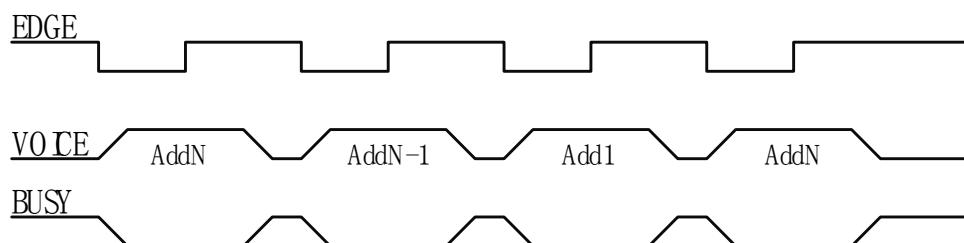
备注：负脉冲触发。用一个按键触发播放语音。一个负脉冲触发播放一段语音，下一个负脉冲播放上一段语音，播放完最前一段，则不再向前触发播放语音。重复操作，只能播放到最前一段声音。

## 5.1.8、下一曲可循环



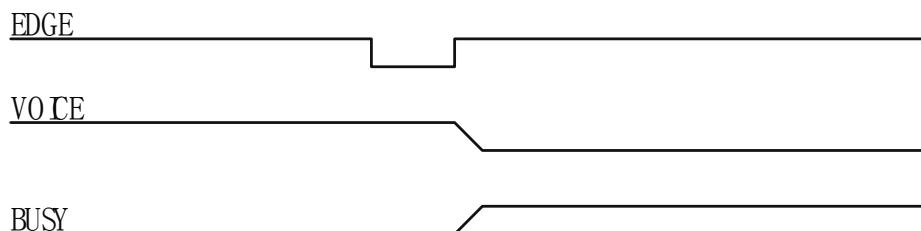
备注：负脉冲触发。用一个按键触发播放语音。一个负脉冲触发播放一段语音，下一个脉冲触发播放下一段语音，重复操作，播放完最后一段语音，则会点播到第一段语音，如此循环触发播放语音。

## 5.1.9、上一曲可循环



备注：负脉冲触发。用一个按键触发播放语音。一个负脉冲触发播放一段语音，下一个脉冲触发播放上一段语音，重复操作，播放完最前一段语音，则会点播到最后一段语音，如此循环触发播放语音。

## 5.1.10、停止



备注：负脉冲触发。用一个按键触发播放语音。一个负脉冲触发播放一段语音，下一个脉冲触发播放上一段语音，重复操作，播放完最前一段语音，则会点播到最后一段语音，如此循环触发播放语音。

## 5.2、一线串口触发模式

通过一根数据线发送串口数据，时序协议为占空比数据位，一线串口可以实现语音播放及控制命令等功能。

## 5.2.1、I/O 口分配表

封装形式	管脚	
	P04	P05
MSK010B	---	DATA

## 5.2.2、命令码描述

命令码	定义	描述
E0H~E7H	音量调节	在语音播放或者待机状态发此命令可以调节 8 级音量，E0H 最小，E7H 音量最大。
F2H	循环播放	发送此命令可循环播放当前地址语音。
F3H	停止循环播放	发送此命令可停止循环播放当前地址语音。
FEH	停止语音播放	停止播放语音命令。

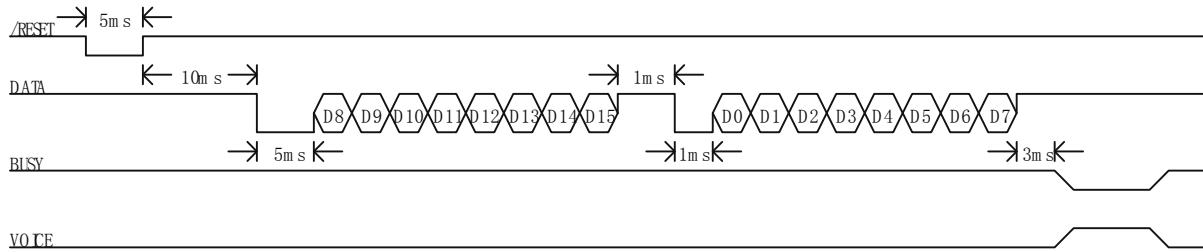
## 语音地址对应关系

数据（十六进制）	功能
00 00H	播放第 1 段语音
00 01H	播放第 2 段语音
00 02H	播放第 3 段语音
.....	.....
0C FDH	播放第 3325 段语音
0C FEH	播放第 3326 段语音
0C FFH	播放第 3327 段语音

## 5.2.3、控制时序

## 发送地址数据

一线串口控制时序只有一条数据通信线，依照电平占空比不同来代表不同的数据位。高电平与低电平数据占空比位 1: 3 即代表数据位 0，高电平与低电平数据位占空比为 3: 1 代表数据位 1，高电平在前，低电平在后。地址数据需要发送两个字节，先发高字节再发低字节。单个字节中数据先发低位再发高位。发送完第一个字节，把 DATA 拉高等待 1ms，再发送第二个字节。如要触发第 300 段语音，就发送 01H+2CH。详细时序请见下图：

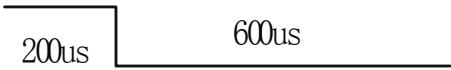


## 发送控制命令

控制命令只需发送一个字节即可



DATA 为一线串口数据通信线，复位信号发送完成等待 5ms，先拉低 5ms 以唤醒芯片，再发送数据，数据先发低位再发高位。数据位占空比对应请见下图。



高电平: 低电平=1: 3 表示 0



高电平: 低电平=3: 1 表示 1

#### 5.2.4、一线串口控制程序范例

```
//MCU: STC10F04
```

```
//晶振: 11.0592MHz
```

```
#include "reg51.h" /* reg51 头文件 */
```

```
sbit RESET = P0^0; //复位端
```

```
sbit DATA_1LINE = P0^1; //一线串口 DATA 数据端
```

```
/*-----*
```

```
;模块名称:Delay_100us();
```

```
;功 能:延时函数 STC10F04 100us @ 11.0592MHz
```

```
;入 参:unsigned int z 延时次数
```

```
;出 参:无
```

```
;-----*/
```

```
void Delay_100us(unsigned int z)
```

```
{
```

```
    unsigned int j;
```

```
for (z>0;z--)  
{  
    for (j=85;j>0;j--);  
}  
  
/*-----  
;模块名称:Delay_1ms();  
;功    能:延时函数  STC10F04 1ms @11.0592MHz  
;入    参:unsigned int z   延时次数  
;出    参:无  
-----*/  
  
void Delay_1ms(unsigned int z)  
{  
    unsigned int i,j;  
    for (i=z;i>0;i--)  
    {  
        for (j=850;j>0;j--);  
    }  
}
```

```
/*-----*
```

;模块名称: *SendAddr\_1LINE*

;功 能: *MSK010B* 一线串口控制方式的发送地址子函数

;入 参: *addr 16bit* 语音地址

;出 参: 无

```
-----*/
```

```
void SendAddr_1LINE(unsigned int addr) //发送一线地址的程序
```

```
{
```

```
unsigned char i,tmp;
```

```
RESET=0;
```

```
Delay_1ms(5);
```

```
RESET=1;
```

```
Delay_1ms(10);
```

```
DATA_1LINE=0;
```

```
Delay_1ms(5);
```

```
//发送 bit8~bit16
```

```
tmp = addr>>8;
```

```
for (i=0;i<8;i++)
```

```
{
```

```
DATA_1LINE=1;
```

```
if (tmp&0x01)
```

```
{
```

```
Delay_100us(6);
```

```
DATA_1LINE=0;
```

```
Delay_100us(2);
```

```
}
```

```
else
```

```
{
```

```
Delay_100us(2);
```

```
DATA_1LINE=0;
```

```
Delay_100us(6);
```

```
}
```

```
tmp>>=1;
```

```
}
```

```
DATA_1LINE=1;
```

```
Delay_1ms(1);
```

```
DATA_1LINE=0;
```

```
Delay_1ms(1);
```

```
//发送 bit0~bit7
```

```
tmp = addr;
```

```
for (i=0;i<8;i++)
```

```
{
```

```
    DATA_1LINE=1;
```

```
    if (tmp&0x01)
```

```
{
```

```
        Delay_100us(6);
```

```
        DATA_1LINE=0;
```

```
        Delay_100us(2);
```

```
}
```

```
else
```

```
{
```

```
        Delay_100us(2);
```

```
        DATA_1LINE=0;
```

```
        Delay_100us(6);
```

```
}
```

```
    tmp>>=1;
```

```
}
```

```
    DATA_1LINE=1;
```

```
}
```

```
/*-----*
```

;模块名称: *SendCmd\_1LINE*

;功 能: *MSK010B* 一线串口控制方式的发送命令子函数

;入 参: *cmd 8bit* 命令

;出 参: 无

```
-----*/
```

```
void SendCmd_1LINE(unsigned char cmd) //发送一线控制命令的程序
```

```
{
```

```
    unsigned char i;
```

```
    DATA_1LINE=0;
```

```
    Delay_1ms(5);
```

```
    for (i=0;i<8;i++)
```

```
{
```

```
    DATA_1LINE=1;
```

```
    if (cmd&0x01)
```

```
{
```

```
        Delay_100us(6);
```

```
        DATA_1LINE=0;
```

```
    Delay_100us(2);
```

```
}
```

```
else
```

```
{
```

```
    Delay_100us(2);
```

```
    DATA_1LINE=0;
```

```
    Delay_100us(6);
```

```
}
```

```
    cmd>>=1;
```

```
}
```

```
    DATA_1LINE=1;
```

```
}
```

```
/*-----*
```

```
;模块名称: Main
```

```
;功    能: 主函数
```

```
;入    参: void
```

```
;出    参: void
```

```
*-----*/
```

```
void Main(void)
```

{

```
//端口初始化
```

```
RESET = 1;
```

```
DATA_1LINE = 1;
```

```
Delay_1ms(100);
```

```
SendAddr_1LINE(0x0000); //播放 0000H 地址语音
```

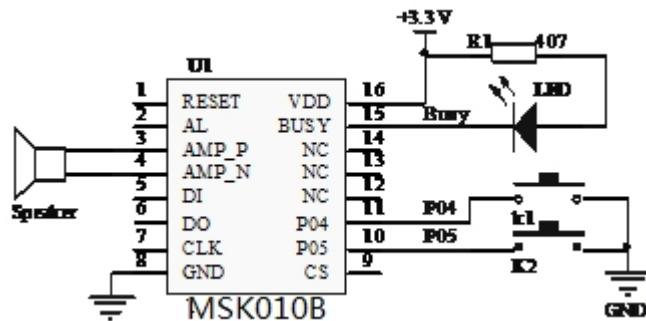
```
SendCmd_1LINE(0xf2); //发送循环播放命令
```

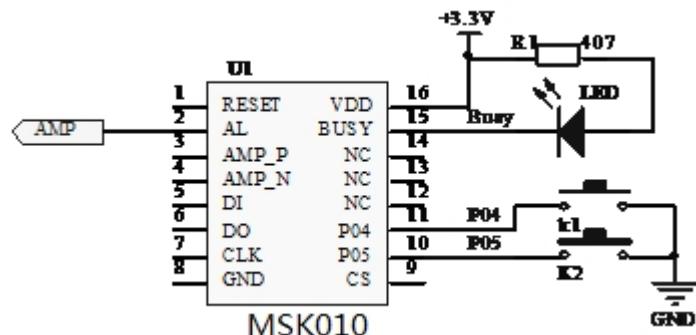
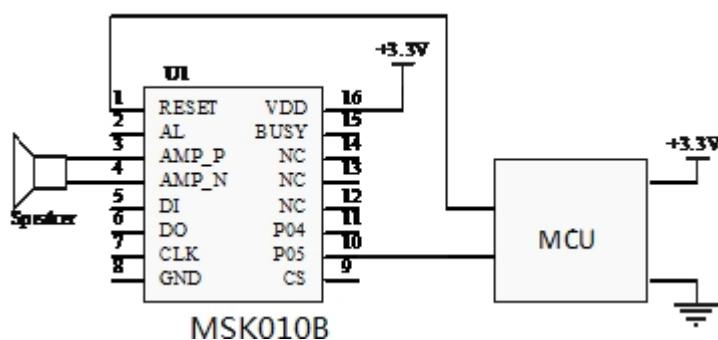
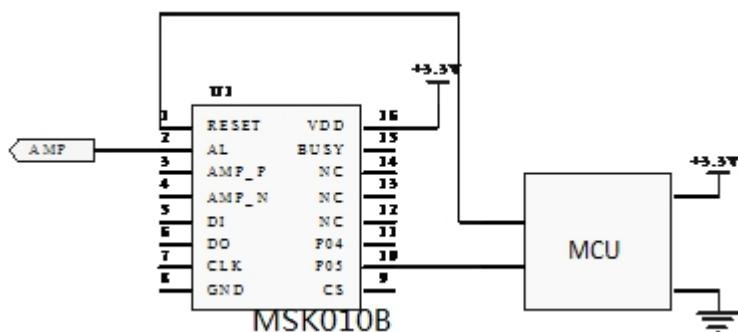
```
while (1);
```

}

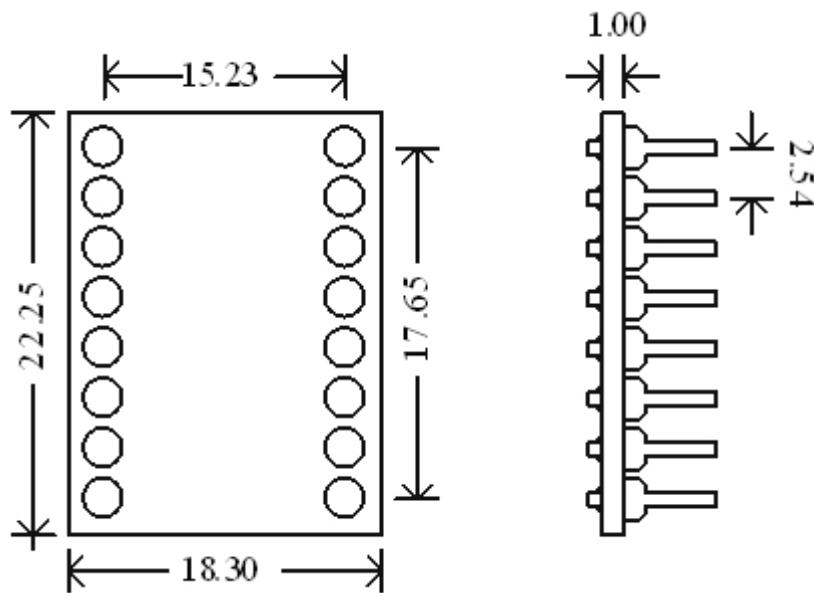
## 6、典型应用电路

### 6. 1、按键控制 PWM 输出应用电路



**6. 2、按键控制 DAC 输出应用电路**

**6. 3、一线串口 PWM 输出应用电路**

**6. 4、一线串口 DAC 输出应用电路**

**7、封装尺寸图**

单位: mm



## 8、历史版本记录

版本	日期	描述
V1. 0	2010-3-11	原始版本
V1. 1	2010-9-4	更新联系方式
V1. 2	2011-8-22	修正音量命令描述
V1. 3	2011-9-21	增加MCU控制参考程序范例
V1. 4	2012-09-25	修改了产品特征描述，去掉混音功能和插播功能 (该版本仅适用于MSK010B V1. 2模块)
V2. 1	2012-09-25	去掉三线串口控制方式，去掉两个I/O口，修改管脚图、应用电路 (该版本仅适用于MSK010B V2. 1模块)

深圳市迈思凯科技有限公司是一家集产品开发，生产，服务于一体的技术型企业，多年来一直专注于语音方案的开发，生产，销售，并承接电子产品的开发和小批量生产，成熟后采用 OEM 方式批量生产和供货。产品涵盖汽车、安防、电力、教学、医疗、小家电、通讯、玩具及礼品行业等领域。经过多年的技术及业务积累，涉及产品领域在不断扩大，市场也从最早的国内向东南亚、非洲，欧美等市场延伸。

Mysky 以打造“中华语音品牌”为目标，得益于此，Mysky 现已拥有一支经验丰富的技术开发团队，以及高素质的业务队伍。相信我们专业的技术服务，能得到您的认可。

请您与 Mysky 一起品味优质的科技生活!



Technology Co., Ltd. 迈思凯科技

适用于 MSK010B V2.1 模块使用资料

---

公司: 深圳市迈思凯科技有限公司

电话: 0755-29068287

传真: 0755-29165264

地址: 深圳市宝安区民治沙元埔大厦 502