

概述

ZLG72128 是广州致远电子股份有限公司自行设计的数码管显示驱动及键盘扫描管理芯片。一片 ZLG72128 能够直接驱动 12 位共阴式数码管（或 96 只独立的 LED），同时还可以扫描管理多达 32 只按键。其中有 8 只按键还可以作为功能键使用，就像电脑键盘上的 Ctrl、Shift、Alt 键一样。另外 ZLG72128 内部还设置有连击计数器，能够使某键按下后不松手而连续有效。通信采用 I²C 总线接口，与微控制器的接口仅需两根信号线。该芯片为工业级芯片，抗干扰能力强，在工业测控中已有大量应用。

产品特性

- ◆ 直接驱动 12 位共阴式数码管（1 英寸以下）或 96 只独立的 LED；
- ◆ 能够管理多达 32 只按键，自动消除抖动，其中有 8 只可以作为功能键使用；
- ◆ 利用功率电路可以方便地驱动 1 英寸以上的大型数码管；
- ◆ 具有位闪烁、位消隐、段点亮、段熄灭、功能键、连击键计数等强大功能；
- ◆ 软件配置支持 0~12 位数码管显示驱动模式；
- ◆ 提供有 10 种数字和 21 种字母的译码显示功能，或者直接向显示缓存写入显示数据；
- ◆ 与微控制器之间采用 I²C 串行总线接口，只需两根信号线，节省 I/O 资源。
- ◆ 提供器件一个地址设置端口，可设置两种不同的器件地址，一对 I²C 总线即可挂接两片 ZLG72128 芯片。

产品应用

- ◆ 仪器仪表
- ◆ 工业控制器
- ◆ 条形显示器
- ◆ 控制面板

订购信息

型号	温度范围	封装
ZLG72128	-40℃ ~ +85℃	TSSOP28

修订历史

版本	日期	原因
1.0.00	2016/11/08	创建文档
1.0.01	2017/01/16	按键中断和键值读取添加详细说明
1.0.02	2017/02/13	I ² C 通信关于字节传输添加详细说明
2.0.00	2018/07/01	关于器件地址设置端口 A4 添加详细说明
2.0.01	2019/01/17	重新排版整理手册结构框架
2.0.02	2019/05/09	完善典型电路图的复位电路
2.0.03	2019/06/24	修稿文档模板具体编码
2.0.04	2019/09/12	更新模板
2.0.05	2020/12/17	更新 Logo 模板

目 录

1. 引脚信息.....	1
1.1 引脚图	1
1.2 引脚功能描述.....	1
2. 功能详解.....	3
2.1 功能概述.....	3
2.2 寄存器详解	3
2.2.1 系统寄存器 SystemReg（地址：00H）	3
2.2.2 键值寄存器 Key（地址：01H）	3
2.2.3 连击计数器 RepeatCnt（地址：02H）	3
2.2.4 功能键寄存器 FunctionKey（地址：03H）	3
2.2.5 命令缓冲区 CmdBuf0 和 CmdBuf1（地址：07H 和 08H）	4
2.2.6 闪烁控制寄存器 FlashOnOff（地址：0BH）	4
2.2.7 消隐寄存器 DispCtrl0(地址：0CH)和 DispCtrl1(地址：0DH)	4
2.2.8 闪烁寄存器 Flash0(地址：0EH)和 Flash1(地址：0FH)	4
2.2.9 显示缓冲区 DispBuf0~DispBuf11（地址：10H~1BH）	4
2.3 控制命令详解.....	5
2.3.1 段寻址（SegOnOff）	5
2.3.2 下载数据并译码（Download）	5
2.3.3 复位命令(Reset).....	6
2.3.4 测试命令(Test).....	6
2.3.5 左移命令(ShiftLeft).....	6
2.3.6 循环左移命令(CyclicShiftLeft)	7
2.3.7 右移命令(ShiftRight).....	7
2.3.8 循环右移命令(CyclicShiftRight).....	7
2.3.9 数码管扫描位数设置命令（Scanning）	7
3. 时序接口.....	9
3.1 I ² C 数据传输时序.....	9
3.1.1 起动（START）和停止（STOP）条件.....	9
3.1.2 位传输	9
3.1.3 应答位	9
3.1.4 字节传输.....	10
3.2 ZLG72128 I ² C 数据传输.....	10
4. 应用电路.....	12
4.1 典型应用电路.....	12
4.2 两片 ZLG72128 器件地址设置电路	12
5. 电气参数.....	14
6. 封装尺寸.....	15
7. 免责声明.....	16

1. 引脚信息

1.1 引脚图

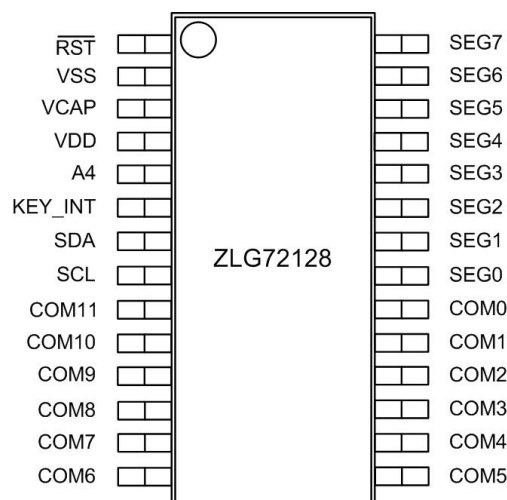


图 1.1 ZLG72128 引脚图

1.2 引脚功能描述

表 1.1 ZLG72128 引脚功能描述

引脚序号_28	引脚名称	功能描述
1	RST	复位信号，低电平有效
2	GND	接地
3	VCAP	外置电容端口
4	VDD	电源 3.0~5.5V
5	A4	器件地址设置端口
6	KEY_INT	键盘中断输出，低电平有效
7	SDA	I ² C 总线数据信号
8	SCL	I ² C 总线时钟信号
9	COM11/KR3	数码管位选信号 11/键盘行信号 3
10	COM10/KR2	数码管位选信号 10/键盘行信号 2
11	COM9/KR1	数码管位选信号 9/键盘行信号 1
12	COM8/KR0	数码管位选信号 8/键盘行信号 0
13	COM7/KC7	数码管位选信号 7/键盘列信号 7
14	COM6/KC6	数码管位选信号 6/键盘列信号 6
15	COM5/KC5	数码管位选信号 5/键盘列信号 5
16	COM4/KC4	数码管位选信号 4/键盘列信号 4
17	COM3/KC3	数码管位选信号 3/键盘列信号 3
18	COM2/KC2	数码管位选信号 2/键盘列信号 2
19	COM1/KC1	数码管位选信号 1/键盘列信号 1
20	COM0/KC0	数码管位选信号 0/键盘列信号 0
21	SEG0	数码管 a 段

续上表

引脚序号_28	引脚名称	功能描述
22	SEG1	数码管 b 段
23	SEG2	数码管 c 段
24	SEG3	数码管 d 段
25	SEG4	数码管 e 段
26	SEG5	数码管 f 段
27	SEG6	数码管 g 段
28	SEG7	数码管 dp 段

2. 功能详解

2.1 功能概述

一片 ZLG72128 可以扫描管理多达 32 个按键，K1~K24 为普通按键，F0~F7 为功能键。普通按键有连击检测功能。

ZLG72128 内部有 12 个显示缓冲寄存器 DispBuf0~DispBuf11，它们直接决定数码管显示的内容。ZLG72128 提供有两种显示控制方式，一种是直接向显存写入字型数据，另一种是通过向命令缓冲寄存器写入控制指令实现自动译码显示。

访问这些寄存器需要通过 I²C 总线接口来实现。ZLG72128 具有一个器件地址设置引脚 A4，当 A4 上拉或悬空时，ZLG72128 的 I²C 总线器件地址是 60H（写操作）和 61H（读操作）；当 A4 接地时，ZLG72128 的 I²C 总线器件地址是 40H（写操作）和 41H（读操作）。访问内部寄存器要通过“子地址”来实现。

2.2 寄存器详解

2.2.1 系统寄存器 SystemReg（地址：00H）

系统寄存器的第 0 位（LSB）称作 KeyAvi，标志着按键是否有效，0—无按键信号，1—有按键信号。SystemReg 寄存器的其它位暂时没有定义。当按下某个按键或者功能键按下抬起时，ZLG72128 的 INT 引脚会由高电平跳变成低电平，KeyAvi 置为 1。当读走系统寄存器值之后，中断信号就会自动撤销（变为高电平），同时 KeyAvi 也将变为 0。正常情况下，有按键按下微控制器只需要判断 INT 引脚就可以了。通过不断查询 KeyAvi 位也能判断是否有键按下，这样就可以节省微控制器的一根 I/O 口线，但是 I²C 总线处于频繁的活动状态，多消耗电流并且不利于抗干扰。

2.2.2 键值寄存器 Key（地址：01H）

如果某个普通键（图 4.1 中的 K1~K24）被按下，则微控制器可以从键值寄存器 Key 中读取相应的键值 1~24。如果微控制器发现 ZLG72128 的 INT 引脚产生了中断请求，而从 Key 中读到的键值是 0，则表示按下的可能是功能键。键值寄存器 Key 的值在被读走后自动变成 0。

2.2.3 连击计数器 RepeatCnt（地址：02H）

ZLG72128 为普通键（图 4.1 中的 K1~K24）提供了连击计数功能。所谓连击是指按住某个普通键不松手，经过两秒钟的延迟后，开始连续有效，连续有效间隔时间约两百毫秒。这一特性跟电脑上的键盘很类似。在微控制器能够及时响应按键中断并及时读取键值的前提下，当按住某个普通键一直不松手时：首先会产生一次中断信号，这时连击计数器 RepeatCnt 的值仍然是 0；经过一秒延迟后，会连续产生中断信号，每中断一次 RepeatCnt 就自动加 1；当 RepeatCnt 计数到 255 时就不再增加，而中断信号继续有效。在此期间，键值寄存器的值每次都会产生。

2.2.4 功能键寄存器 FunctionKey（地址：03H）

ZLG72128 提供有 8 个功能键（图 4.1 中的 F0~F7）。功能键常常是配合普通键一起使用的，就像电脑键盘上的 Shift、Ctrl 和 Alt 键。当然功能键也可以单独去使用，就像电脑键盘上的 F1~F12。当按下某个功能键时，在 INT 引脚也会像按普通键那样产生中断信号。功能键的键值是被保存在 FunctionKey 寄存器中的。功能键寄存器 FunctionKey 的初始值是

FFH, 每一个位对应一个功能键, 第 0 位 (LSB) 对应 F0, 第 1 位对应 F1, 依次类推, 第 7 位 (MSB) 对应 F7。某一功能键被按下时, 相应的 FunctionKey 位就清零。功能键还有一个特性就是“二次中断”, 按下时产生一次中断信号, 抬起时又会产生一次中断信号; 而普通键只会在被按下时产生一次中断。

2.2.5 命令缓冲区 CmdBuf0 和 CmdBuf1 (地址: 07H 和 08H)

通过向命令缓冲区写入相关的控制命令可以实现段寻址、下载显示数据功能。详见第 2.3 节。

2.2.6 闪烁控制寄存器 FlashOnOff (地址: 0BH)

FlashOnOff 寄存器决定闪烁频率和占空比。复位值为 0111,0111B。高 4 位表示闪烁时亮的持续时间, 低 4 位表示闪烁时灭的持续时间。改变 FlashOnOff 的值, 可以同时改变闪烁频率和占空比。FlashOnOff 取值 00H 时可获得最快的闪烁速度。

亮和灭的时间计算公式如下:

$$T = N \times 50 + 150\text{ms}$$

T 为闪烁时亮或灭的持续时间, N 为寄存器的高 4 位或者低 4 位的值, 取值 0~15。最快闪烁频率为 3.33Hz (周期 300ms), 最慢闪烁频率为 0.55Hz (周期 1.8S)。

特别说明: 单独设置 FlashOnOff 寄存器的值, 并不会看到显示闪烁, 而应该配合闪烁控制命令一起使用。

2.2.7 消隐寄存器 DispCtrl0(地址: 0CH)和 DispCtrl1(地址: 0DH)

DispCtrl0 (0CH)								DispCtrl1 (0DH)							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	B11	B10	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0

DispCtrl0、DispCtrl1 寄存器决定哪些位是否显示, 对应数码管的 1~12 位。寄存器位为 1 时, 对应数码管位不显示。复位值都是 0x00, 即数码管的 12 个位都扫描显示。实际应用中可能需要显示的位数不足 12 位, 例如只显示 8 位, 这时可以把 DispCtrl0 的值设置为 0x0F, 把 DispCtrl1 的值设置为 0x00, 则数码管的第 1~8 位被扫描显示, 而第 9~12 位不会显示。

2.2.8 闪烁寄存器 Flash0(地址: 0EH)和 Flash1(地址: 0FH)

Flash0 (0EH)								Flash1 (0FH)							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	B11	B10	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0

Flash0、Flash 1 寄存器决定哪些位是否闪烁, 对应数码管的 1~12 位。寄存器位为 1 时, 对应数码管位闪烁。复位值都是 0x00, 即数码管的 12 个位都不闪烁。实际应用中可能需要某些位闪烁, 例如最后 2 位闪烁, 这时可以把 Flash0 的值设置为 0x00, 把 Flash1 的值设置为 0x03, 数码管的第 1、2 位闪烁, 而第 3~12 位不会闪烁。

2.2.9 显示缓冲区 DispBuf0~DispBuf11 (地址: 10H~1BH)

DispBuf0~DispBuf11 这 12 个寄存器的取值直接决定了数码管的显示内容。每个寄存器的 8 个位分别对应数码管的 a,b,c,d,e,f,g,dp 段。

DispBuf_n(10H~1BH)							
D7	D6	D5	D4	D3	D2	D1	D0
Dp	g	f	e	d	c	b	a

例如大写字母 H 的字型数据为 76H（不带小数点）或 F6H（带小数点）。

2.3 控制命令详解

寄存器 CmdBuf0（地址：07H）和 CmdBuf1（地址：08H）共同组成命令缓冲区。通过向命令缓冲区写入相关的控制命令可以实现段寻址、下载显示数据、控制闪烁等功能。

2.3.1 段寻址（SegOnOff）

CmdBuf0 (07H)								CmdBuf1 (08H)							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	1	0	0	0	on	B3	B2	B1	B0	S3	S2	S1	S0

在段寻址命令中，12 位数码管(0~11)，每个数码管 8 个段，每一个段实际上就是一只独立的 LED。

双字节命令，在指令格式中，CmdBuf0 的高 4 位的“0001”是命令码；CmdBuf0 的最低位 on 位表示该段是否点亮，0—灭，1—亮。CmdBuf1 的 B3B2B1B0 是位地址，取值 0~11；S3S2S1S0 是 4 位段地址，取值 0~7，对应数码管的 a,b,c,d,e,f,g,dp 段。

2.3.2 下载数据并译码（Download）

CmdBuf0 (07H)								CmdBuf1 (08H)							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	0	A3	A2	A1	A0	dp	flash	0	d4	d3	d2	d1	d0

双字节命令，在指令格式中，CmdBuf0 的高 4 位的“0010”是命令码；A3A2A1A0 是数码管显示数据的位地址，位地址编号按从左到右的顺序依次为 0,1,2,3,……, 11；dp 控制小数点是否点亮，1 为点亮，0 为熄灭；flash 表示是否要闪烁，0—正常显示，1—闪烁；d4d3d2d1d0 是要显示的数据，包括 10 种数字和 21 种字母。显示数据按照表 2.1 中的规则进行译码：

表 2.1 下载数据并译码命令的数据表

d4d3d2d1d0（二进制）					d4d3d2d1d0（十六进制）		显示结果
0	0	0	0	0	00H		0
0	0	0	0	1	01H		1
0	0	0	1	0	02H		2
0	0	0	1	1	03H		3
0	0	1	0	0	04H		4
0	0	1	0	1	05H		5
0	0	1	1	0	06H		6
0	0	1	1	1	07H		7
0	1	0	0	0	08H		8
0	1	0	0	1	09H		9
0	1	0	1	0	0AH		A
0	1	0	1	1	0BH		b

续上表

d4d3d2d1d0（二进制）					d4d3d2d1d0（十六进制）	显示结果
0	1	1	0	0	0CH	C
0	1	1	0	1	0DH	d
0	1	1	1	0	0EH	E
0	1	1	1	1	0FH	F
1	0	0	0	0	10H	G
1	0	0	0	1	11H	H
1	0	0	1	0	12H	i
1	0	0	1	1	13H	J
1	0	1	0	0	14H	L
1	0	1	0	1	15H	o
1	0	1	1	0	16H	p
1	0	1	1	1	17H	q
1	1	0	0	0	18H	r
1	1	0	0	1	19H	t
1	1	0	1	0	1AH	U
1	1	0	1	1	1BH	y
1	1	1	0	0	1CH	c
1	1	1	0	1	1DH	h
1	1	1	1	0	1EH	T
1	1	1	1	1	1FH	（无显示）

2.3.3 复位命令(Reset)

CmdBuf0 (07H)							
D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	1	0	0	0	0

单字节命令，在指令格式中，CmdBuf0 的高 4 位的“0011”是命令码。功能是将所有 LED 熄灭。

2.3.4 测试命令(Test)

CmdBuf0 (07H)							
D7	D6	D5	D4	D3	D2	D1	D0
0	1	0	0	0	0	0	0

单字节命令，在指令格式中，CmdBuf0 的高 4 位的“0100”是命令码。功能是将所有 LED 按照 0.5S 的速率闪烁。

2.3.5 左移命令(ShiftLeft)

CmdBuf0 (07H)							
D7	D6	D5	D4	D3	D2	D1	D0
0	1	0	1	b3	b2	b1	b0

单字节命令，在指令格式中，CmdBuf0 的高 4 位的“0101”是命令码。按照数码管位为

单位，左移 n 位。左移后右边空出的位不显示任何内容，即全部 LED 熄灭状态。 n 的取值范围 1~11，大于 11 的值无效， n 的值由 CmdBuf0 的低 4 位决定，按下列公式计算。

$$n = (b3 \times 8) + (b2 \times 4) + (b1 \times 2) + b0$$

2.3.6 循环左移命令(CyclicShiftLeft)

CmdBuf0 (07H)							
D7	D6	D5	D4	D3	D2	D1	D0
0	1	1	0	b3	b2	b1	b0

单字节命令，在指令格式中，CmdBuf0 的高 4 位的“0110”是命令码。功能是按照数码管位为单位，循环左移 n 位。左移后右边显示从最左边移出的内容。 n 的取值范围 1~11，大于 11 的值无效， n 的值由 CmdBuf0 的低 4 位决定，按下列公式计算。

$$n = (b3 \times 8) + (b2 \times 4) + (b1 \times 2) + b0$$

2.3.7 右移命令(ShiftRight)

CmdBuf0 (07H)							
D7	D6	D5	D4	D3	D2	D1	D0
0	1	1	1	b3	b2	b1	b0

单字节命令，在指令格式中，CmdBuf0 的高 4 位的“0111”是命令码。功能是按照数码管位为单位，右移 n 位。右移后左边空出的位不显示任何内容，即全部 LED 熄灭状态。 n 的取值范围 1~11，大于 11 的值无效， n 的值由 CmdBuf0 的低 4 位决定，按下列公式计算。

$$n = (b3 \times 8) + (b2 \times 4) + (b1 \times 2) + b0$$

2.3.8 循环右移命令(CyclicShiftRight)

CmdBuf0 (07H)							
D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	b3	b2	b1	b0

单字节命令，在指令格式中，CmdBuf0 的高 4 位的“1000”是命令码。功能是按照数码管位为单位，循环右移 n 位。右移后左边显示从最右边移出的内容。 n 的取值范围 1~11，大于 11 的值无效， n 的值由 CmdBuf0 的低 4 位决定，按下列公式计算。

$$n = (b3 \times 8) + (b2 \times 4) + (b1 \times 2) + b0$$

2.3.9 数码管扫描位数设置命令 (Scanning)

CmdBuf0 (07H)							
D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	1	b3	b2	b1	b0

单字节命令，在指令格式中，CmdBuf0 的高 4 位的“1001”是命令码。功能是设置数码管扫描位数 n 。 n 的取值是 0~12，大于 12 按 12 位进行扫描。扫描位数 n 以位选端第 1 位开始到位选端第 n 位扫描有效。 n 的值由 CmdBuf0 的低四位决定，按下列公式计算。

$$n = (b3 \times 8) + (b2 \times 4) + (b1 \times 2) + b0$$

在使用过程中，如果不需要 12 位数码管显示，从最高位开始裁剪，同时将数码扫描位

数设置成相应的数码管位数。

3. 时序接口

ZLG72128 的通信接口是 I²C 总线，I²C 总线用两条线（SDA 和 SCL）使芯片与芯片间传递信息。SDA 为串行数据线，SCL 为串行时钟线。这两根线引脚都是漏极开漏输出结构，在实际使用中两条线都必须通过上拉电阻（R_p, Pull-Up Resistor）与电源相连。上拉电阻一般取值为 3~10KΩ，开漏结构的好处是：当总线空闲时，这两条信号线都保持高电平，几乎不消耗电流；电气兼容性好，上拉电阻接 5V 电源就能与 5V 逻辑器件接口，上拉电阻接 3V 电源又能与 3V 逻辑器件接口；因为是开漏结构，所以不同器件的 SDA 与 SDA 之间、SCL 与 SCL 之间可以直接相连，不需要额外的转换电路。

3.1 I²C 数据传输时序

3.1.1 起动（START）和停止（STOP）条件

总线空闲时，数据线和时钟线保持高电平状态。当数据线在下降沿而时钟线为高电平时为起动信号，数据线在上升沿而时钟线为高电平时为停止信号。

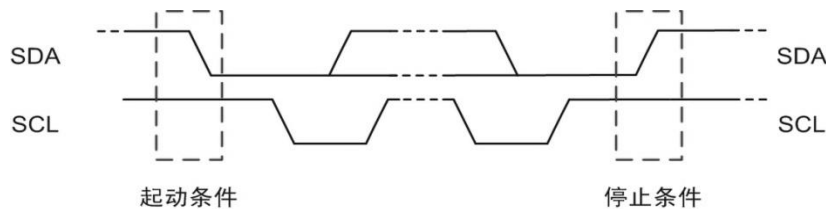


图 3.1 I²C 起动和停止条件示意图

3.1.2 位传输

每个时钟脉冲传输一个数据位，SDA 线上的数据在时钟脉冲为高电平时应保持稳定，也就是这时候的数据是有效的，否则 SDA 线上的数据将变成起动或者停止控制信号了。在时钟脉冲为低电平时允许 SDA 线上的数据变换。

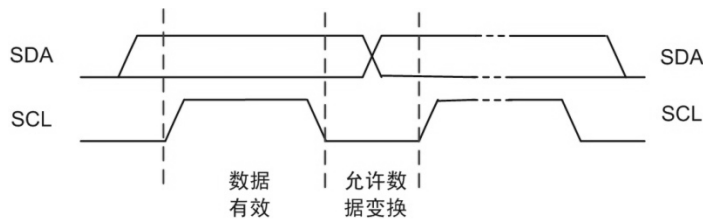
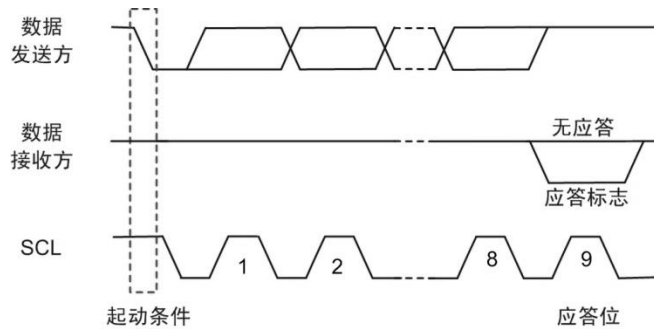


图 3.2 I²C 总线数据有效性示意图

3.1.3 应答位

在起动和停止条件之间数据传输的数量是没有限制的，每 8 位字节后加一个应答位，当然在应答位期间主设备需要附加一个时钟脉冲，当这个应答位时钟脉冲出现时，接收数据方使 SDA 数据线保持低电平，表示应答成功，这时主设备可产生停止条件。

图 3.3 I²C 总线数据接收方应答标识示意图

3.1.4 字节传输

I²C 总线以字节为单位收发数据，传输到 SDA 数据线上的数据必须是 8 位，每次传输的字节数量是不受限制的。数据在传输过程中，高位在前，低位在后。每个字节之后都会跟上一个应答响应位。不过注意一下，在接收数据方接收完最后一个字节数据，或者不能再接收更多数据时，应当产生非应答信号来通知发送数据方。发送数据方发现接收数据方产生的非应答信号，则应当终止发送。

在每个字节传输完成之后，为了保持主机和从机的同步，需要在主从机之间有一个握手过程。这个握手过程使从机在接受和响应完后一个字节后会使得 SCL 时钟线保持低电平，从而迫使主机进入等待状态，直到从机准备好下一个要传输的字节，才会释放 SCL 时钟线，数据传输得以继续。

ZLG72128 进行 I²C 通信时，硬件 I²C 会主动来完成数据传输过程。使用模拟 I²C 进行通信时，传输完一个字节数据响应后，主机需要判断 SCL 时钟线是否释放为高或者延时 15μs，才能继续数据传输。

3.2 ZLG72128 I²C 数据传输

在使用 I²C 总线传递数据时，ZLG72128 将作为一个从设备，这时时钟信号线 SCL 只能是输入信号线，数据信号线 SDA 是一条双向信号线。ZLG72128 的器件地址是一个 7 位地址，利用 A4 引脚可分别设置为两种不一样的器件地址。当 A4 上拉或悬空时，ZLG72128 的器件地址是 0x30；当 A4 接地时，ZLG72128 的器件地址是 0x20。利用 A4 引脚可设置两个不同的 7 位器件地址，这样一对 I²C 总线就能挂接两片 ZLG72128。所以在进行 I²C 通讯时，需要根据 A4 端口的电平状态来确认 ZLG72128 的器件地址。在启动 I²C 通信之后，器件地址与读写命令位组成一个字节进行传输，器件地址占用第一个字节的高 7 位，读写命令占最低位。当读写位为 0，表示主机将向从机写入数据，读写位为 1，表示主机将从从机读取数据。图 3.4 是从机地址与读写命令组合示意图。



图 3.4 从机地址与读写命令组合示意图

图 3.5 是主机向从机发送数据的示意图，I²C 启动后，主机首先向从机发送从机地址和写标志位，从机与自身地址匹配一致，在第 9 个时钟信号到来时，发送一个应答信号。然后主机再发送一个需要访问的从机设备子地址（I²C 设备子地址都是 8 位）的起始地址，从机应答之后，就可以发送数据，每发送完一字节数据，都需要从机进行应答。当主机向从机发

送最后一个数据时，从机可能应答或者不应答，主机都可以停止数据传输。

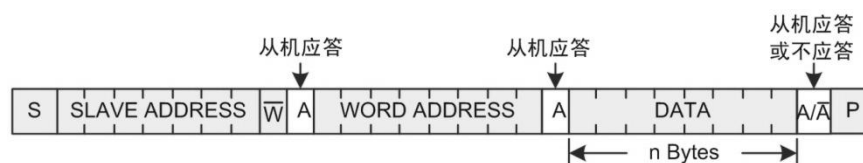


图 3.5 主机向从机发送数据示意图

图 3.6 是主机从从机接收数据的示意图， I^2C 启动后，主机首先向从机发送从机地址和写标志位，从机与自身地址匹配一致，在第 9 个时钟信号到来时，发送一个应答信号，然后主机再发送一个需要访问的从机设备子地址起始地址，从机应答之后，需要重复起始条件，再发送一字节带有从机地址的读命令，从机应答后，就可以从从机读取数据，每次主机读取到一字节数据，就发出应答信号。在主机读取到最后一个数据时，不需要从机应答，也表明数据不再传输，主机发送停止命令停止数据传输。

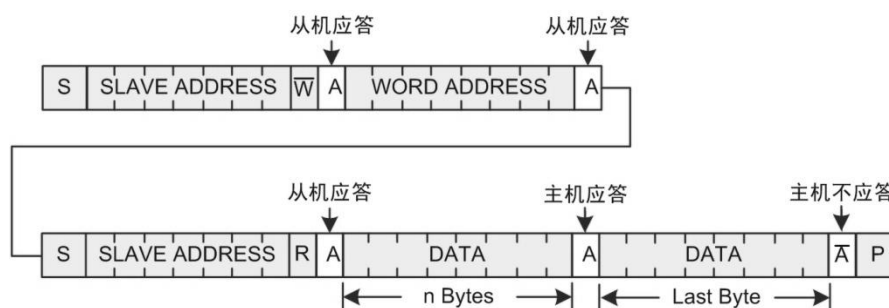


图 3.6 主机从从机接收数据示意图

4. 应用电路

4.1 典型应用电路

图 4.1 是 ZLG72128 的典型应用电路原理图。用户在使用芯片驱动数码管以及管理按键时可参考该电路进行电路设计。

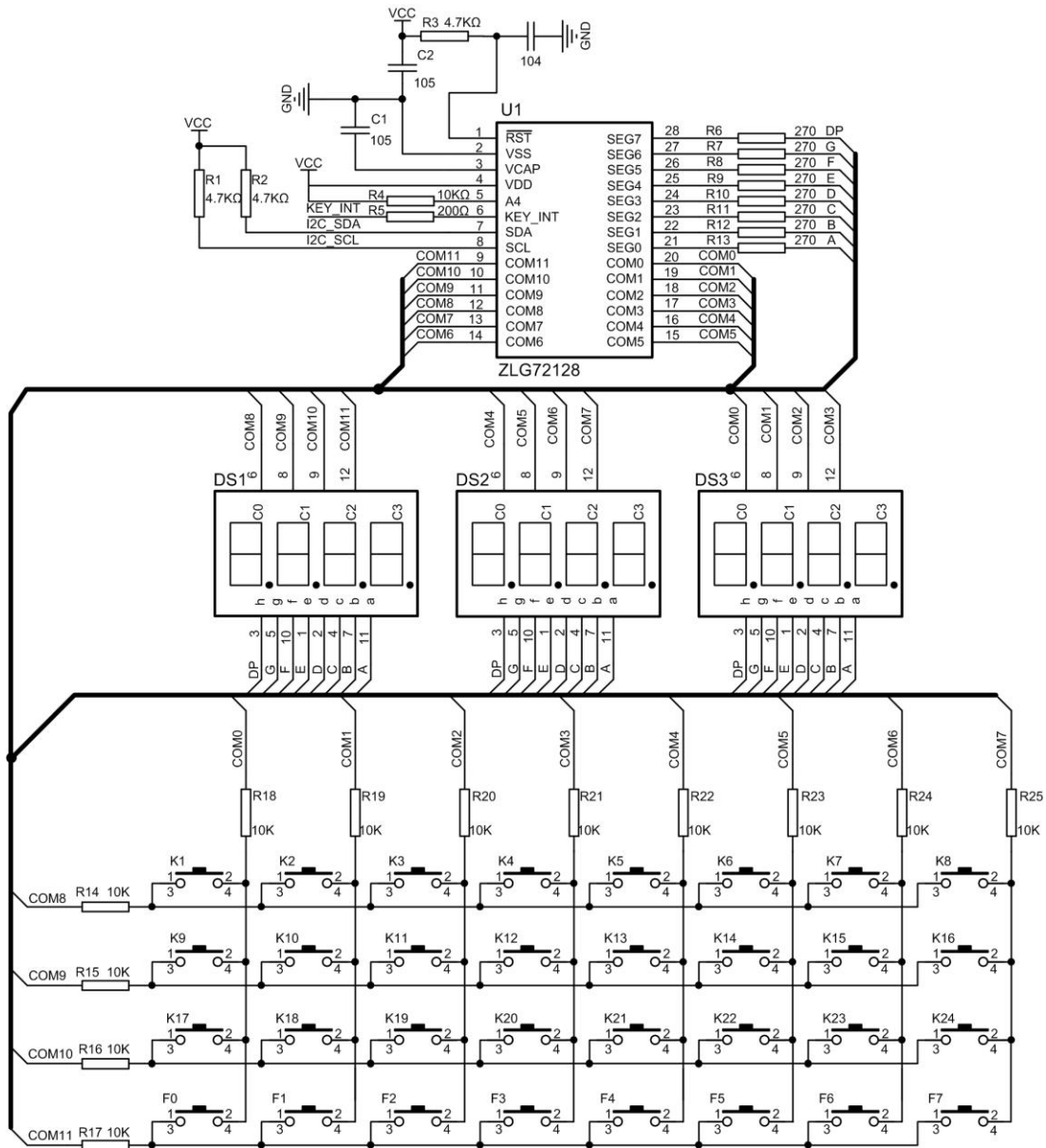
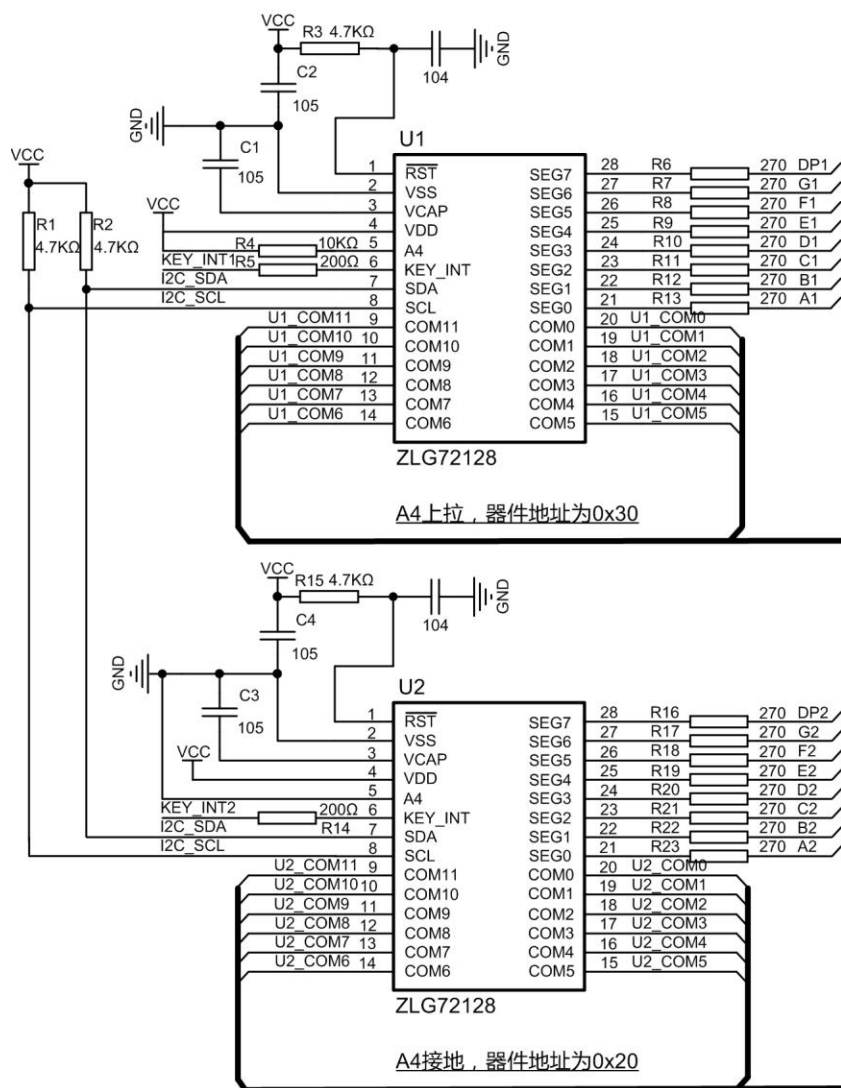


图 4.1 ZLG72128 典型应用电路（A4 上拉，器件地址为 0x30）

4.2 两片 ZLG72128 器件地址设置电路

图 4.2 是两片 ZLG72128 芯片挂接在一对 I²C 总线的电路原理图。ZLG72128 有一个器件地址设置引脚 A4，利用 A4 引脚状态将 ZLG72128 设置成不同的器件地址。图 2.1 中，一片 ZLG72128 的 A4 引脚通过 10 KΩ 上拉电阻拉高（或悬空），把 7 位器件地址设置 0x30；另一片 ZLG72128 的 A4 引脚接地，把 7 位器件地址设置为 0x20，这样一对 I²C 总线就能挂接两片 ZLG72128 芯片，从而可以驱动 24 位 8 段数码显示（或 192 个独立 LED）以及 64



为了使 ZLG72128 芯片电源稳定,一般在 VCC 和 GND 之间接入一个 47~470 μF 的电解电容。按照 I²C 总线协议的要求,信号线 SCL 和 SDA 上必须分别接上拉电阻,其典型值是 4.7K Ω 。当通信速率大于 100kbps 时,建议减小上拉电阻的值。芯片复位引脚 RST 是低电平有效,可以将其接入到 MCU 的 IO 来控制其复位。KEY_INT 引脚可输出按键中断请求信号(低电平有效),可以连接到 MCU 的 IO 来获取按键按下或松开事件。

5. 电气参数

表 5.1 电气特性 (VCC = 5.0V, TA = 25°C)

符号	参数	测试条件	最小	典型	最大	单位
V _{CC}	电源电压		3.0	5	5.5	V
I _{CC}	工作电流	LED 都不亮		2	5	mA
I _{CC}	工作电流	LED 全点亮		40	100	mA
I _{IO}	引脚输出电流				20	mA
V _{IH}	逻辑输入高电平		2.1		5.8	V
V _{IL}	逻辑输入低电平		-0.3		1.65	V
V _{OH}	逻辑输出高电平	V _{CC} = 5V I _{CC} = 10mA	2.8			V
V _{OL}	逻辑输出低电平	V _{CC} = 5V I _{CC} = 10mA			2.0	V
T _{KEY}	按键响应时间	含去抖时间		10		ms
T _{SCLL}	SCL 低电平时间		1.3			μs
T _{SCLH}	SCL 高电平时间		0.6			μs
T _{SUSDA}	SDA 建立时间		100			ns
T _{HSDA}	SDA 数据保持时间				900	ns
T _{STA}	启动时间		0.6			μs
T _{RSTA}	重复启动时间		0.6			μs
T _{RST}	芯片复位启动时间		5			ms

6. 封装尺寸

ZLG72128 采用的是标准的 TSSOP28 封装，如图 6.1 是其参数说明：

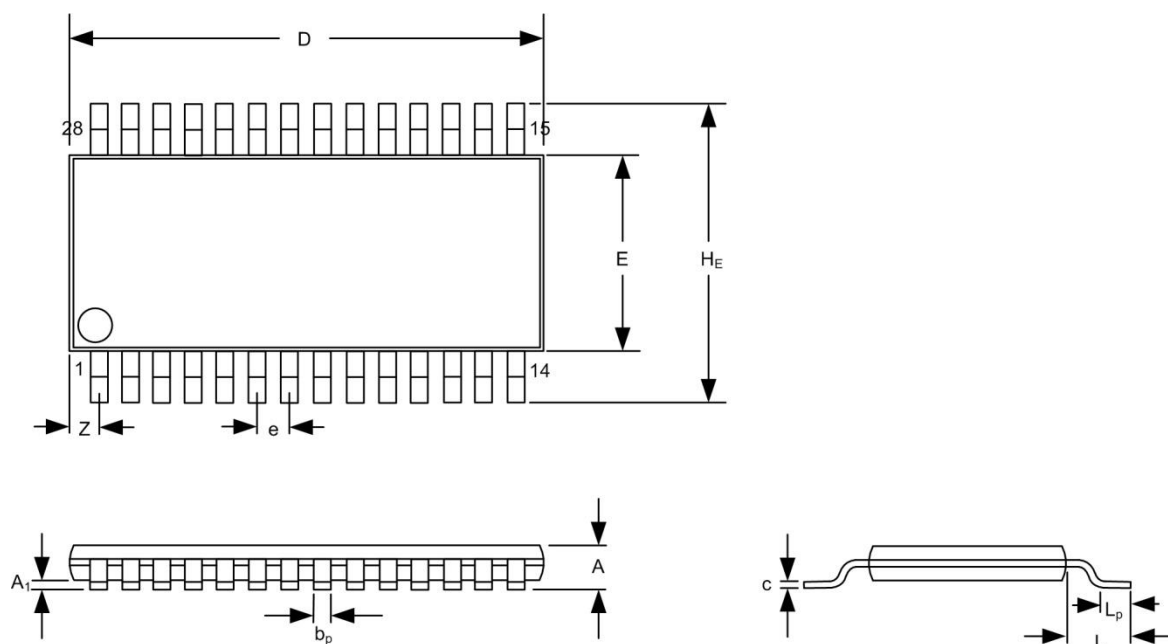


图 6.1 封装尺寸图

单位	D	H _E	E	Z	e	A	A ₁	b _p	c	L _p	L
mm	9.8	6.6	4.5	0.8	0.65	1.1	0.15	0.30	0.2	0.75	1
	9.6	6.2	4.3	0.5	0.65	1.1	0.05	0.19	0.1	0.50	1

7. 免责声明

本着为用户提供更好服务的原则，广州致远微电子有限公司（下称“致远微电子”）在本手册中将尽可能地为用户呈现详实、准确的产品信息。但鉴于本手册的内容具有一定的时效性，致远微电子不能完全保证该文档在任何时段的时效性与适用性。致远微电子有权在没有通知的情况下对本手册上的内容进行更新，恕不另行通知。为了得到最新版本的信息，请尊敬的用户定时访问官方网站或者与致远微电子工作人员联系。感谢您的包容与支持！

诚信共赢，持续学习，客户为先，专业专注，只做第一

广州致远电子有限公司

更多详情请访问
www.zlg.cn

欢迎拨打全国服务热线
400-888-4005



©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.