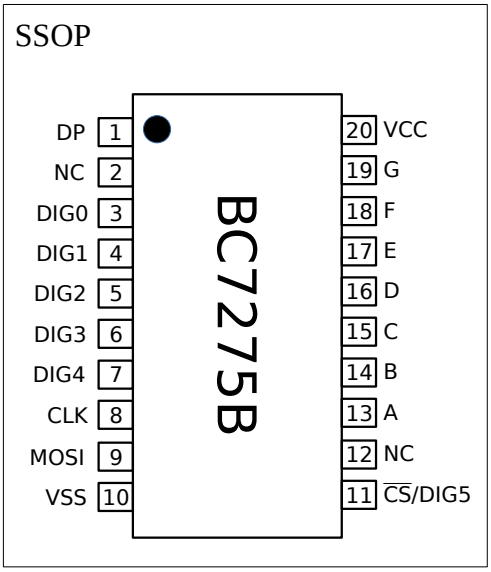
BC7275B

48-segment 6¾ digits LED 7-segment numeric display driver

Features:

- Compatible with BC7275A, supports extended digit
- 6¾ digits, continuous display from 0,000,000 to 6,999,999
- Each segment independently blinkable, adjustable blink speed
- Built-in hexadecimal decoder
- SPI port, with or without CS
- All LEDs can be accessed individually
- Directly write to display register
- Software compatible with other BC727x chips
- SSOP20 package, no external component



Abstract

The BC7275B features a single chip 5¾-digit or 6¾-digit LED display management. By supporting segment addressing, each display segment can be controlled independently so it is also very suitable for driving individual LEDs, up to 48 individual LEDs can be driven. The BC7275B supports blinking display function with adjustable blinking speed, each LED can be controlled independently for blinking property, or blinking can be controlled by digit for a numeric display.

The BC7275B provides an internal decoding function that allows the user to write a value directly into the decoder register to obtain the corresponding numeric display. The decimal point segment is not affected by the decoding, so the user only needs to update the figure data and does not need to consider about refreshing the decimal point, which is particularly convenient for users who has fixed decimal point and will use that segment as a separate LED. It also supports writing to display register directly, which can be used for displaying some special characters such as ‘L’, ‘H’, ‘P’.

The SPI port supports communication rate up to 64Kbps. The unique internal SPI port reset logic allows the SPI interface to be reset when the communication is idle, ensuring error-free communication when CS is not present. When CS is not used, it can be used as an extra digit drive pin, get 8 display segments more.

Compatibility

BC7275B is 100% compatible with BC7275A, can be used as a pin to pin upgrade. Comparing to BC7275A, the BC7275B has the following improvements:

1. Some commands related to the extended display digit are added to make it possible to use the decimal point segments to form an extended display digit to get an additional ¾ display.
2. Refresh rate has been increased up to 20%, less flickering effect.

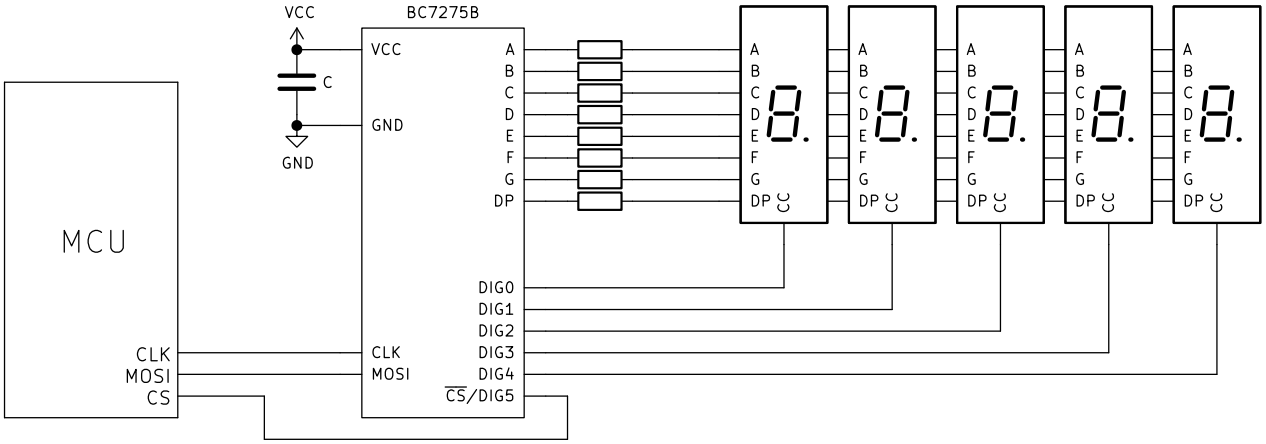
Pin Descriptions

Pin Name	Pin Number	Description
DIG0-DIG4	3,4,5,6,7	Digit drive output, connected to the common cathode pins of 7-segment numeric LED display
CLK	8	SPI clock, input, bit rate <=64kbps, idle high, internal pull-up
MOSI	9	SPI data input, connected to data output of MCU
GND	10	Ground
$\overline{CS}$/DIG5	11	Chip Select, active low. Internal pull-up resistor. Can be used as 6^{th} digit driving output
A-G, DP	13,14,15,16,17,18,19	Segment drive output, connected to the anode pins of LEDs or 7-segment displays.
VCC	20	Power supply, 2.7-5.5V

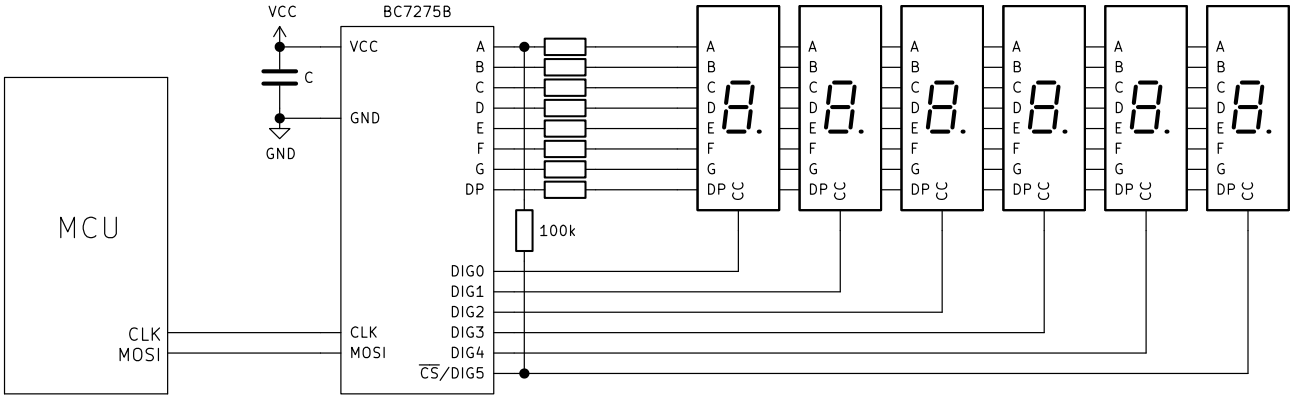
Typical Application

The BC7275B circuit is simple and only requires 8 current limiting resistors to form a multi-digit LED display, and the BC7275B can be configured as a 5 or 6-digit display, with the number of digits determined by the connection of CS/DIG5 pins. If CS/DIG5 is pulled-up or directly grounded at power-up, the chip will operate in 5-digit mode; if CS/DIG5 is connected to any of the segment drive outputs (A-DP) behind the LEDs, the BC7275B will operate in 6-bit mode, and CS/DIG5 is used as the digit driver for the 6th digit. To ensure reliable mode selection, an additional resistor (100K typical) can be added between CS/DIG5 and any of the segment drive (A-DP) outputs (see figure).

5 digits 8 segments mode (can have decimal point on each digit)



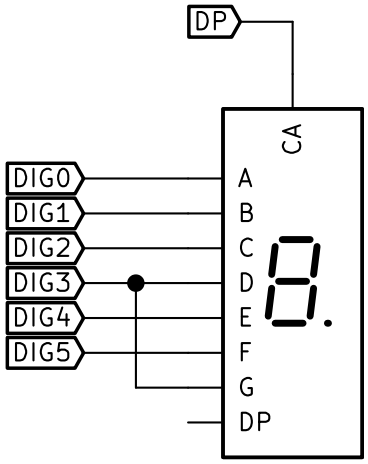
6 digits 8 segments mode (can have decimal point on each digit)



Extended Digit

When the decimal point is not needed, or the decimal point position is fixed, the decimal point segment of each digit can be used to drive an additional digit, adding ¾ display digit (i.e., the full 0-9 range of all digits cannot be displayed, but 1-3 or 0-6 can be displayed depending on the configuration). Although the display of numbers can be achieved by controlling each decimal point individually through segment addressing instructions, this requires multiple instructions. BC7275B provides two registers dedicated to the use of extended digit, which can be written directly to the extended digit in a decoded or direct manner, requiring only one instruction. For the circuit connection of the extended digit, two options are available.

Option 1



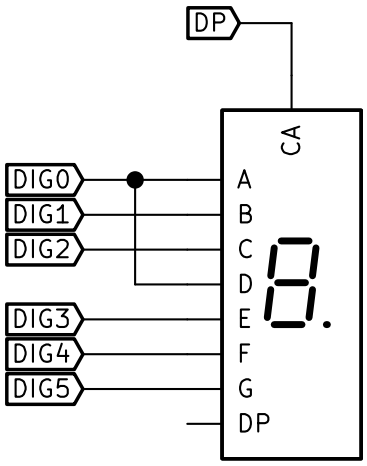
Option 1 uses the most natural connection method, the DP segment of the 0th digit is used as the extended bit A segment, the DP segment of the 1st digit is used as the extended digit B segment, and so on to the DP segment of the 5th digit is used as the F segment. Because there are only 6 DP segments available and the G segment is missing, the G segment of the extended digit is put on the D segment and the two are lit and off at the same time, so that the numbers 1-3 can be displayed continuously, and also 5,6,7,8,9, all the numbers with the same status of the D and G segments, can be displayed. This scheme only needs to use DIG0-DIG4 when displaying 1-3, so it can also be used when in 5-digit mode. Please see the section "Extended Digit Decoding Register" later.

Attention should be paid that the expansion digit and the normal digit circuits are different. The normal digit of the BC7275B uses a common cathode 7-segment display, while the expansion digit, requires the use of a common anode type.

Option 2

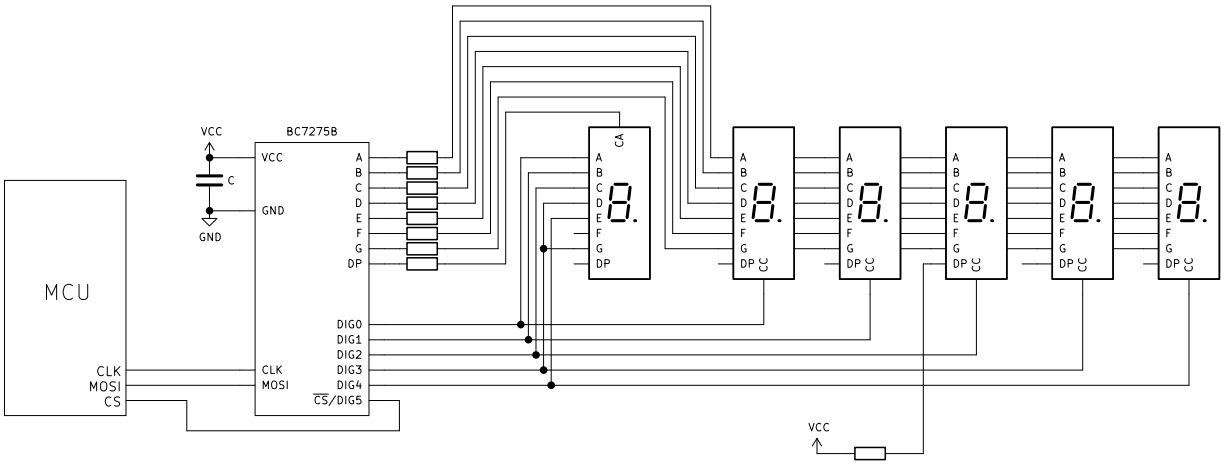
Connection option 1 is simple to use, it can use the decoding instruction, only need to feed the value to be displayed, but this scheme can not display the number "4", so the range of continuous display is limited. To solve this problem, Option 2 can be used, which can continuously display numbers from 0 to 6 and can be used in a wider range of applications. This solution is to connect segment A and segment D in parallel, which requires the use of a special decoding table, so it is not possible to use the internal decoding commands, and the decoding can only be done by the user program, which is slightly complicated in programming. Moreover, since DIG5 is used for the display of figure 2, it can only be used when the BC7275B is operating in 6-digit mode. Please see the section "Extended Digit Mapping Register" later.

Same as in option 1, the extended digit should use a common anode type.

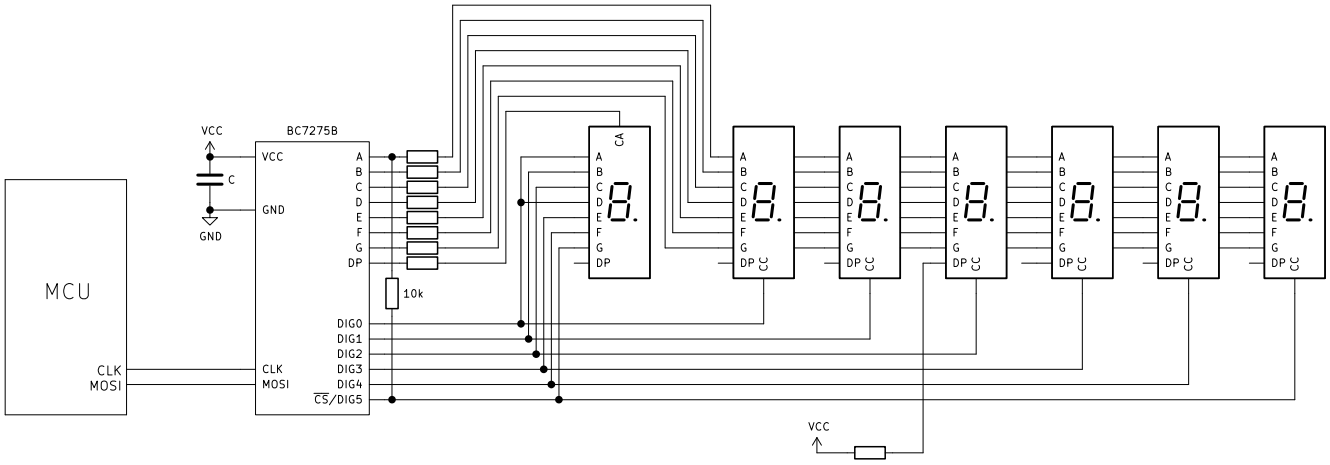


Typical ciruit with extended digit

5¾ digit 7 segments mode (fixed decimal point position, circuit option 1, built-in decoding, continuous display range 000.00-3999.99):



6¾-digit 7 segments mode (fixed decimal point position, circuit option 2, extended digit must use user-defined mapping table, continuous display range 0000.000-6999.999).



Selection of current limiting resistors

The segment drive current limiting resistor can be calculated using the following approximate formula.

$$R = 67*(V_{CC}-V_{LED}) - 100$$

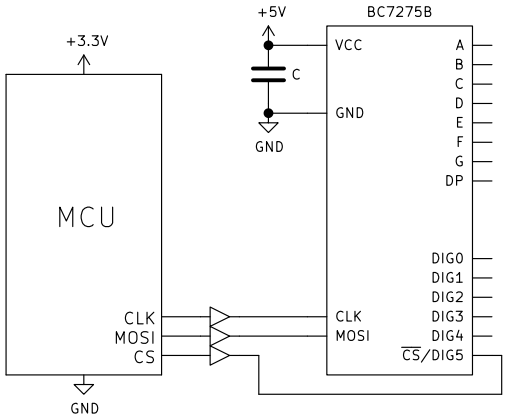
Where VCC is the supply voltage on the BC7275B, U_{LED} is the forward voltage drop of the LED (@10mA), the voltage unit is volts, and the calculated resistance unit is in ohms. For example, when VCC = 5V and U_{LED} is 1.85V, the calculated resistance is about 111Ω, can take an approximate value of 100Ω. When VCC is 3.3V and U_{LED} is 1.85V, the calculated current limiting resistance is -2.85Ω, you can use an approximate value of small resistance, or just omit. When VCC is 3V, the calculation output is -23Ω, indicating that the current-limiting resistance can be omitted in this situation, and the brightness might be compromised.

Interface

The BC7275B can be connected to a standard synchronous serial interface, such as the SPI interface, in a standard 3-wire configuration, including data (MOSI), clock (CLK), and chip select (CS), or in a 2-wire configuration by omitting the chip select CS signal. In the 2-wire mode, CS/DIG5 can be directly grounded or used as the bit driver for the 6th digit.

Since the high level minimum input voltage of the interface section is 3.7V at 5V supply voltage, the BC7275B cannot be directly interfaced to a 3V system if the supply voltage of the BC7275B is 5V, since the BC7275B itself can operate at 3V supply voltage, it is recommended to use the same supply voltage for the BC7275B when the system MCU is powered by 3V or 3.3V.

If you really need to make the BC7275B work at 5V, you need to add a level conversion circuit. 74HCT series, when the supply voltage is 5V, the high level minimum input voltage is 2V, which can directly accept the output of 3V system. The level matching problem can be solved by using such a buffer on the communication line. As shown in the figure.



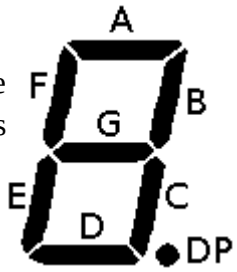
Registers

There are 19 internal registers, including 6 display registers, and 13 control registers. The address range is 00H-20H, where 00H-05H are display registers and the rest are control registers.

Addr	Name	Default	Comment
00H	Display register for digit 0	FFH	Each bit is mapped to a display segment, 1=OFF，0=ON
01H	Display register for digit 1	FFH	
02H	Display register for digit 2	FFH	
03H	Display register for digit 3	FFH	
04H	Display register for digit 4	FFH	
05H	Display register for digit 5	FFH	Only available when working in 6 digits mode
...	(not used)		
10H	Blink control for digit 0	00H	Each bit is mapped to a display segment, 1=blink，0=normal
11H	Blink control for digit 1	00H	
12H	Blink control for digit 2	00H	
13H	Blink control for digit 3	00H	
14H	Blink control for digit 4	00H	
15H	Blink control for digit 5	00H	Only available when working in 6 digits mode
...			
18H	Character blink control	00H	Bit0-bit5 are mapped to digit 0-5, 1=blink,0=normal
19H			
1AH	Blink frequency control	10H	Smaller value for faster blinking
1BH	Hexadecimal decoder	-	Value written to this register is displayed as a 0-F hexadecimal character on a 7-segment display
1CH	Segment control	-	To control each segment individually
1DH	Global write	FFH	Value will be written to all display registers

Addr	Name	Default	Comment
1EH	Hexadecimal decoder for extended digit	-	The output is mapped to DP segment of each digit
20H	Virtual display register for extended digit	-	The output is mapped to DP segment of each digit

Display Registers：Address 00H-05H



The display register is directly mapped to each LED display segment, and the mapping relationship between the bits and each LED on the 7-segment is as shown in the figure

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
DP	G	F	E	D	C	B	A

User can directly change the contents of the display register to update the display, which is mainly used for displaying special characters that are not in the decoding table, such as 'H', 'L', 'P', '-' etc.

If a bit in the display register is set to 0, the display segment is ON. After reset, the contents of all display registers are set to FFH.

Blink Control Registers：Address 10H-15H

Similar to the display registers, the blink control registers also uses the same bit mapping scheme to control the blink of the LEDs, each bit corresponds to a display segment.

When the bit is 1, the corresponding LED has the blink property ON. Blinking occurs only when that display segment is lit. If that display segment is OFF (the corresponding display register bit is set to 1), nothing is displayed for that display segment. When the content of the display register is cleared (the corresponding bit is set to 1), its blinking property is not affected, and when that segment(LED) is set to 0(ON) again, it will still be a blinking display.

After reset, the blink control registers are all cleared to zero (non-blinking).

Character Blink Control Register: Address 18H

The character blink control register controls the overall blink property of a 7-segment character, each bit in the register corresponds to a display character, the correspondence is as follows

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
-	-	Digit 5	Digit 4	Digit 3	Digit 2	Digit 1	Digit 0

When a bit is 1, the character is in blinking mode. Blinking occurs only if that character has some display content; if that character is blank (all bits of the display register are 1), there will be no blinking. When reset, the character blinking control register is set to 00H (no blinking).

Blink Frequency Control Register: Address 1AH

The BC7275B has an adjustable blink frequency, and it can be easily controlled by simply changing the blink frequency control register. The larger the value in the register, the slower the blink speed. After reset, the value of this register is 10H, and at this value, the blink rate is about 2Hz.

Hexadecimal Decoder Register: Address 1BH

Through the decoding register, the user can get the hexadecimal number displayed on a 7-segment directly by feeding in the value, saving the user from the trouble of preparing the mapping table by himself. The format of the data written into the decoding register is as follows.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
A_3	A_2	A_1	A_0	D_3	D_2	D_1	D_0

Where A3:A0 is the character address, which determines the position of the hexadecimal digit display. The d3:d0 is the value to be displayed. When the address is 0000b, the decoding result is displayed on digit 0, and when it is 0011b, it is displayed on digit 3. A3 : A0 has the value range of 0000-0011. d3:d0 is the value to be displayed. The decoding table is as follows.

d_3	d_2	d_1	d_0	d_3:d_0(hex value)	Display
0	0	0	0	0	
0	0	0	1	1	
0	0	1	0	2	
0	0	1	1	3	
0	1	0	0	4	
0	1	0	1	5	
0	1	1	0	6	
0	1	1	1	7	
1	0	0	0	8	
1	0	0	1	9	

d_3	d_2	d_1	d_0	d_3:d_0(hex value)	Display
1	0	1	0	A	
1	0	1	1	B	
1	1	0	0	C	
1	1	0	1	D	
1	1	1	0	E	
1	1	1	1	F	

Updating the display through the decoder register will not affect the decimal point, i.e. the state of the DP segment will remain unchanged. With this feature, when data with decimal point is displayed, or in the case of using the decimal point as independent indicator, the data update will be very convenient, because it can be done without considering refreshing the decimal point. The control of the decimal point can be done by segment control register.

Segment Control Registers: Address 1CH

The display can be controlled on a segment (individual LED) basis, which is achieved through segment control register.

By assigning an address to each display segment (LED), the segment control registers can be used to control the ON and OFF of each display segment. The address of each segment is as follows.

digit	DP	G	F	E	D	C	B	A
0	07H	06H	05H	04H	03H	02H	01H	00H
1	0FH	0EH	0DH	0CH	0BH	0AH	09H	08H
2	17H	16H	15H	14H	13H	12H	11H	10H
3	1FH	1EH	1DH	1CH	1BH	1AH	19H	18H
4	27H	26H	25H	24H	23H	22H	21H	20H
5	2FH	2EH	2DH	2CH	2BH	2AH	29H	28H

The data format written to the segment control register is as follows.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SEG	A_6	A_5	A_4	A_3	A_2	A_1	A_0

Where A0-A6 is the segment address. Seg is the data written to the segment. when Seg is 0, the LED is ON, and when Seg is 1, the LED is OFF.

Global Write Register: Address 1DH

The global write register is a special register, the value written to this register is copied to all display registers at the same time. For purpose of clearing the display or turning on all LEDs, it can be easily achieved by writing FFH or 00H to this register. Note that writing to this register does not change other registers such as blink control, so if a display segment or character is set to blink before writing, even if FFH is written to the global write register, all the blink settings will remain unchanged.

Hex decoder for extended digit: Address 1EH

When the decimal point segment DP of each display digit is used for an extended display digit (see the extended display digit circuit diagram in the previous section), writing to this register is as writing hexadecimal decoder for a normal digit. Register data format.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
-	-	-	-	D_3	D_2	D_1	D_0

When the extended digit is connected according to scheme 1, the numbers 1-3 can be displayed directly by writing to the decoder register. With this scheme the numbers from 00000-399999 continuously can be displayed. The display decode table is as follows.

D_3	D_2	D_1	D_0	hex value	ON segments after writing decoder reg.	Display on extended digit
0	0	0	1	01H	DP_1, DP_2	
0	0	1	0	02H	DP_0, DP_1, DP_3, DP_4	
0	0	1	1	03H	DP_0, DP_1, DP_2, DP_3	
0	1	0	1	05H	DP_0, DP_2, DP_3, DP_5	
0	1	1	0	06H	DP_0, DP_2, DP_3, DP_4, DP_5	

D_3	D_2	D_1	D_0	hex value	ON segments after writing decoder reg.	Display on extended digit
0	1	1	1	07H	DP_0, DP_1, DP_2	
1	0	0	0	08H	DP_0, DP_1, DP_2, DP_3, DP_4, DP_5	
1	0	0	1	09H	DP_0, DP_1, DP_2, DP_3, DP_5	

Virtual display register for extended digit: Address 20H

The function of this register is that when data is written to this register, the data will be mapped to the corresponding segment of the extended display digit, similar to the function of writing to the display register for a normal digit. The true action is to map the written data to bit7 (DP bit) of each display register, so when using the 8-segment connection method (see the section on connection method later), you can also use this instruction if you need to change the state of multiple decimal point in one action.

Data format for writing to registers:

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
-	-	DP_5	DP_4	DP_3	DP_2	DP_1	DP_0

Where DP0-DP5 represent the DP segment from digit 0 to digit 5, and the corresponding DP segment lights up when the written data is 0.

Using the mapping method, if the BC7275B works in 6-digit mode, with the appropriate circuit connection scheme (as shown below) and code table, it is possible to display all the numbers except '7' on the extended digit, forming a 6¾-digit display

D_5	D_4	D_3	D_2	D_1	D_0	hex value	ON segments after writing decoder reg.	Display on extended digit
1	0	0	0	0	0	20H	DP_0,DP_1,DP_2,DP_3,DP_4	
1	1	1	0	0	1	39H	DP_1,DP_2	
0	1	0	1	0	0	14H	DP_0,DP_1,DP_3,DP_5	
0	1	1	0	0	0	18H	DP_0,DP_1,DP_2,DP_5	
0	0	1	0	0	1	09H	DP_1,DP_2,DP_4,DP_5	
0	0	1	0	1	0	0AH	DP_0,DP_2,DP_4,DP_5	

D_5	D_4	D_3	D_2	D_1	D_0	hex value	ON segments after writing decoder reg.	Display on extended digit
0	0	0	0	1	0	02H	DP_0,DP_2,DP_3,DP_4,DP_5	
0	0	0	0	0	0	00H	DP_0,DP_1,DP_2,DP_3,DP_4,DP_5	
0	0	1	0	0	0	08H	DP_0,DP_1,DP_2,DP_4,DP_5	

SPI Port

The BC7275B uses an SPI serial interface that can be connected to a standard 3-wire SPI port (with or without using the CS chip select signal). The bit rate is up to 64Kbps. For MCUs without hardware SPI resources, it can also be implemented with I/O ports in a very short code (bit banging).

Generally the SPI interface on a MCU can be set to different modes, and when connected to the BC7275B, it should be set to host (Master) mode, with CLK idle high and data sent on the first clock edge and sampled on the second clock edge. The clock frequency must be below 64KHz.

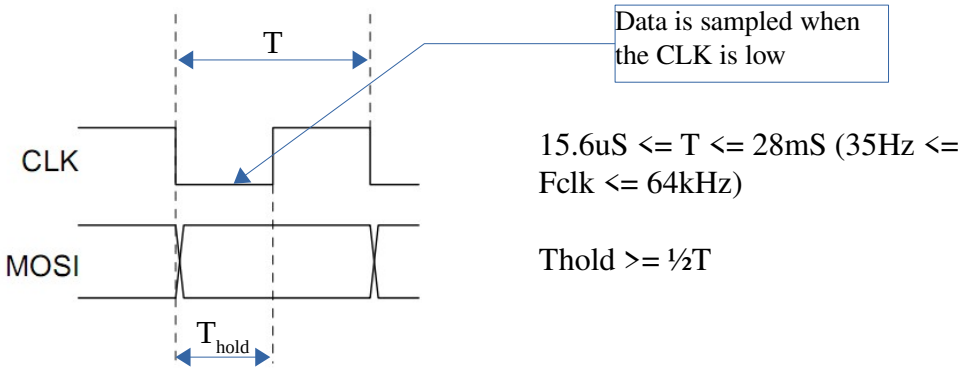
1. Data Format

All the BC7275B commands are formed by 2 bytes. The first is the register address, and the second is the data. The most significant bit (MSB) comes first when it's transmitted. The data structure is as follows.

Register Address								Data							
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	0	A_5	A_4	A_3	A_2	A_1	A_0	d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0
MSB → LSB															

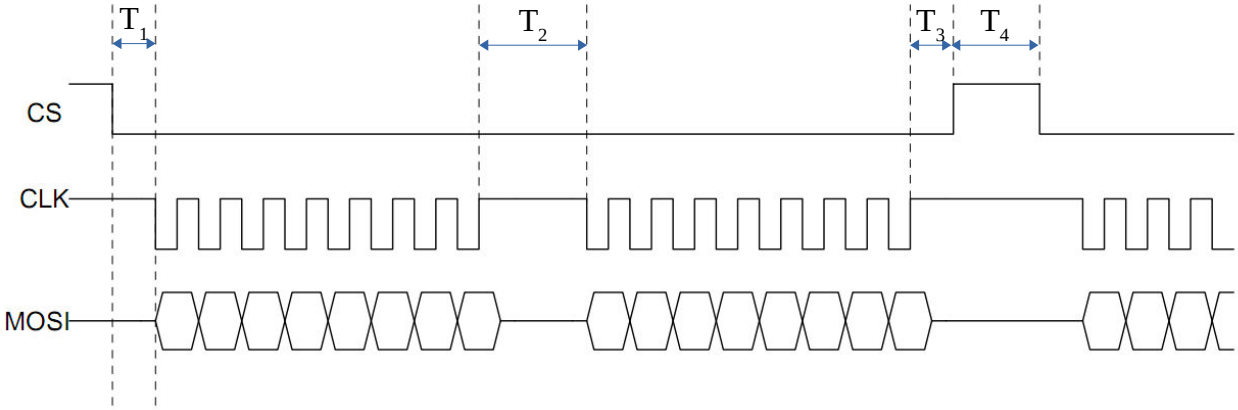
The lower 6 bits A5-A0 of the command byte are the register address, and the valid address range is 00H-20H. If the value is over the valid range, that command will be ignored.

2. Timing



If you are using hardware SPI, please set the port to the following parameters: CLK idle high, data changed at the first clock edge (falling edge), sampling happened at the second clock edge (rising edge) with the data rate ≤ 64Kbps.

If using the bit banging method, since the data is sampled by the BC7275B on average during the CLK low level, it is necessary to set the data line before the CLK level changes to low and maintain the data line stable for at least half the clock cycle or more (Thold>½T).



In the above figure, T1 is the time from CS falling edge to the first CLK clock pulse falling edge, it is required it's more than ½ clock cycle T. When using hardware SPI interface, T1 is generally guaranteed by hardware. User will set CS to low, and then write the SPI register, the hardware will add a delay before the first CLK pulse is output.

T2 is the time from the rising edge of the last clock pulse of the address byte to the falling edge of the first clock pulse of the data byte in a command. T2 should not be shorter than ½T. When using the hardware SPI, this time is guaranteed by the hardware, but if your controller cannot generate this time, or if the SPI is emulated by software, it is necessary to manually added a delay here. If the SPI can supports 16-bit mode, the address byte and data byte can be combined into a single 16-bit data transmission and the T2 is guaranteed.

T3 is the time from the rising edge of the last clock pulse to the rising edge of CS, which is required to be greater than ½ clock cycle T. When using the hardware SPI, if the user program waits for the

SPI interface state to change to "transmit complete" before setting CS to 1, this time is guaranteed by hardware because there is exactly ½ clock cycle between the last rising edge of the clock pulse and the end of the entire clock cycle. There are exactly ½ clock cycle left between the last rising clock pulse and the end of the entire clock cycle. If the SPI port used does not conform to this feature, or if the SPI is emulated by software, this delay time needs to be added manually.

T4 is the delay time between the rising edge of the previous CS signal and the falling delay of the next CS signal, which is required to be greater than ½ clock cycle T.

The CS can be set to high between each instruction. CS has two roles, one is to act as a chip select signal when there are multiple SPI devices in the system, and the other is to reset the SPI serial registers internally at the falling edge of the CS to ensure correct data transmission. The operation of restoring CS to high after each instruction is not mandatory. When transmitting a batch of instructions continuously, CS can be kept low and then restored to high when all instructions are completed.

3. Interface Reset Mechanism

The SPI port must have some mechanism to allow the serial interface to be reset in case the transmitted data is corrupted when the clock and data lines are out of sync for some reason. When the CS signal is used, the interface is reset on the falling edge of CS.

When the CS is directly grounded or used as digit drive, the BC7275B has a unique time reset mechanism that takes effect when the clock line is "muted" for more than 2 display scan cycles (about 32mS in 6-digit mode). In other words, when there is no level change on the CLK pin for more than 2 scan cycles, the currently received data will be discarded and the data corresponding to the next CLK pulse will be treated as the MSB for the new command.

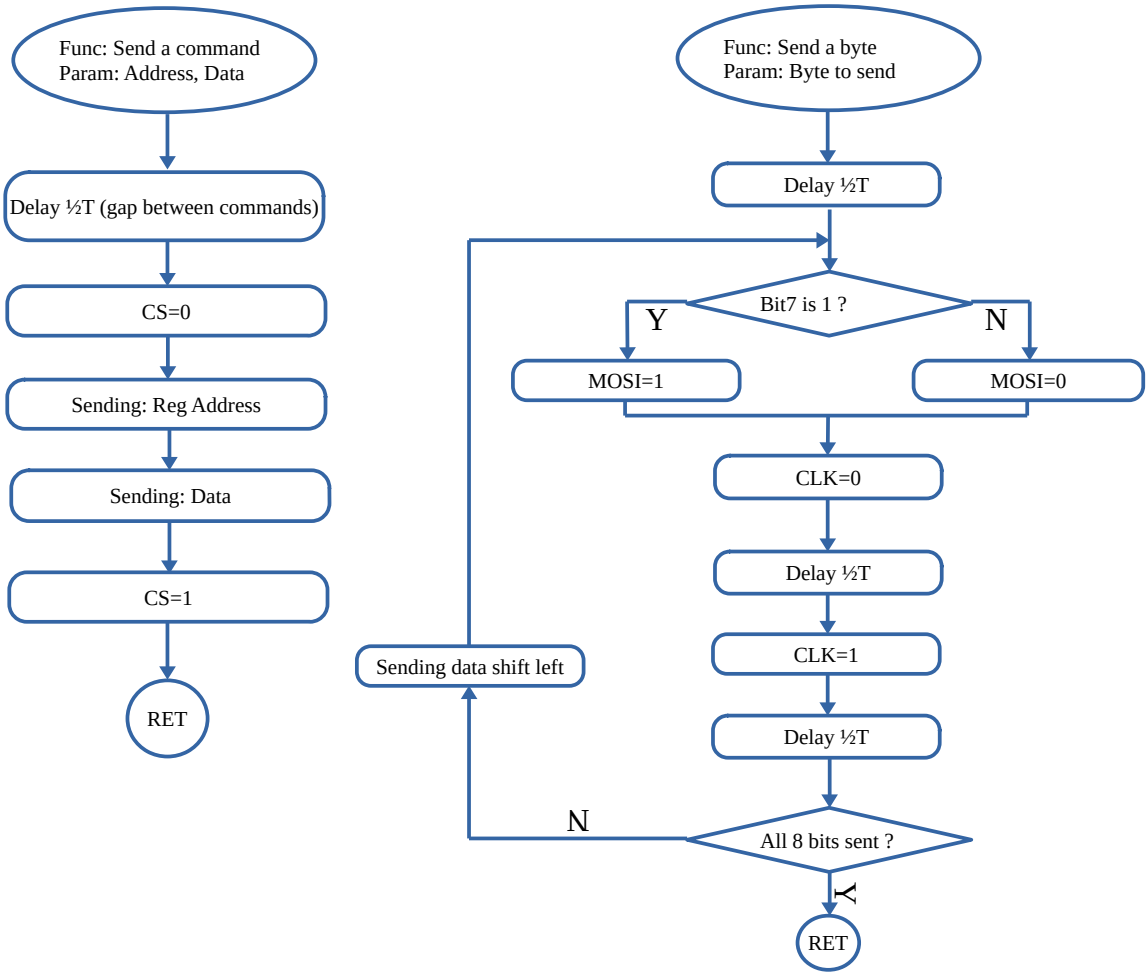
This mechanism will only be activated if CS/DIG5 is low at system power-up, or when the BC7275B has LEDs connected to CS/DIG5 pins. If CS/DIG5 is high when the system is powered on, the time reset mechanism will not be activated and the BC7275B will not detect the time when the CLK signal is idle, only the CS signal can reset the SPI interface.

When the BC7275B is in 6-digit mode, the CS will still be functional even if the time reset mechanism is in effect, i.e., if CS goes high, the BC7275B will not receive any data, while the BC7275B's serial buffer will still be reset on the falling edge of the CS signal.

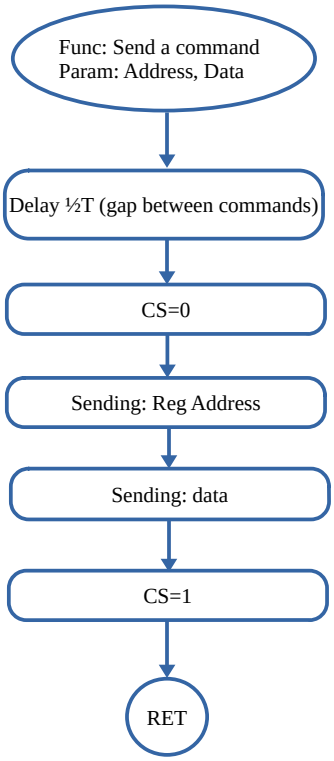
Program Flowcharts

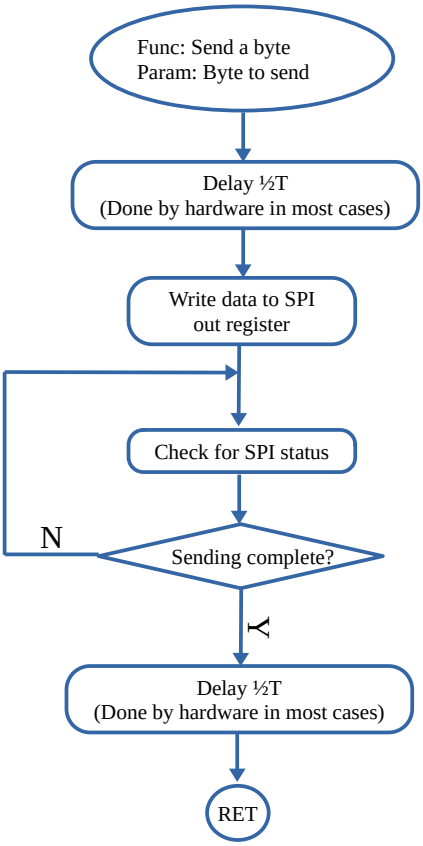
For users who do their own coding instead of using the driver library, please refer to the following flowcharts, which is divided into three types: software simulated(bit banging) SPI, hardware SPI interface + query method and hardware SPI + interrupt method.

Flowchart For Software SPI:



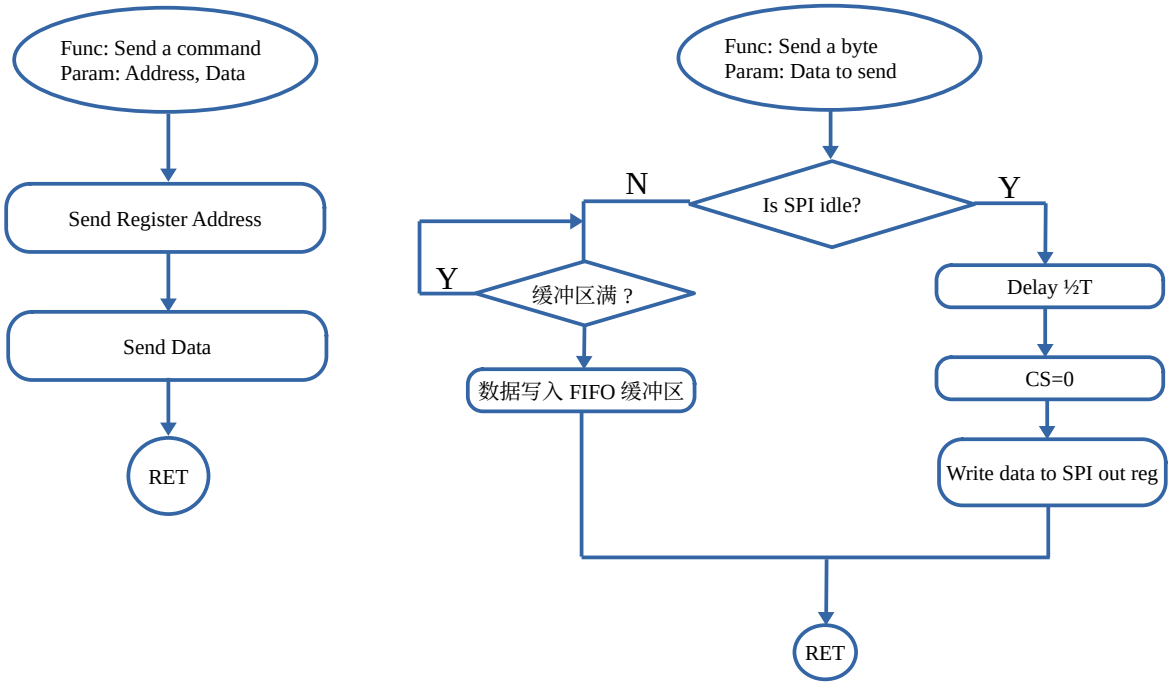
Flowchart For Hardware SPI+Query



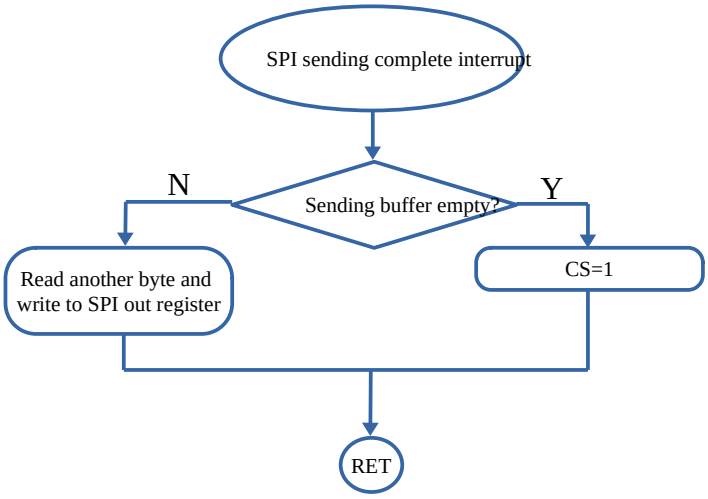


Flowchart For Hardware SPI+Interrupt:

(Note: The flowchart omits the ½T delay that may be required after the CS falling edge and before the rising edge)



ISR(Interrupt Service Routine)



Note: In the interrupt mode, due to the sending buffer, there is a problem that the call of the "send a command" subroutine and the actual time when the data is sent on the SPI port are not synchronized, and for the same reason, the "send a byte" subroutine cannot return with the read-in data. Therefore, when using the interrupt method, a receive buffer is needed to store the received keyboard mapping data along with the send buffer. The SPI interface buffer increases the difficulty of keyboard processing.

Absolute Ratings

(Note: Exceeding the listed range may cause permanent damage to the device)

Storage Temperature	-65 to +150°C
Working Temperature	-40 to +85°C
Voltage on any pin	-0.5 to 6.0V

Electrical Characteristics

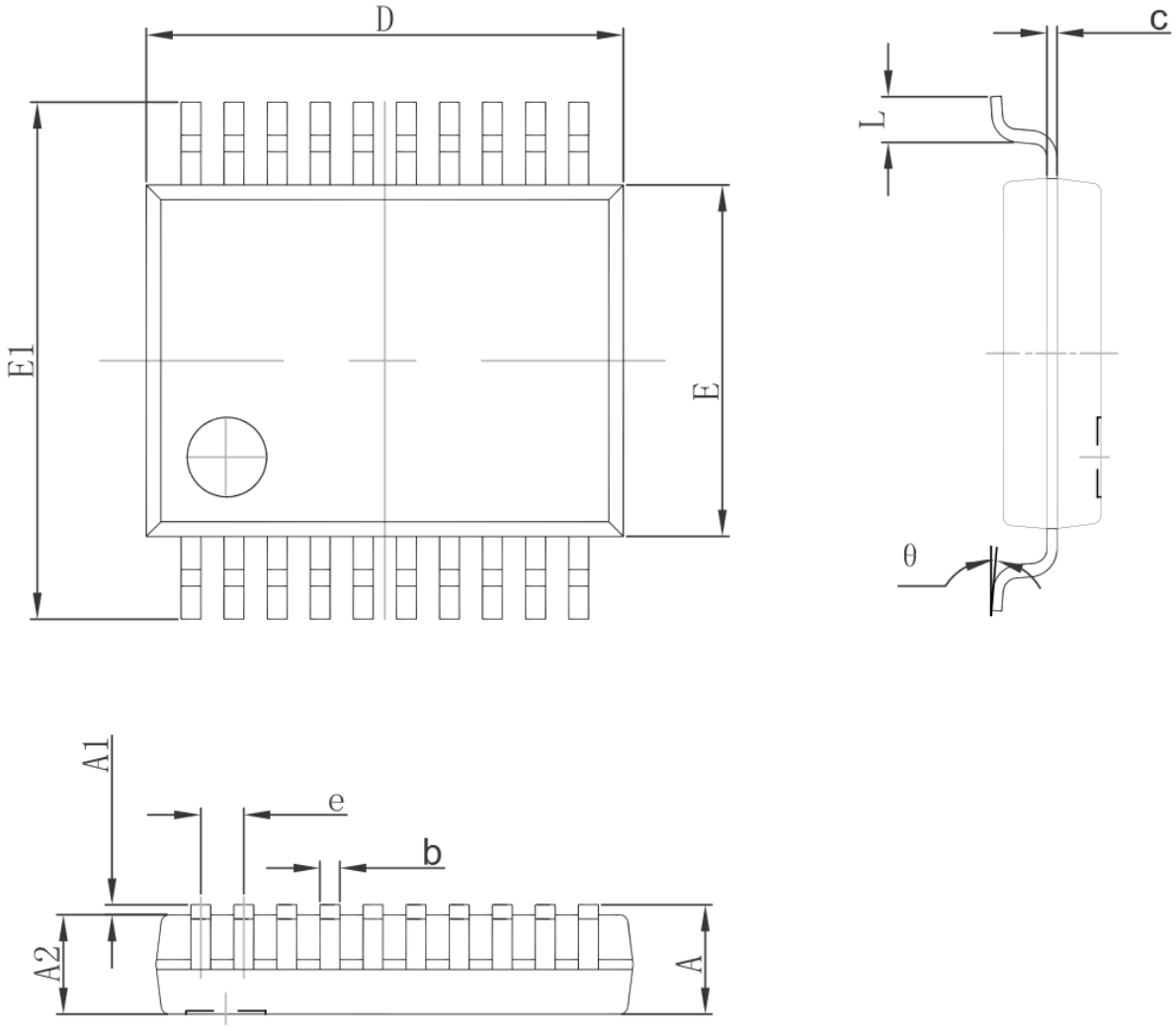
(Unless otherwise specified, TA=25°C, Vcc=5.0V)

Characteristic	Min	Typical	Max	Unit	Comments
Supply Voltage	2.7	5.0	5.5	V	
Working Current		4.9		mA	No LED loaded
Low Level Input Voltage			1.4	V	
High Level Input Voltage	3.7			V	VCC=5V
	1.9				VCC=3V
Output Low Level Voltage			0.1	V	Digital interface only
Output High Level Voltage	4.4			V	Digital interface only
Display Scan Period		13.3		mS	5-digit mode
		15.9			6-digit mode

Ordering Information

Ordering number	Packaging	Quantity in each package
BC7275BEC-T	tube	6600
BC7275BEC-RS	tape & reel	1000
BC7275BEC-RL	tape & reel	2000

Packaging 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	0.010
D	7.000	7.400	0.276	0.291
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
θ	0°	8°	0°	8°

Appendix

BC727X SERIES

	BC7275B	BC7276	BC7277	BC7278
Package	SSOP20	SSOP20	SSOP24	SSOP20
Digits Driving	6¾	8 or 16	9	4
Keyboard	no	16-key	16-key	16-key
Type	Common Cathode	Common Anode	Common Cathode	Common Cathode
External Driver	no need	transistor & shift reg	no need	no need
Large Display	no	yes	no	no