

# BC6088(A)

## 单线 UART 接口 88 键键盘矩阵接口芯片

### 特点

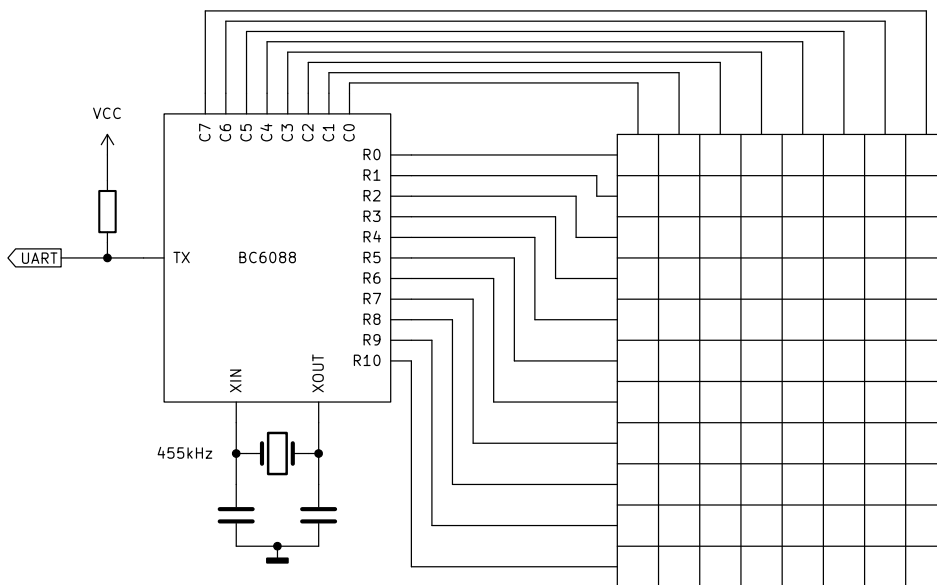
- 2.0~5.5V 工作电压
- 单线 UART 接口
- 支持常开/常闭按键
- 支持任意数量组合键
- 直接键值输出
- 支持任意键长按检测
- 72uA 工作电流(BC6088A@2.5V)
- SSOP24 封装
- 与 BC6301, BC6561 协议兼容

### 简介

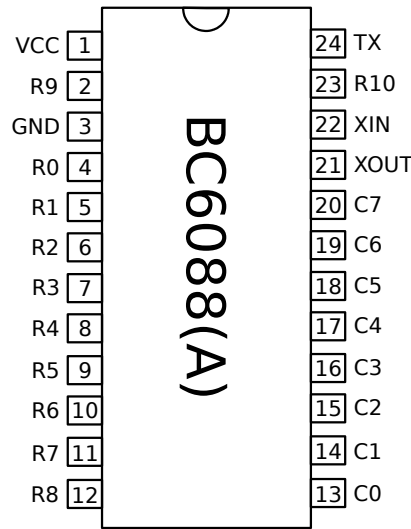
BC6088(A)是专用的键盘接口芯片，可以驱动 11×8 的 88 键键盘矩阵。可以适用于金属接触式或导电橡胶等各种类型的键盘，也可适用于机械式的拨动开关等。数据输出采用 UART 接口，接口电路得到简化，只需占用 1 根口线，并且可以配合如光耦、RS-485 等接口电路，方便实现电气隔离和远程键盘。

BC6088(A)直接输出键值，使得编程得以极大简化，用户程序只需根据 UART 接口所接收到的键值数据执行相应动作即可。同时 BC6088(A)可以支持任意数量的组合按键和常开常闭按键，使得应用更加灵活。配合定时器，可以实现用户所需任何时长的长按检测，包括对组合键的长按检测。

宽工作电压范围和低工作电流，使得 BC6088(A)可以和绝大多数 MCU 配合工作，并可用于电池供电设备。



# 引脚定义



# 引脚说明

名称	序号	功能
VCC	1	正电源引脚，电压范围 2.0~5.5V
GND	3	接地端
XIN	22	振荡器输入端，接 455kHz 谐振器或外部时钟源
XOUT	21	振荡器输出端，接 455kHz 谐振器或悬空
TX	24	UART 输出
R0~R10	2,4,5,6,7,8,9,10,11,12,23	键盘矩阵行接口，内部带下拉电阻
C0~C7	13,14,15,16,17,18,19,20	键盘矩阵列接口

# 键值对应码表

	C0	C1	C2	C3	C4	C5	C6	C7
R0	0(0x00)	11(0x0b)	22(0x16)	33(0x21)	44(0x2c)	55(0x37)	66(0x42)	77(0x4d)
R1	1(0x01)	12(0x0c)	23(0x17)	34(0x22)	45(0x2d)	56(0x38)	67(0x43)	78(0x4e)
R2	2(0x02)	13(0x0d)	24(0x18)	35(0x23)	46(0x2e)	57(0x39)	68(0x44)	79(0x4f)
R3	3(0x03)	14(0x0e)	25(0x19)	36(0x24)	47(0x2f)	58(0x3a)	69(0x45)	80(0x50)
R4	4(0x04)	15(0x0f)	26(0x1a)	37(0x25)	48(0x30)	59(0x3b)	70(0x46)	81(0x51)
R5	5(0x05)	16(0x10)	27(0x1b)	38(0x26)	49(0x31)	60(0x3c)	71(0x47)	82(0x52)

R6	6(0x06)	17(0x11)	28(0x1c)	39(0x27)	50(0x32)	61(0x3d)	72(0x48)	83(0x53)
R7	7(0x07)	18(0x12)	29(0x1d)	40(0x28)	51(0x33)	62(0x3e)	73(0x49)	84(0x54)
R8	8(0x08)	19(0x13)	30(0x1e)	41(0x29)	52(0x34)	63(0x3f)	74(0x4a)	85(0x55)
R9	9(0x09)	20(0x14)	31(0x1f)	42(0x2a)	53(0x35)	64(0x40)	75(0x4b)	86(0x56)
R10	10(0x0a)	21(0x15)	32(0x20)	43(0x2b)	54(0x36)	65(0x41)	76(0x4c)	87(0x57)

## 输出格式

BC6088(A)的输出为波特率 9600，8 个数据位，1 个起始位，1 个停止位的标准 UART 格式。每 1 个字节为一组。

当键盘状态发生变化时，如有按键按下（导通）或释放（断开），BC6088 即输出有变化按键的键值，键值即键值表中的数值。如果按键状态是由断开变为导通状态，则输出不加修改的键值；如果按键状态是由导通变为断开，则输出键值的最高位会被置为 ‘1’，如原键值为 0x01，则变为 0x81，如原值为 0x17，则变化为 0x97，以此类推。如果同时有多个按键状态产生变化，则将所有有变化的按键的值按原键值从小到大的顺序依次输出。如按键 5,18,32 三个按键同时变化，按键 5 和 32 是从断开变为导通，按键 18 从导通变为断开，则 UART 上输出的数据依次为 0x05, 0x92, 0x20。

在系统上电时，BC6088(A)芯片默认所有的按键均为断开状态。如果在上电时，有按键已经处于导通状态，则 BC6088(A)在开始键盘扫描后，对所有已经处于导通状态的按键，都会输出其键值。最极端的情况下，如果所采用的全部是常闭状态的按键，即上电时所有按键都是导通状态，则 BC6088(A)在上电后将连续输出 88 个字节的数据。

BC6088(A)在上电后，在开始扫描键盘前，有一约 200ms 的延时，以确保 MCU 完成上电复位过程并能正确接收 UART 数据。

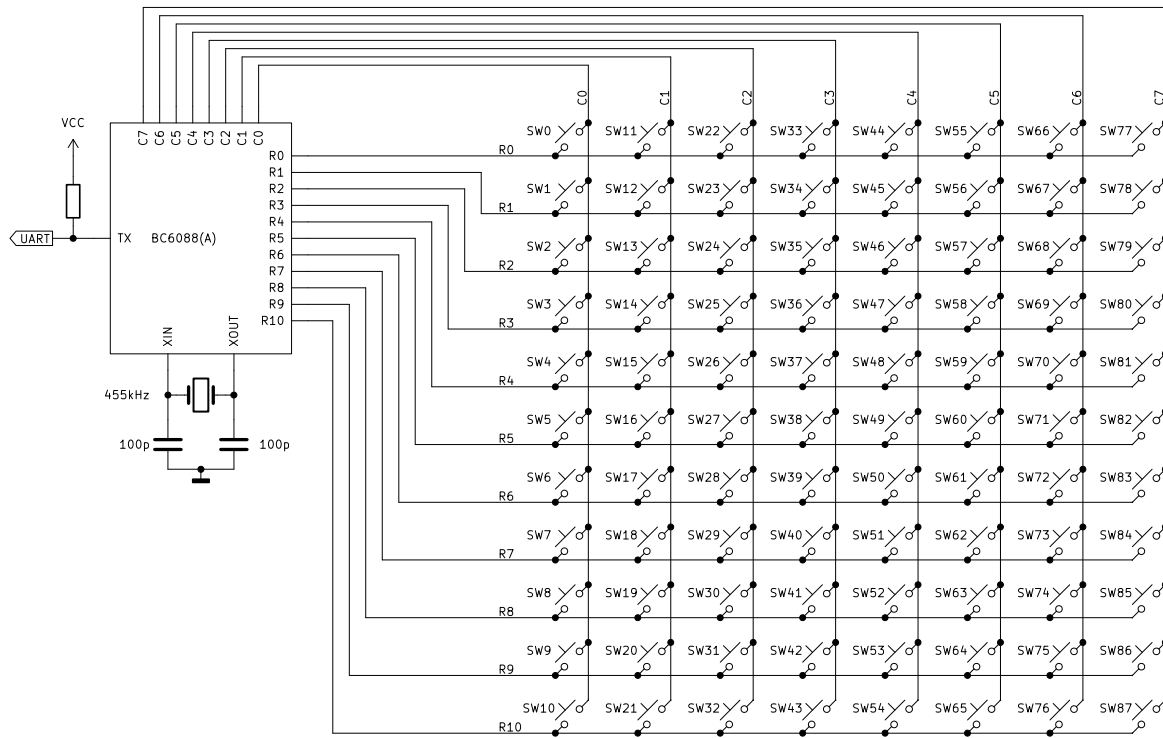
## 典型电路

BC6088(A)由外接的廉价 455kHz 陶瓷谐振器提供 UART 所需的时间基准，如果系统能够提供 455kHz 时钟信号，则也可直接使用输入时钟而不使用陶瓷谐振器。输入的外部时钟信号，应连接到 XIN 引脚。

键盘矩阵由 11 行 8 列组成。列信号 C0-C7 轮流输出正扫描脉冲，行信号 R0-R10 为输入信号，BC6088(A)带有内部下拉电阻，一般应用时，无需再另外接外部下拉电阻。典型电路如下：(见下页)

BC6088(A)的列信号输出的扫描脉冲，宽度约为 1.26ms，扫描周期 BC6088 为 111ms，BC6088A 为 40ms，防抖动时间为 250us。即确保按键动作能够被识别所需的持续时间，为 111ms 或 40ms(普通按键操作，持续时间约在 100-500ms 之间)，而能识别的最短按键持续时间，为 250us。

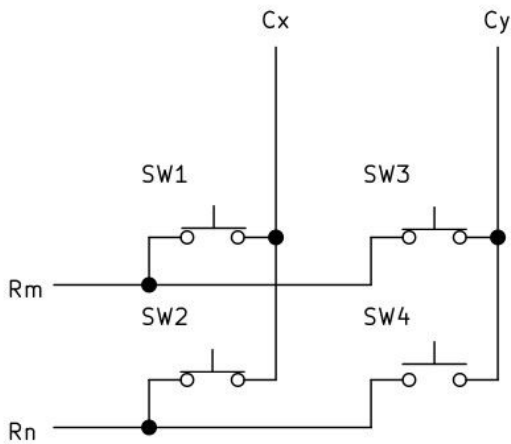
如果键盘的分布面积较大，或周围环境有较强干扰，可以考虑给行信号 Cn 和列信号 Rn 都外接额外的下拉电阻，增强对干扰的抑制能力。所外接的下拉电阻阻值，以不小于 1kΩ 为佳，阻值越小系统整体工作电流将越大。



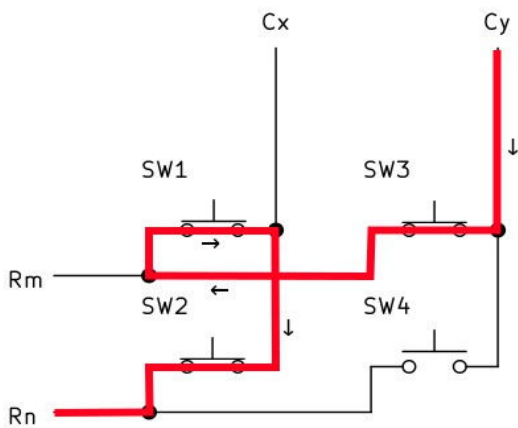
BC6088(A)典型电路图，图中按键标号即为按键键值

## 消除影子按键（鬼影按键）

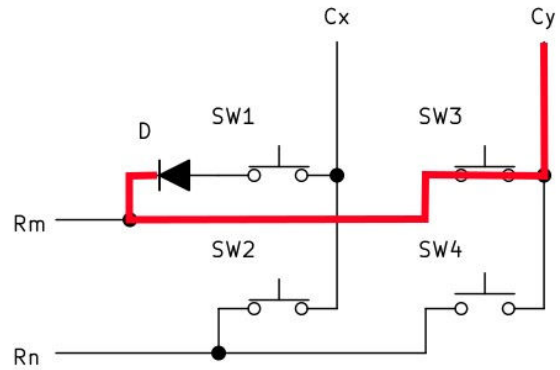
影子按键是矩形键盘矩阵的特有现象，表现为当处在一个四边形四个角上的四只按键中有三只处于导通状态时，系统会误认为第四只按键也处于导通状态。如图中当 SW1,SW2,SW3 三个键都按下时，系统会误以为 SW4 也是按下的状态。



产生这种现象的原因，在于当 SW1-SW3 三只按键均处于导通状态时，因为三只按键所形成的电流回路，实际上第四只按键 SW4 的两端，也是处于连通的状态。假设系统扫描到 SW4 时，Cy 输出高电平，则电流通过如下的回路，会使 Rn 为高电平，和 SW4 被按下效果相同。如图：



四只按键的电流形成回路，因此系统也就不会认为第四只按键是导通状态了。如图：



要彻底消除产生影子按键的可能，需要对所有按键均串入二极管，但这样成本较高，一般实际应用中，仅需对需要防止产生影子按键的个别键采取措施即可。因为二极管会产生正向管压降，从而降低了键盘扫描电路的可靠性，因此，应尽量选用正向导通管压降比较低的二极管，比如肖特基二极管 1N60 等。

解决这个问题的方法，是在没有按下的按键的对角线方位的按键 SW1 中，串入一只二极管。该二极管可以在三只按键均导通时，阻止扫描第

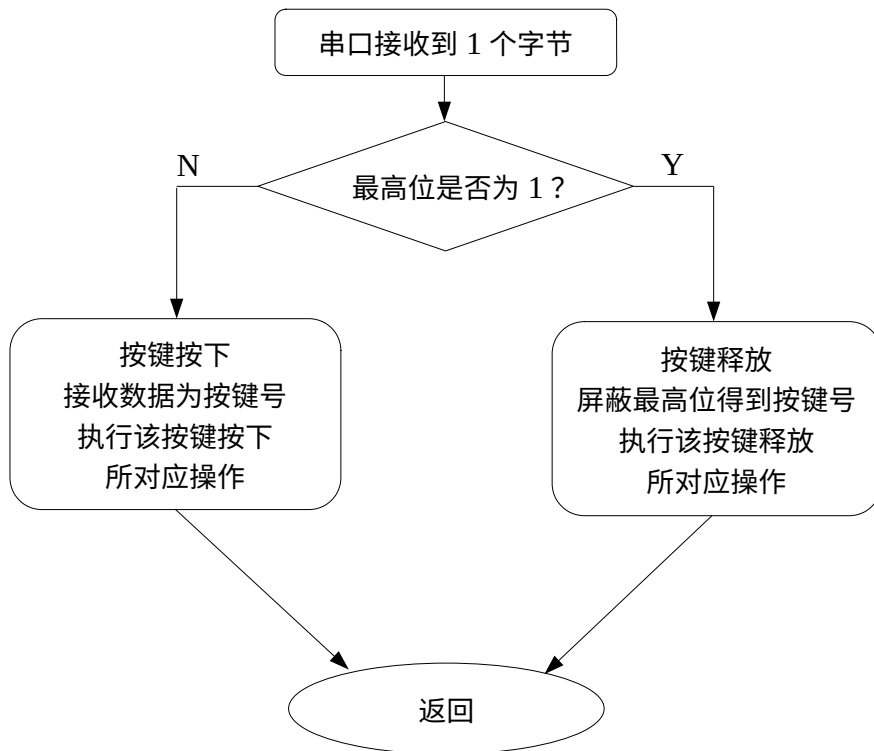
## 典型按键处理流程

BC6088(A)的最佳应用方式，是配合硬件 UART 接口，以中断的方式来处理按键信号。这种方式对系统负担最低，处理器可以将大部分时间用于执行其它操作。如果使用查询的方式，甚至使用软件模拟方式实现 UART 接口接收数据的方式，将耗费大量处理器时间，并不推荐使用。

对于 BC6088(A)所使用的 9600 的波特率，传送一个字节大约需要 1ms 左右的时间，即使在可能有同时多个按键状态变化，BC6088(A)会连续发送多个字节数据的情况下，1ms 的时间对于绝大多数的处理器来说，应该足够在下一个字节到来之前，完成按键的处理。如果应用条件下存在出现多个按键同时变化的可能，且系统不能保证在 1ms 的时间内完成按键的处理，应该考虑

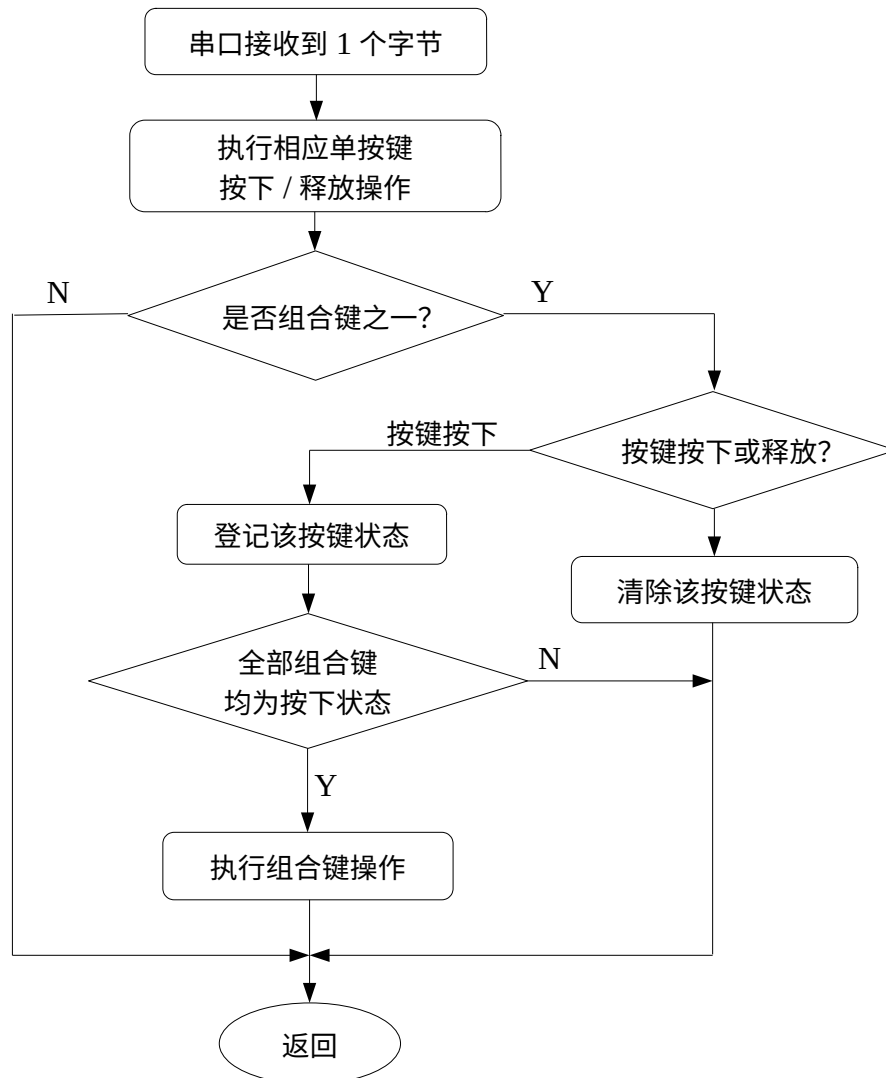
给 UART 分配接收缓存，缓存的大小应大于可能同时变化的按键数量，以免缓存溢出按键数据丢失。

### 普通单次按键（按下和释放分别处理）



## 组合键

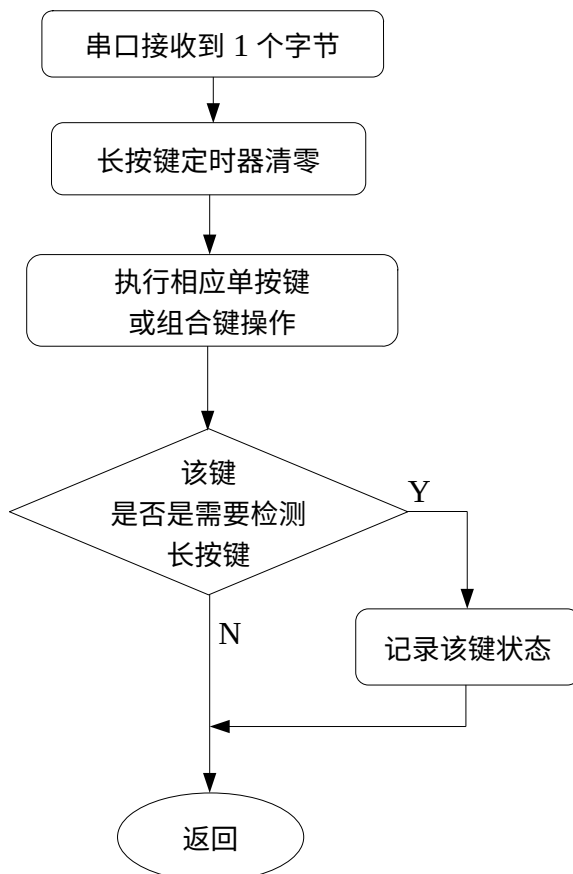
组合键的使用要求程序能记忆按键组合中各键的状态。



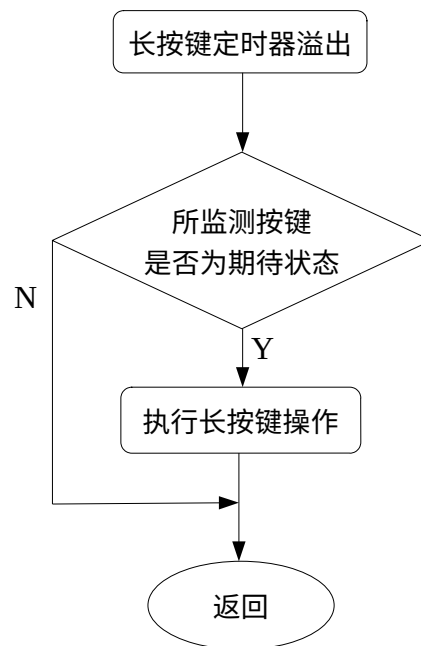
## 长按键

BC6088(A)在键盘状态发生变化时，即从 UART 接口发出相应信息，无论是有按键按下，或有按键释放。换言之，如果 UART 上没有数据输出，即表明键盘维持在上次数据输出时的状态没有改变。利用这个特点，可以很容易地做到长按键的检测，不光是单个按键的长按键检测，组合按键也可以用同样方法检测长按键。用户只需要提供一个计时器，用来确定需要检测的长按键的时长。

串口中断处理程序：



定时器中断处理程序：

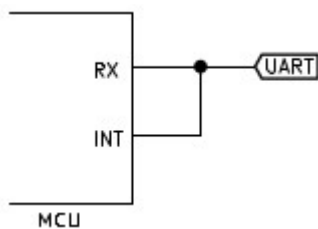


## 控制器(MCU)唤醒

BC6088(A)有着极低的功耗，非常适用于电池供电设备。电池供电设备通常设计为控制器(MCU)在闲时处于休眠状态，而在有用户操作(键盘操作)时，唤醒控制器。

UART 输出是 BC6088(A)传递给 MCU 的唯一信号，因此，如果需要用键盘操作唤醒 MCU，则需要让 MCU 能够被 UART 信号唤醒。根据不同的设计，有些 MCU 可以支持由 UART 直接唤醒，这些产品可以和 BC6088(A)完美配合，不做更多讨论。这里主要探讨无法由 UART 信号直接唤醒的 MCU。

这里提供一个设计思路，供无法由 UART 唤醒的 MCU 使用。一般的 MCU 均具有由外部中断唤醒的功能，而 UART 信号空闲时为高电平，并以一个起始位 '0' 打头，即所有 UART 的数据均以一个下降沿开始。因此，可以将 UART 信号同时提供给 MCU 的外部中断，用来唤醒 MCU。



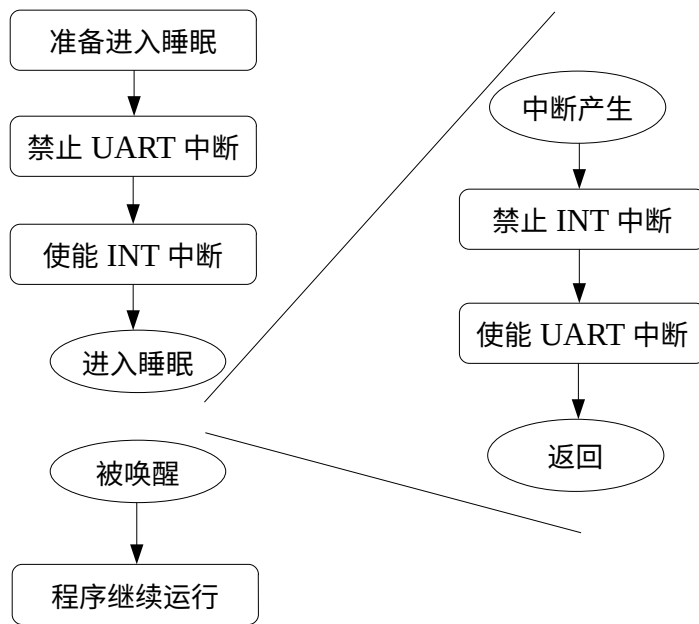
因为 MCU 的唤醒启动时间一般较长，因此使用这个方式唤醒 MCU，UART 上第一个字节数据通常是无法正常接收的，这一般来说并不是问题，可以直接将其忽略。从使用者角度来讲，用户的体验将是，在机器睡眠状态下，按任意键，机器开机，显示、指示灯等得到恢复，然后再按需要的功能键进行所需要的操作。

如果确需捕捉第一个按键，也可以考虑用以下方法：因为 BC6088(A)在键按下和释放时会分别发出两个信号，中间间隔按键的持续时间，这个时间一般在 100ms 到 500ms 间。这时间应可足够让 MCU 进入运行状态，因此按键释放的信号可以确保被捕捉到，程序可以利用按键释放信号确定键值，并执行相应动作。

使用以上的方法，用户须确切知道所用 MCU 从睡眠状态到被唤醒进入运行状态所用的时间，如果这个时间短于 UART 上传送一个字节的的时间(9600 波特率下，8 个数据位加上一个起始位，时间是 938us)，即在唤醒 MCU 的字节传送到中途时 MCU 的 UART 进入了接收状态，会造成收到的第一个字节为随机乱码，这种情况下必须将唤醒后所收到的第一个字节丢弃。以下流程图假设 MCU 的唤醒时间 > 938us。

主程序：

唤醒中断 (INT)：



## 极限参数

储存温度	-65°C - +150°C
工作温度	-40°C - +85°C
任意脚对地电压	-0.5 - +6.0V

## 电气特性

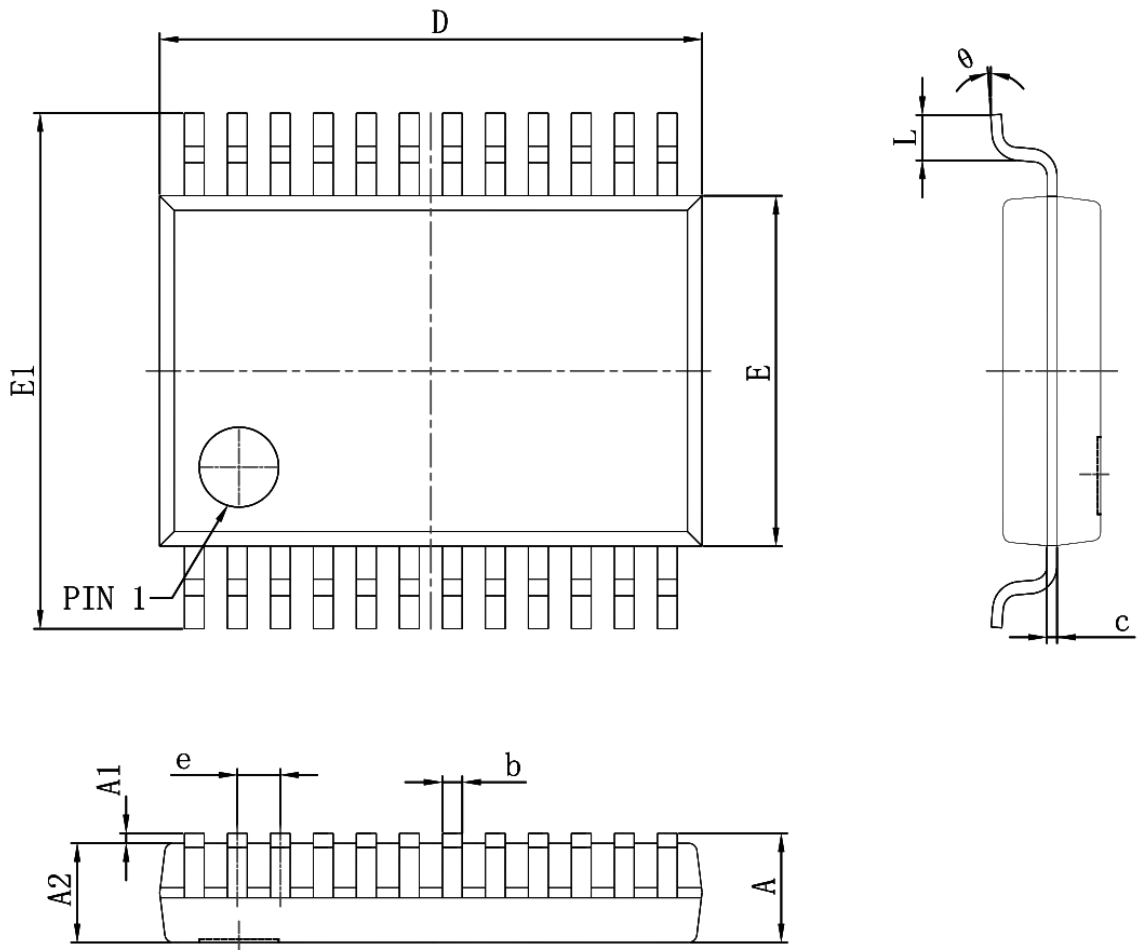
(注：以下数据除特别注明外， $T_A=25^\circ\text{C}$ )

参数	最小值	典型值		最大值	备注
		BC6088	BC6088A		
$V_{CC}$ 工作电压	2.0V			5.5V	
$I_{CC}$ 工作电流*		23uA	72uA		$V_{CC}=2.5V$
		39uA	132uA		$V_{CC}=3.0V$
		169uA	479uA		$V_{CC}=5.0V$
$V_{RH}$ 行扫描输入高电平	2.0V				$V_{CC}=3.0V$
	4.0V				$V_{CC}=5.0V$
$T_{CL}$ 键盘扫描周期		111ms	40ms		$V_{CC}=3.0V$
$T_{PL}$ 键盘扫描脉冲宽度		1.26ms			
$T_{db}$ 去抖动时间		250us			
$T_{pu}$ 上电等待时间		200ms			$V_{CC}=3.0V$
$R_{PD}$ 行输入等效下拉电阻		240K			$V_{CC}=3.0V$
		128K			$V_{CC}=5.0V$

\* 工作电流为无外接键盘电路情况下的电流，当接入键盘电路时，分布容性负载将使功耗增加

# 封装尺寸

## SSOP24(209mil) PACKAGE OUTLINE DIMENSIONS



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A		1.730		0.068
A1	0.050	0.230	0.002	0.009
A2	1.400	1.600	0.055	0.063
b	0.220	0.380	0.009	0.015
c	0.090	0.250	0.004	0.01
D	8.000	8.400	0.315	0.331
E	5.100	5.500	0.201	0.217
E1	7.600	8.000	0.299	0.315
e	0.65 (BSC)		0.026 (BSC)	
L	0.550	0.950	0.022	0.037
theta	0°	8°	0°	8°