

一、概述

TM1638N是一种带键盘扫描接口的LED（发光二极管显示器）驱动控制专用IC, 内部集成有MCU 数字接口、数据锁存器、LED驱动、键盘扫描等电路。本产品质量可靠、稳定性好、抗干扰能力强。主要适用于家电设备(智能热水器、微波炉、洗衣机、空调、电磁炉)、机顶盒、电子称、智能电表等数码管或LED显示设备。

二、特性说明

- 采用CMOS 工艺
- 显示模式 10 段×8 位
- 键扫描（8×3bit）
- 辉度调节电路（占空比 8 级可调）
- 串行接口（CLK, STB, DIO）
- 振荡方式：RC 振荡
- 内置上电复位电路
- 封装形式：SOP28

三、管脚定义：

K1	<input type="checkbox"/>	1	28	<input type="checkbox"/>	STB
K2	<input type="checkbox"/>	2	27	<input type="checkbox"/>	CLK
K3	<input type="checkbox"/>	3	26	<input type="checkbox"/>	DIO
VCC	<input type="checkbox"/>	4	25	<input type="checkbox"/>	GND
SEG1/KS1	<input type="checkbox"/>	5	24	<input type="checkbox"/>	GRID1
SEG2/KS2	<input type="checkbox"/>	6	23	<input type="checkbox"/>	GRID2
SEG3/KS3	<input type="checkbox"/>	7	22	<input type="checkbox"/>	GRID3
SEG4/KS4	<input type="checkbox"/>	8	21	<input type="checkbox"/>	GRID4
SEG5/KS5	<input type="checkbox"/>	9	20	<input type="checkbox"/>	GRID5
SEG6/KS6	<input type="checkbox"/>	10	19	<input type="checkbox"/>	GRID6
SEG7/KS7	<input type="checkbox"/>	11	18	<input type="checkbox"/>	GND
SEG8/KS8	<input type="checkbox"/>	12	17	<input type="checkbox"/>	GRID7
SEG9	<input type="checkbox"/>	13	16	<input type="checkbox"/>	GRID8
SEG10	<input type="checkbox"/>	14	15	<input type="checkbox"/>	VDD

四、管脚功能说明：

符号	管脚名称	管脚号	说明
DIO	数据输出输入	26	在时钟上升沿输入串行数据, 从低位开始。在时钟下降沿输出串行数据, 从低位开始。输出时为N管开漏输出
CLK	时钟输入	27	在上升沿读取串行数据, 下降沿输出数据。
STB	片选输入	28	在下降沿初始化串行接口, 随后等待接收指令。STB为低后的第一个字节作为指令, 当处理指令时, 当前其它处理被终止。当STB为高时, CLK 被忽略。
K1~K3	键扫信号输入	1~3	输入该脚的数据在显示周期结束后被锁存
SGE1/KS1~ SEG8/KS8	输出 (段)	5~12	段输出 (也用作键扫描输出), P管开漏输出
GRID1~GRID8	输出 (位)	24~19 17~16	位输出, N管开漏输出
SEG9 ~SEG110	输出 (段)	13~14	段输出, P管开漏输出
VDD	逻辑电源	4、15	接电源正
GND	逻辑地	18、25	接系统地

▲ **注意：**DIO口输出数据时为N管开漏输出，在读键的时候需要外接 1K-10K的上拉电阻。本公司推荐 10K的上拉电阻。DIO在时钟的下降沿控制N管的动作，此时读数时不稳定，在时钟的上升沿读数时才稳定。

五、指令说明：

在STB下降沿后由DI0输入的第一个字节作为一条指令。经过译码，取最高B7、B6 两位比特位以区别不同的指令。

B7	B6	指令
0	1	数据命令设置
1	0	显示控制命令设置
1	1	地址命令设置

如果在指令或数据传输时STB被置为高电平，串行通讯被初始化，并且正在传送的指令或数据无效（之前传送的指令或数据保持有效）。

5. 1 数据命令设置

该指令用来设置数据写和读，B1 和B0 位不允许设置 01 或 11。

MSB

LSB

B7	B6	B5	B4	B3	B2	B1	B0	功能	说明
0	1	无关项， 填 0				0	0	数据读写模式 设置	写数据到显示寄存器
0	1					1	0		读键扫数据
0	1				0			地址增加模式 设置	自动地址增加
0	1				1				固定地址
0	1			0				测试模式设置 (内部使用)	普通模式
0	1			1					测试模式

5. 2 地址命令设置

MSB

LSB

B7	B6	B5	B4	B3	B2	B1	B0	显示地址
1	1	无关项， 填 0		0	0	0	0	00H
1	1			0	0	0	1	01H
1	1			0	0	1	0	02H
1	1			0	0	1	1	03H
1	1			0	1	0	0	04H
1	1			0	1	0	1	05H
1	1			0	1	1	0	06H
1	1			0	1	1	1	07H
1	1			1	0	0	0	08H
1	1			1	0	0	1	09H
1	1			1	0	1	0	0AH
1	1			1	0	1	1	0BH
1	1			1	1	0	0	0CH
1	1			1	1	0	1	0DH
1	1			1	1	1	0	0EH
1	1			1	1	1	1	0FH

该指令用来设置显示寄存器的地址。

如果地址设为 10H 或更高，数据被忽略，直到有效地址被设定。上电时，地址默认设为 00H。

5. 3 显示控制

MSB				LSB				功能	说明
B7	B6	B5	B4	B3	B2	B1	B0		
1	0	无关项， 填 0			0	0	0	消光数量设置	设置脉冲宽度为 1/16
1	0				0	0	1		设置脉冲宽度为 2/16
1	0				0	1	0		设置脉冲宽度为 4/16
1	0				0	1	1		设置脉冲宽度为 10/16
1	0				1	0	0		设置脉冲宽度为 11/16
1	0				1	0	1		设置脉冲宽度为 12/16
1	0				1	1	0		设置脉冲宽度为 13/16
1	0				1	1	1		设置脉冲宽度为 14/16
1	0			0				显示开关设置	显示关
1	0			1					显示开

六、显示寄存器地址：

该寄存器存储通过串行接口从外部器件传送到TM1638N的数据，地址从 00H-0FH 共 16 字节单元，分别与芯片 SEG 和 GRID 管脚所接的 LED 灯对应，分配如下图：

写 LED 显示数据的时候，按照从显示地址从低位到高位，从数据字节的低位到高位操作。

SEG1	SEG2	SEG3	SEG4	SEG5	SEG6	SEG7	SEG8	SEG9	SEG10	X	X	X	X	X	X	
xxHL（低四位）				xxHU(高四位)				xxHL(低四位)				xxHU（高四位）				
B0	B1	B2	B3	B4	B5	B6	B7	B0	B1	B2	B3	B4	B5	B6	B7	
00HL				00HU				01HL				01HU				GRID1
02HL				02HU				03HL				03HU				GRID2
04HL				04HU				05HL				05HU				GRID3
06HL				06HU				07HL				07HU				GRID4
08HL				08HU				09HL				09HU				GRID5
0AHL				0AHU				0BHL				0BHU				GRID6
0CHL				0CHU				0DHL				0DHU				GRID7
0EHL				0EHU				0FHL				0FHU				GRID8

七、显示

1、驱动共阴数码管：

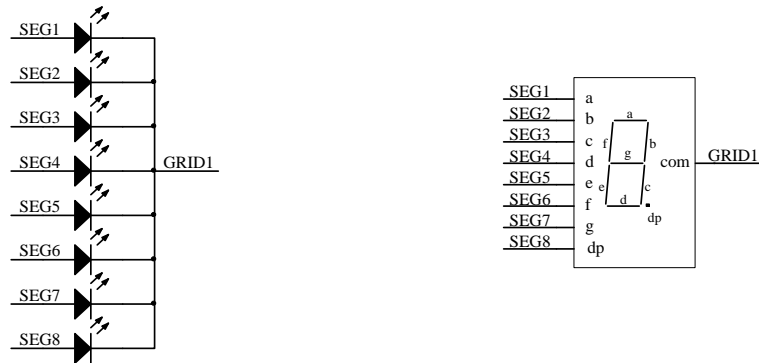


图 (7)

图 7 给出共阴极数码管的连接示意图, 如果让该数码管显示“0”, 只需要向 00H (GRID1) 地址中从低位开始写入 0x3F 数据即可, 此时 00H 对应每一个 SEG1~SEG8 的数据如下表格。

SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	
0	0	1	1	1	1	1	1	GRID1 (00H)
B7	B6	B5	B4	B3	B2	B1	B0	

2、驱动共阳数码管：

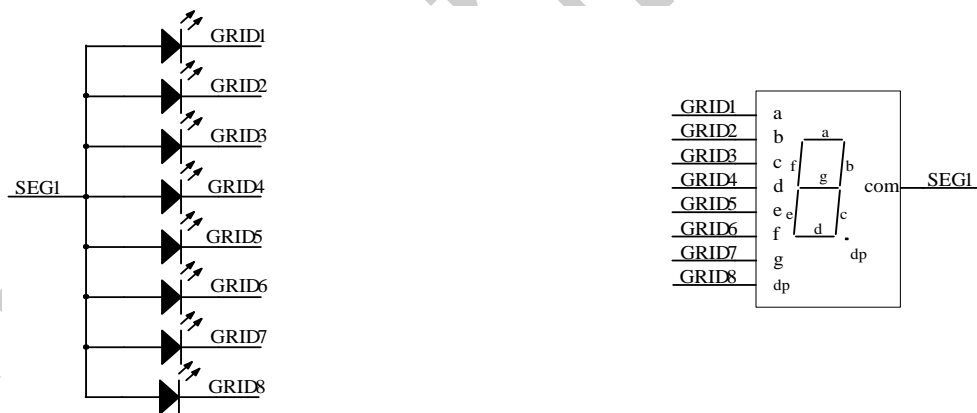


图 (8)

图 8 给出共阳极数码管的连接示意图, 如果让该数码管显示“0”, 需要向地址单元 00H (GRID1)、02H (GRID2)、04H (GRID3)、06H (GRID4)、08H (GRID5)、0AH (GRID6) 里面分别写数据 01H, 其余的地址 0CH (GRID7)、0EH (GRID8) 单元全部写数据 00H。每一个 SEG1~SEG8 对应的数据如下表格。

SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	
0	0	0	0	0	0	0	1	GRID1 (00H)
0	0	0	0	0	0	0	1	GRID2 (02H)
0	0	0	0	0	0	0	1	GRID3 (04H)
0	0	0	0	0	0	0	1	GRID4 (06H)
0	0	0	0	0	0	0	1	GRID5 (08H)

0	0	0	0	0	0	0	1	GRID6 (0AH)
0	0	0	0	0	0	0	0	GRID7 (0CH)
0	0	0	0	0	0	0	0	GRID8 (0EH)
B7	B6	B5	B4	B3	B2	B1	B0	

▲注意：无论是驱动共阴极数码管还是驱动共阳极数码管, SEG引脚只能接LED的阳极, GRID只能接LED的阴极, 不可反接。

八、 键扫描和按键复用：

键扫描矩阵为3×8bit, 如图 (3) 所示：

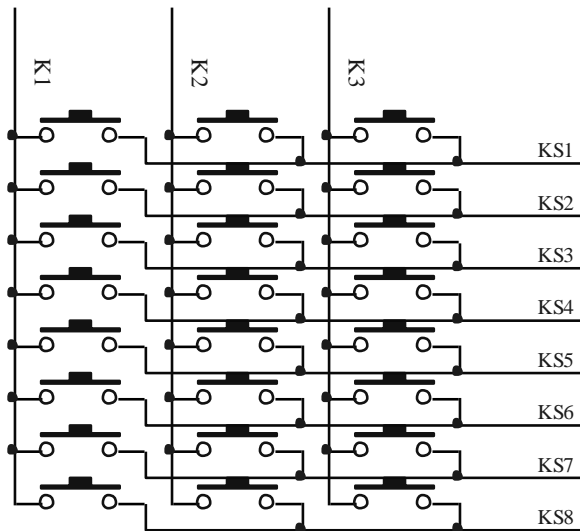


图 (3)

键扫描数据储存地址如图 (4) 所示, 先发读键命令后, 开始读取按键数据BYTE1—BYTE4 字节, 读数据从低位开始输出, 芯片K和KS引脚对应的按键按下时, 相对应的字节内的 BIT位为 1。

B0	B1	B2	B3	B4	B5	B6	B7	
K3	K2	K1	X	K3	K2	K1	X	
KS1				KS2				BYTE1
KS3				KS4				BYTE2
KS5				KS6				BYTE3
KS7				KS8				BYTE4

按键扫描与组合按键:

(1) **按键扫描:** 按键扫描由TM1638N自动完成, 不受用户控制, 用户只需要按照时序读键值。完成一次键扫描需要1个显示周期, 一个显示周期大概需要 $T=4.7\text{ms}$, 在 4.7ms 内先后按下了2个不同的按键, 2次读到的键值都是先按下的那个按键的键值。

(2) 组合按键

复合按键的问题异常: SEG1/KS1-SEG8/KS8 是显示和按键扫描复用的。以图(12)为例, 显示需要D1 亮, D2 灭, 需要让SEG1 为“0”, SEG2 为“1”状态, 如果S1, S2 同时被按下, 相当于SEG1, SEG2 被短路, 这时D1, D2 都被点亮。

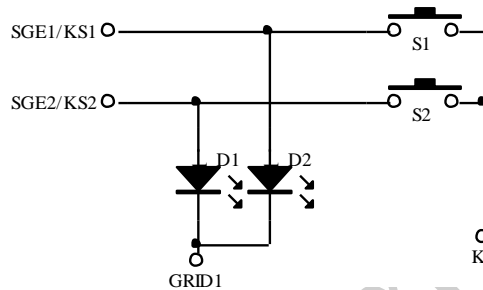


图 (12)

解决方案:

1、在硬件上, 可以将需要同时按下的键设置在不同的K线上面如图(13)所示,

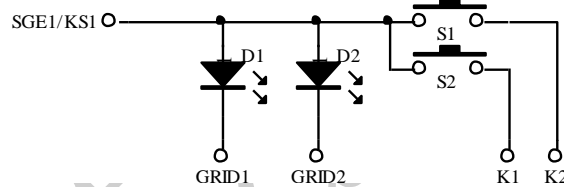


图 (13)

2、串联二极管如图(14)所示。

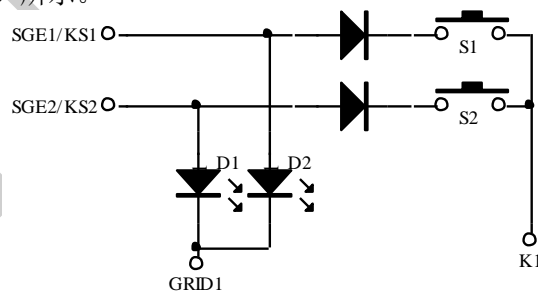
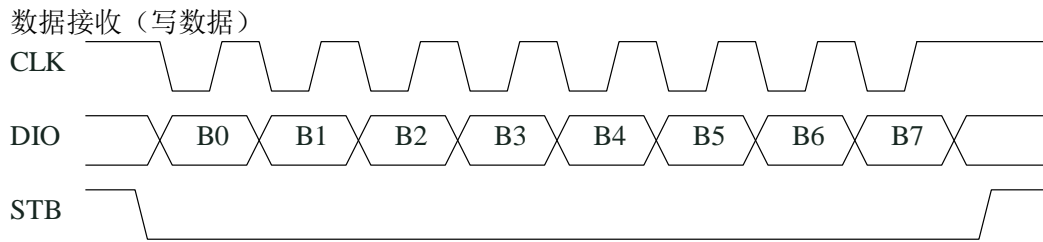


图 (14)

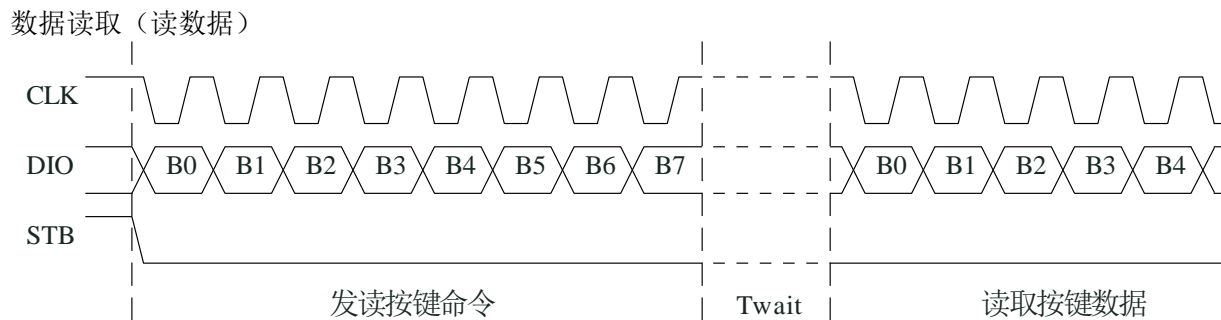
▲**注意:** 建议使用同一个KS不同的K键作为复合按键。

九、串行数据传输格式：

读取和接收 1 个BIT都在时钟的上升沿操作。



图（5）



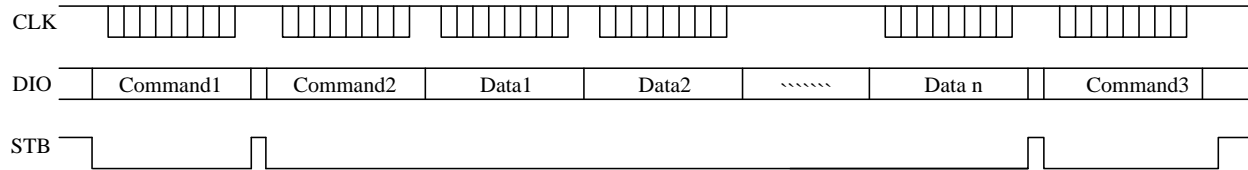
图（6）

▲注意：1、读取数据时，从串行时钟CLK的第 8 个上升沿开始设置指令到CLK下降沿读数据之间需要一个等待时间Twait(最小 2 μ S)。具体参数见时序特性表。

十、应用时串行数据的传输：

(1) 地址增加模式

使用地址自动加 1 模式, 设置地址实际上是设置传送的数据流存放的起始地址。起始地址命令字发送完毕, “STB” 不需要置高紧跟着传数据, 最多 16BYTE, 数据传送完毕才将 “STB” 置高。



Command1: 设置数据命令

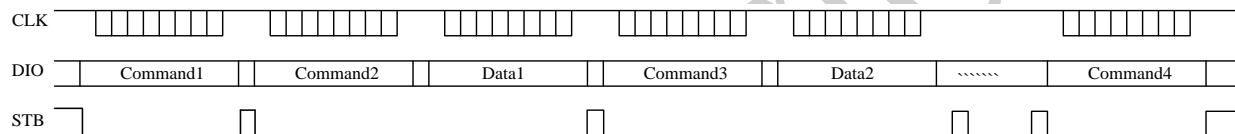
Command2: 设置显示地址

Data1~n: 传输显示数据至Command2 地址和后面的地址内 (最多 16bytes)

Command3: 显示控制命令

(2) 固定地址模式

使用固定地址模式, 设置地址实际上是设置需要传送的 1BYTE 数据存放的地址。地址发送完毕, “STB” 不需要置高, 紧跟着传 1BYTE 数据, 数据传送完毕才将 “STB” 置高。然后重新设置第 2 个数据需要存放的地址, 最多 16BYTE 数据传送完毕, “STB” 置高。



Command1: 设置数据命令

Command2: 设置显示地址 1

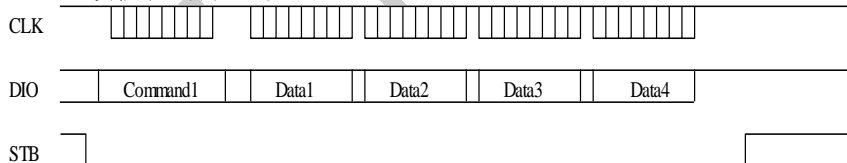
Data1: 传输显示数据 1 至Command2 地址内

Command3: 设置显示地址 2

Data2: 传输显示数据 2 至Command3 地址内

Command4: 显示控制命令

(3) 读按键时序

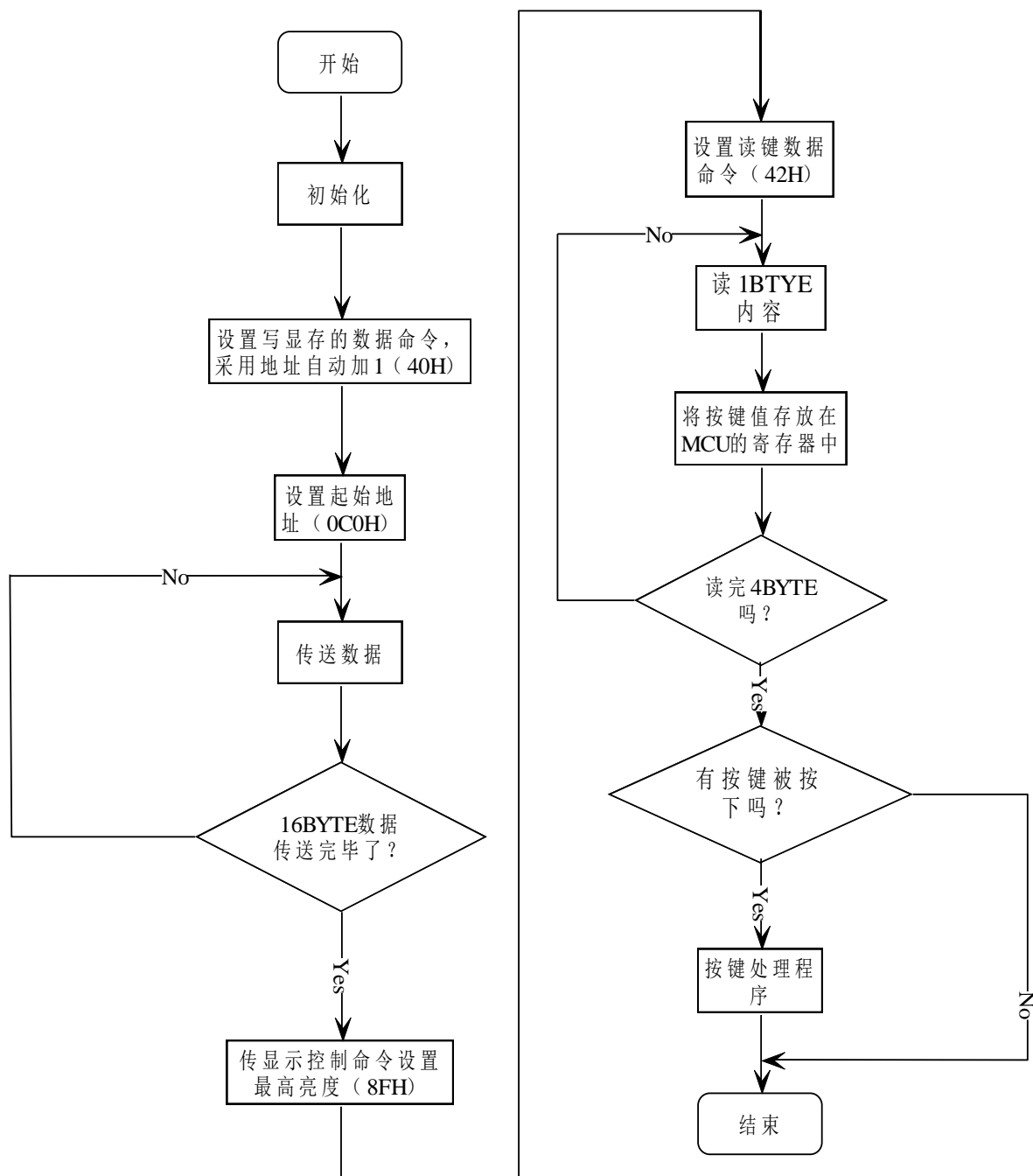


Command1: 设置读按键命令

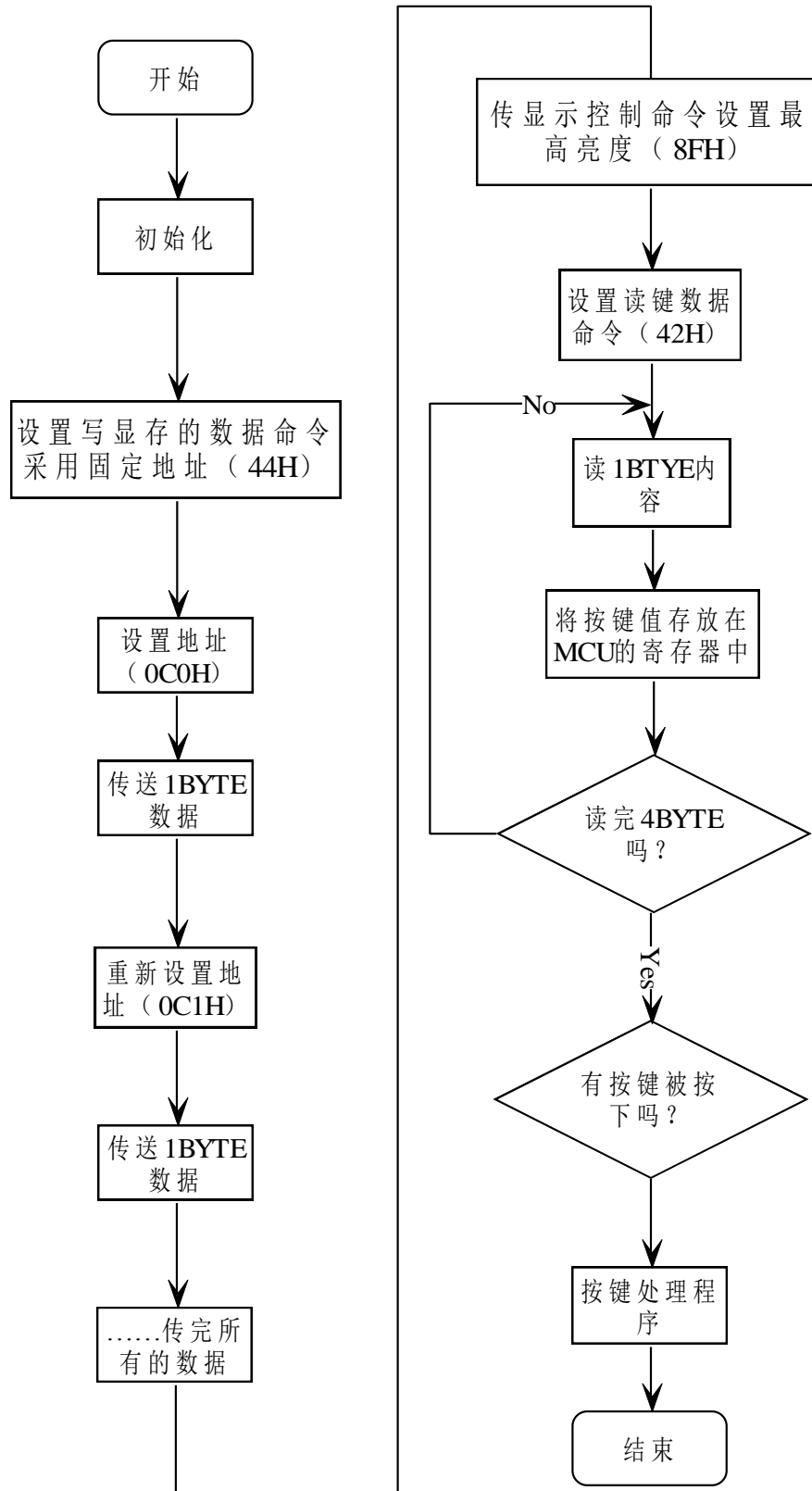
Data1~4: 读取按键数据

(4) 采用地址自动加一和固定地址方式的程序设计流程图:

采用地址自动加一程序设计流程图:

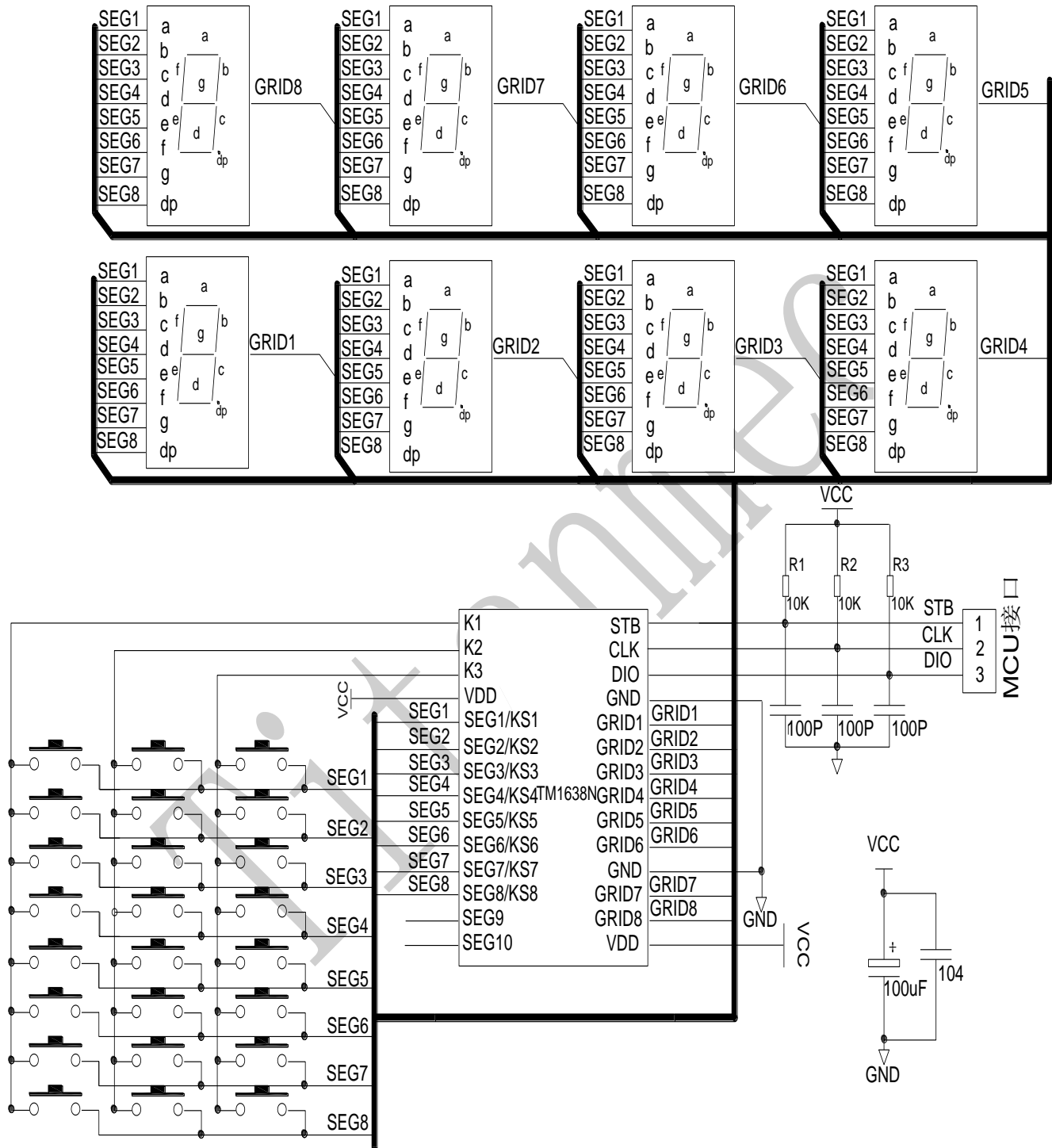


采用固定地址的程序设计流程图：

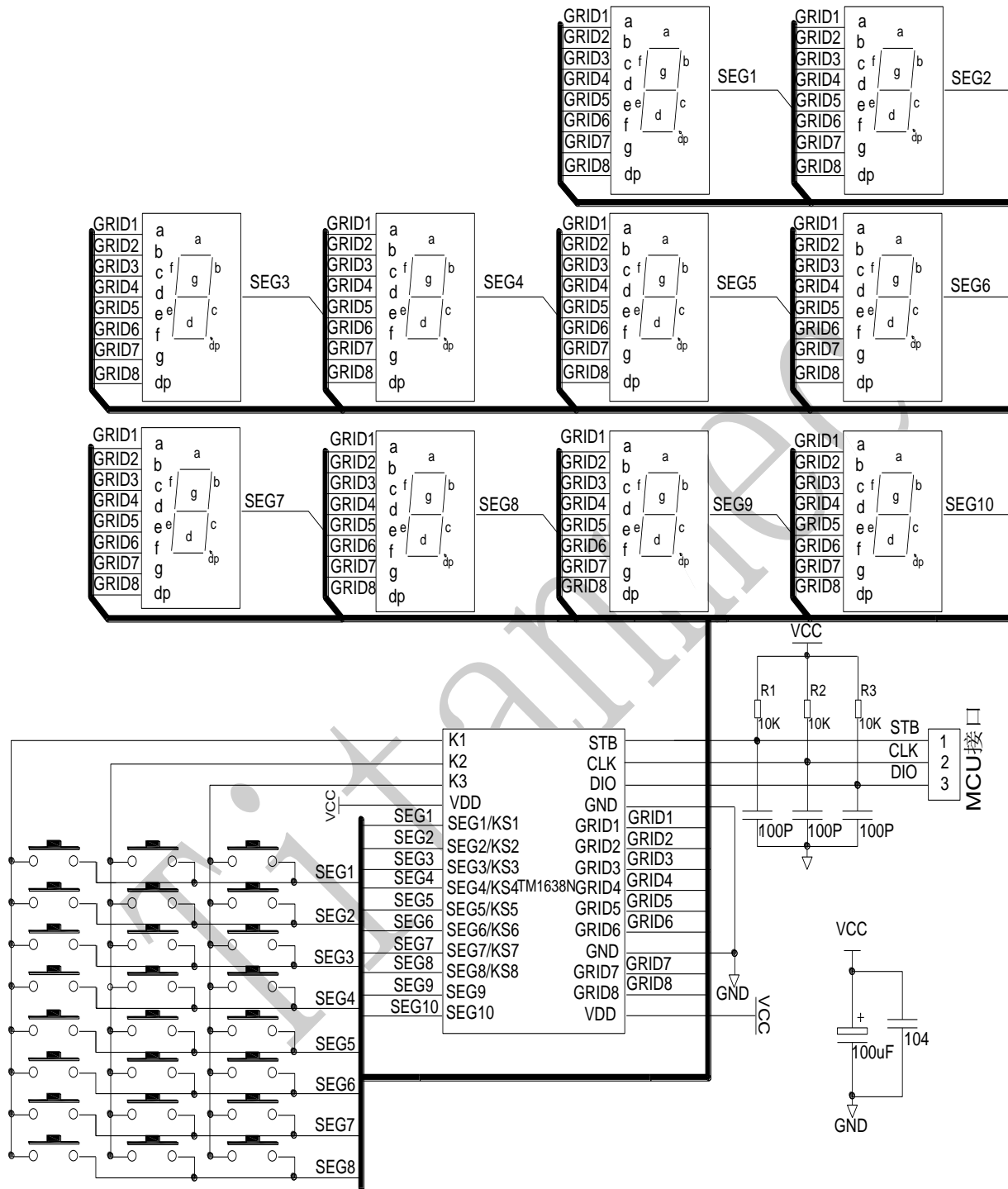


十一. 应用电路:

TM1638N驱动共阴数码屏硬件电路图



TM1638N驱动共阳数码屏硬件电路图



- ▲注意:
- 1、VDD、GND之间滤波电容在PCB板布线应尽量靠近TM1638N芯片放置，加强滤波效果。
 - 2、连接在DIO、CLK、STB通讯口上三个 100P电容可以降低对通讯口的干扰。
 - 3、因蓝光数码管的导通压降约为 3V，因此TM1638N供电应选用 5V。

十二、 电气参数:

极限参数 ($V_{SS} = 0\text{ V}$)

参数	符号	范围	单位
逻辑电源电压	VDD	-0.5 ~ +7.0	V
逻辑输入电压	VI1	-0.5 ~ VDD + 0.5	V
LED Seg 驱动输出电流	IO1	-50	mA
LED Grid 驱动输出电流	IO2	+200	mA
功率损耗	PD	400	mW
工作温度	Topt	-40 ~ +85	°C
储存温度	Tstg	-65 ~ +150	°C

正常工作范围 ($V_{SS} = 0\text{ V}$)

参数	符号	最小	典型	最大	单位	测试条件
逻辑电源电压	VDD		5		V	-
高电平输入电压	VIH	0.7 VDD	-	VDD	V	-
低电平输入电压	VIL	0	-	0.3 VDD	V	-

电气特性 ($VDD = 4.5 \sim 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

参数	符号	最小	典型	最大	单位	测试条件
SEG驱动拉电流	Ioh1	20	25	40	mA	SGE1~SEG10 $V_O = VDD - 2V$
	Ioh2	20	30	50	mA	SGE1~SEG10 $V_O = VDD - 3V$
GRID驱动灌电流	IOL1	80	140	-	mA	GRID1~GRID8 $V_O = 0.3V$

输出下拉电阻	RL		10		K Ω	K1~K3
输入电流	II	—	—	± 1	μA	VI = VDD / VSS
高电平输入电压	VIH	0.7 VDD	—		V	CLK, DIO, STB
低电平输入电压	VIL	—	—	0.3 VDD	V	CLK, DIO, STB
滞后电压	VH	—	0.35	—	V	CLK, DIO, STB
动态电流损耗	IDDdyn	—	—	5	mA	无负载, 显示关

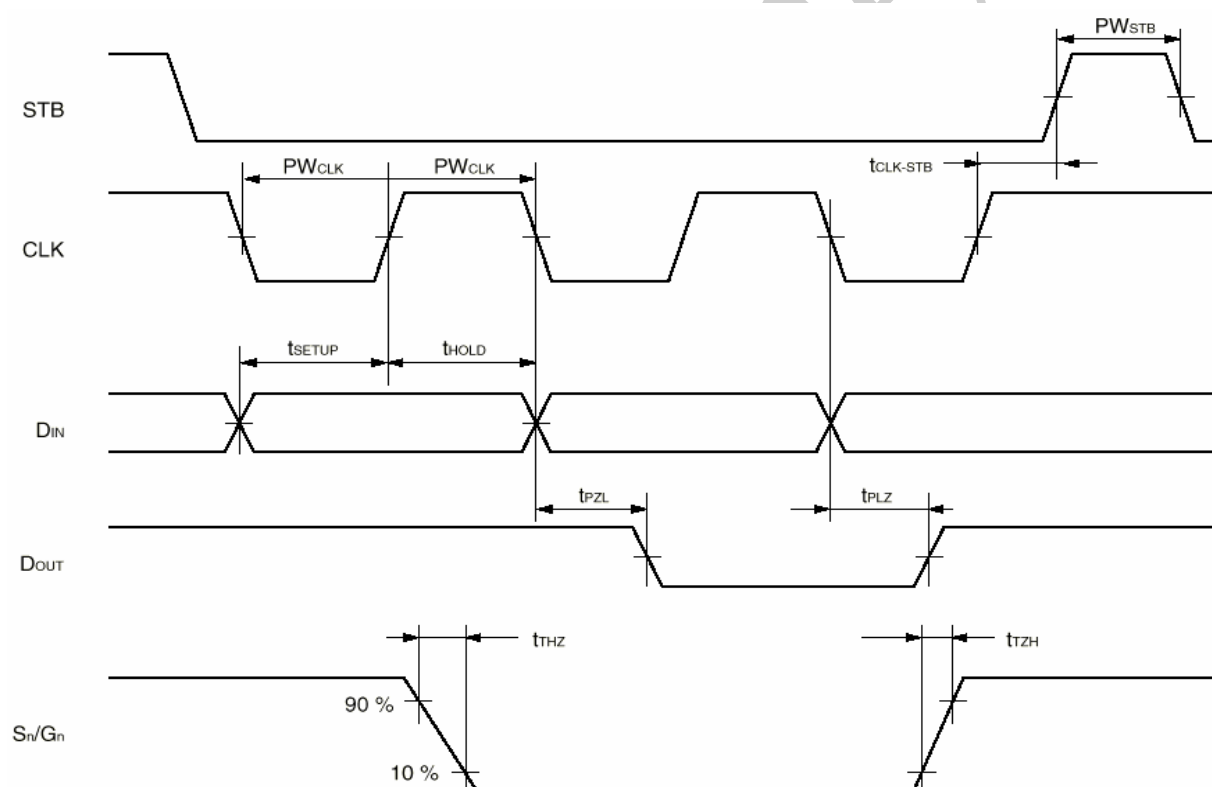
开关特性 (VDD = 4.5 ~ 5.5 V)

参数	符号	最小	典型	最大	单位	测试条件	
振荡频率	fosc	—	500	—	KHz	R = 16.5 K Ω	
传输延迟时间	tPLZ	—	—	300	ns	CLK \rightarrow DIO	
	tPZL	—	—	100	ns	CL = 15pF, RL = 10K Ω	
上升时间	TTZH 1	—	—	2	μs	CL = 300p F	SEG1~SEG10
下降时间	TTHZ	—	—	120	μs	CL = 300pF, SEGN, GRIDN	
最大时钟频率	Fmax	—	—	1	MHz	占空比 50%	
输入电容	CI	—	—	15	pF	—	

时序特性 (VDD = 4.5 ~ 5.5 V)

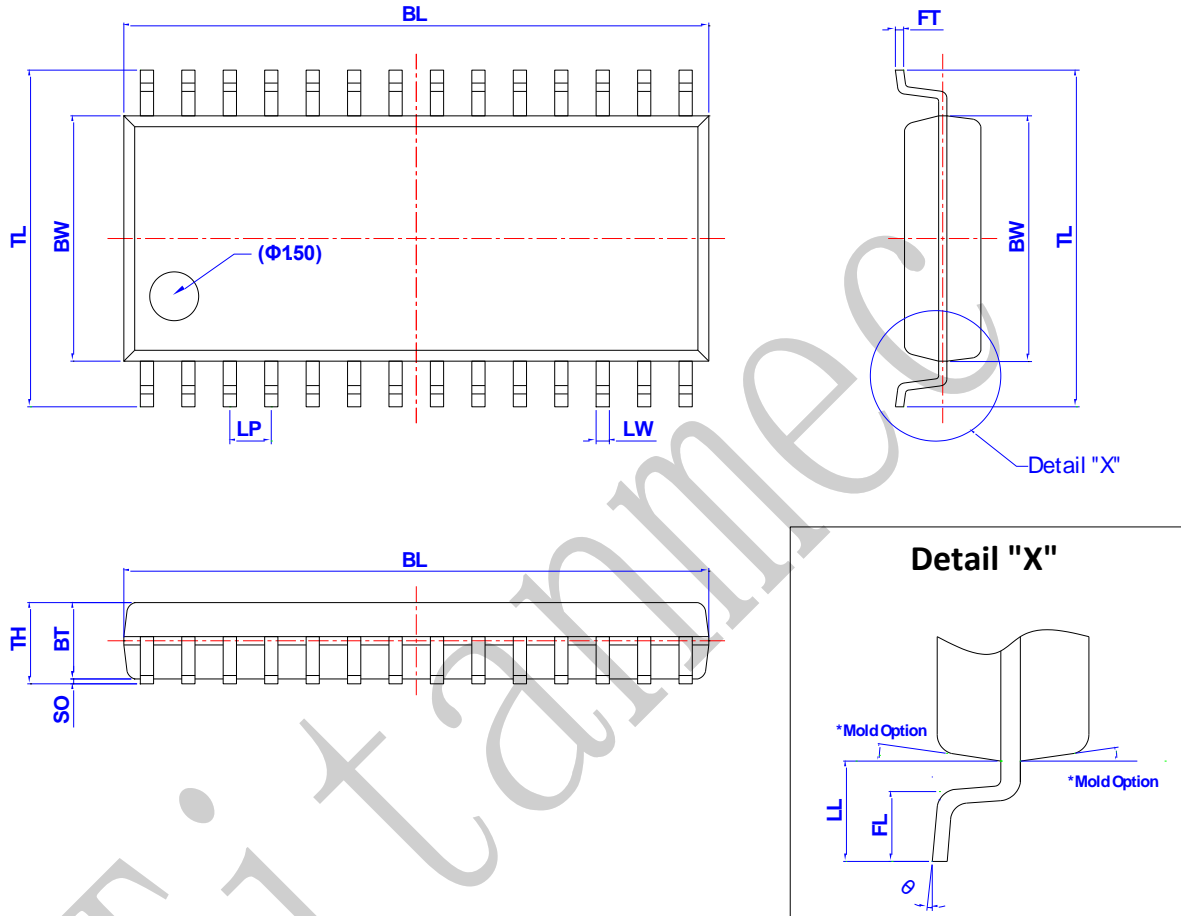
参数	符号	最小	典型	最大	单位	测试条件
时钟脉冲宽度	PWCLK	400	—	—	ns	—
选通脉冲宽度	PWSTB	1	—	—	μs	—
数据建立时间	t _{SETUP}	100	—	—	ns	—
数据保持时间	t _{HOLD}	100	—	—	ns	—
CLK → STB 时间	t _{CLK-STB}	1	—	—	μs	CLK ↑ → STB ↑
等待时间	t _{WAIT}	1	—	—	μs	CLK ↑ → CLK ↓

时序波形图:



十三、封装尺寸

SOP28 的封装尺寸:



Dimensions

Item	BL	BW	TL	LW	LP	FT	BT	SO	TH	LL	FL	Θ
表示	总长	胶体宽度	跨度	脚宽	脚间距	脚厚	胶体厚度	站高	胶体高度	单边长	脚长	脚角度
Unit	mm	mm	mm	mm	mm	mm	mm	mm	mm	mm	mm	°
Spec	18.03 (17.93) 17.83	7.62 (7.52) 7.42	10.56 (10.37) 10.21	0.406 TYP	1270 TYP	0.300 (0.250) 0.200	2.44 (2.34) 2.24	0.250 (0.150) 0.100	2.590 Max.	150 (140) 130	0.90 (0.80) 0.70	8 (4) 0

All specs and applications shown above subject to change without prior notice.
(以上电路及规格仅供参考, 如本公司进行修正, 恕不另行通知)