4000 CMOS Series

General characteristics

- **Supply:** 3 to 15V, small fluctuations are tolerated.
- **Inputs** have very high impedance (resistance), this is good because it means they will not affect the part of the circuit where they are connected. However, it also means that unconnected inputs can easily pick up electrical noise and rapidly change between high and low states in an unpredictable way. This is likely to make the IC behave erratically and it will significantly increase the supply current. To prevent problems all unused inputs MUST be connected to the supply (either +Vs or 0V), this applies even if that part of the IC is not being used in the circuit!
- **Outputs** can sink and source only about 1mA if you wish to maintain the correct output voltage to drive CMOS inputs. If there is no need to drive any inputs the maximum current is about 5mA with a 6V supply, or 10mA with a 9V supply (just enough to light an LED). To switch larger currents you can connect a transistor.
- **Fan-out:** one output can drive up to 50 inputs.
- **Gate propagation time**: typically 30ns for a signal to travel through a gate with a 9V supply, it takes a longer time at lower supply voltages.
- **Frequency**: up to 1MHz, above that the 74 series is a better choice.
- **Power consumption** (of the IC itself) is very low, a few µW. It is much greater at high frequencies, a few mW at 1MHz for example.

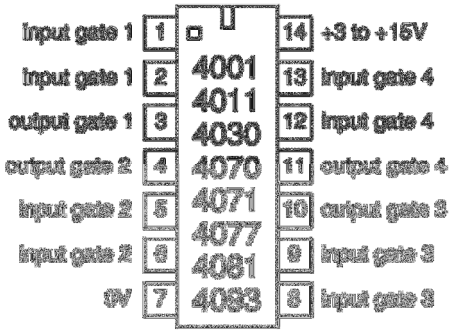
There are many ICs in the 4000 series and this page only covers a selection, concentrating on the most useful gates, counters, decoders and display drivers. For each IC there is a diagram showing the pin arrangement and brief notes explain the function of the pins where necessary. The notes also explain if the IC's properties differ substantially from the standard characteristics listed above.

Static precautions

The CMOS circuitry means that 4000 series ICs are static sensitive. Touching a pin while charged with static electricity (from your clothes for example) may damage the IC. In fact most ICs in regular use are quite tolerant and earthing your hands by touching a metal water pipe or window frame before handling them will be adequate. ICs should be left in their protective packaging until you are ready to use them.

Gates

Quad 2-input gates



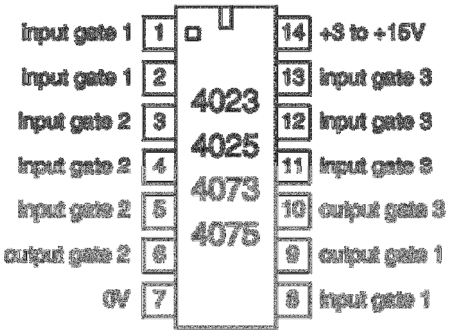
- 4001 quad 2-input NOR
- 4011 quad 2-input NAND
- 4030 quad 2-input EX-OR (now obsolete)
- 4070 quad 2-input EX-OR
- 4071 quad 2-input OR
- 4077 quad 2-input EX-NOR
- 4081 quad 2-input AND
- 4093 quad 2-input NAND with Schmitt trigger inputs

The 4093 has Schmitt trigger inputs to provide good noise immunity. They are ideal for slowly changing or noisy signals. The hysteresis is about 0.5V with 4.5V supply and 2V with a 9V supply.

Triple 3-input gates

- 4023 triple 3-input NAND
- 4025 triple 3-input NOR
- 4073 triple 3-input AND
- 4075 triple 3-input OR

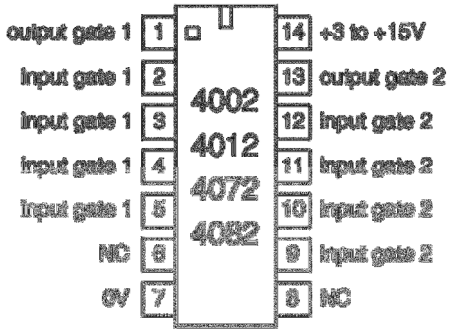
Notice how gate 1 is spread across the two ends of the package.



Dual 4-input gates

- 4002 dual 4-input NOR
- 4012 dual 4-input NAND
- 4072 dual 4-input OR
- 4082 dual 4-input AND

NC = No Connection (a pin that is not used).

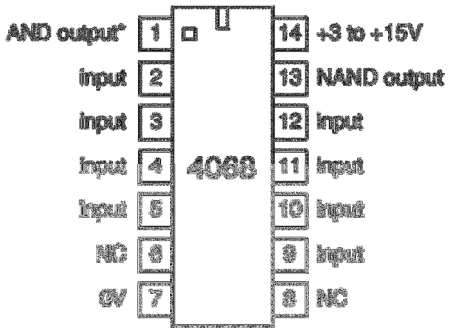


4068 8-input NAND/AND* gate

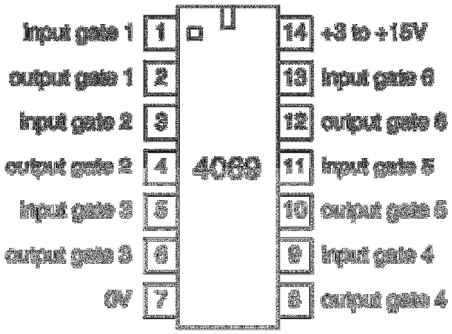
This gate has a propagation time which is about 10 times longer than normal so it is not suitable for high speed circuits.

NC = No Connection (a pin that is not used).

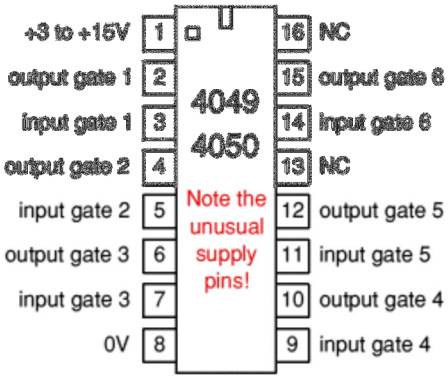
* = The AND output (pin 1) is not available on some versions of the 4068.



4069 hex NOT (inverting buffer)



4049 hex NOT and 4050 hex buffer



- 4049 hex NOT (inverting buffer)
- 4050 hex non-inverting buffer

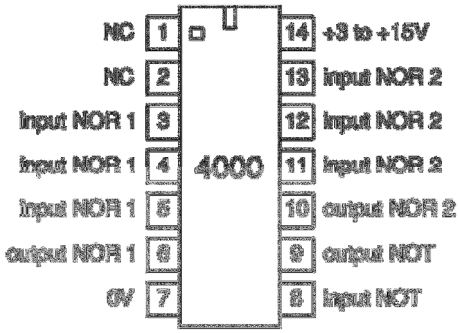
Inputs: These ICs are unusual because their gate inputs can withstand up to +15V even if the power supply is a lower voltage.

Outputs: These ICs are unusual because they are capable of driving 74LS gate inputs directly. To do this they must have a +5V supply (74LS supply voltage). The gate output is sufficient to drive four 74LS inputs.

NC = No Connection (a pin that is not used).

Note the unusual arrangement of the power supply pins for these ICs!

4000 dual 3-input NOR gate and NOT gate

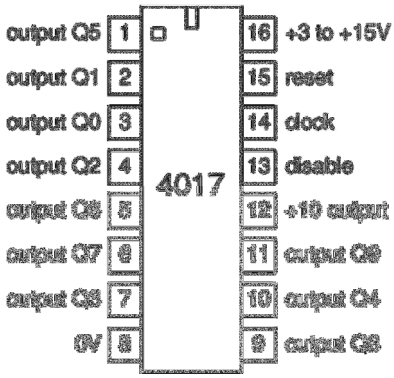


Two 3-input NOR gates and a single NOT gate in one package.

NC = No Connection (a pin that is not used).

Decade and 4-bit Counters

4017 decade counter (1-of-10)



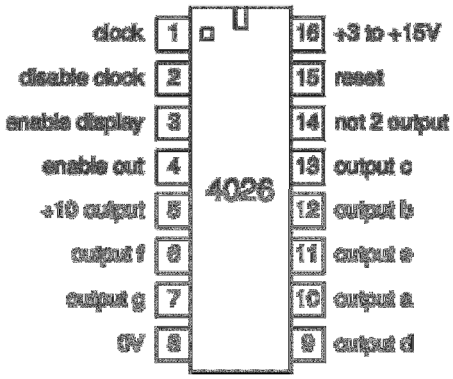
The count advances as the **clock** input becomes high (on the rising-edge). Each **output** Q0-Q9 goes high in turn as counting advances. For some functions (such as flash sequences) outputs may be combined using diodes.

The reset input should be low (0V) for normal operation (counting 0-9). When high it resets the count to zero (Q0 high). This can be done manually with a switch between reset and +Vs and a 10k resistor between reset and 0V. **Counting to less than 9** is achieved by connecting the relevant output (Q0-Q9) to reset, for example to count 0,1,2,3 connect Q4 to reset.

The **disable** input should be low (0V) for normal operation. When high it disables counting so that clock pulses are ignored and the count is kept constant.

The **÷10 output** is high for counts 0-4 and low for 5-9, so it provides an output at $^1/_{10}$ of the clock frequency. It can be used to drive the clock input of another 4017 (to count the tens).

4026 decade counter and 7-segment display driver



The count advances as the **clock** input becomes high (on the rising-edge).
The **outputs a-g** go high to light the appropriate segments of a common-cathode 7-segment display as the count advances. The maximum output current is about 1mA with a 4.5V supply and 4mA with a 9V supply. This is sufficient to directly drive many 7-segment LEDdisplays. The table below shows the segment sequence in detail.

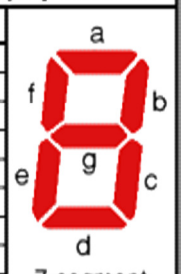
The **reset** input should be low (0V) for normal operation (counting 0-9). When high it resets the count to zero.

The **disable clock** input should be low (0V) for normal operation. When high it disables counting so that clock pulses are ignored and the count is kept constant.

The **enable display** input should be high (+Vs) for normal operation. When low it makes outputs a-g low, giving a blank display.
The **enable out** follows this input but with a brief delay.

Outputs from the 4026 counter and display driver IC

Count	a	b	c	d	e	f	g	h
0	●	●	●	●	●	●		●
1		●	●					●
2	●	●		●	●		●	●
3	●	●	●	●			●	●
4		●	●			●	●	●
5	●		●	●		●	●	
6	●		●	●	●	●	●	
7	●	●	●					
8	●	●	●	●	●	●	●	
9	●	●	●	●		●	●	



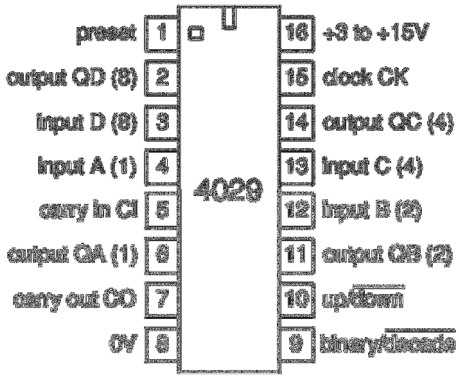
7-segment display

● = segment on. h is used to drive other counters.

The **÷10 output** (h in table) is high for counts 0-4 and low for 5-9, so it provides an output at $^1/_{10}$ of the clock frequency. It can be used to drive the clock input of another 4026 to provide multi-digit counting.

The **not 2 output** is high unless the count is 2 when it goes low.

4029 up/down synchronous counter with preset



The 4029 is a **synchronous** counter so its outputs change precisely together on each clock pulse. This is helpful if you need to connect the outputs to logic gates because it avoids the glitches which occur with ripple counters.

The count occurs as the **clock** input becomes high (on the rising-edge). The **up/down** input determines the direction of counting: high for up, low for down. The state of **up/down** should be changed when the clock is high.

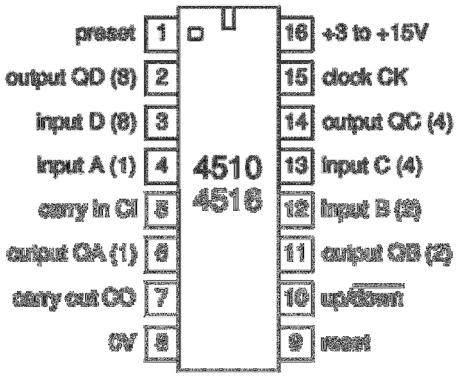
For normal operation (counting) **preset**, and **carry in** should be low.

The **binary/decade** input selects the type of counter: 4-bit binary (0-15) when high; decade (0-9) when low.

The counter may be **preset** by placing the desired binary number on the **inputs A-D** and briefly making the **preset** input high. There is no reset input, but **preset** can be used to reset the count to zero if **inputs A-D** are all low.

4510 up/down decade (0-9) counter with preset
4516 up/down 4-bit (0-15) counter with preset

These are **synchronous** counters so their outputs change precisely together on each clock pulse. This is helpful if you need to connect their outputs to logic gates because it avoids the glitches which occur with ripple counters.



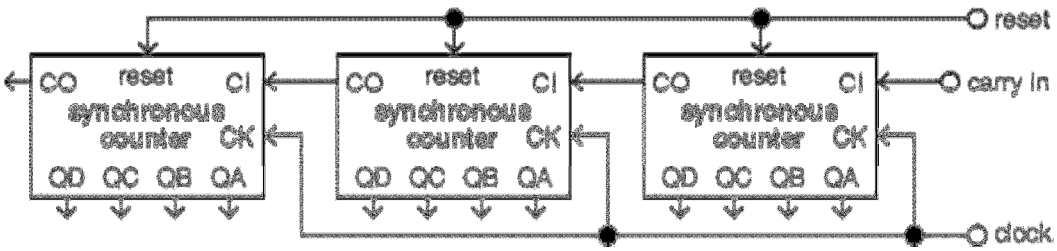
The count occurs as the **clock** input becomes high (on the rising-edge). The **up/down** input determines the direction of counting: high for up, low for down. The state of **up/down** should be changed when the clock is high.

For normal operation (counting) **preset**, **reset** and **carry in** should be low. When **reset** is high it resets the count to zero (0000, QA-QD low). The **clock** input should be low when resetting.

The counter may be **preset** by placing the desired binary number on the **inputs A-D** and briefly making the **preset** input high, the **clock** input should be low when this happens.

Connecting synchronous counters in a chain

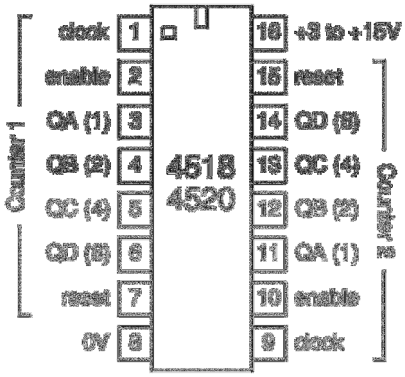
The diagram below shows how to link synchronous counters, notice how all the **clock (CK)** inputs are linked. **Carry out (CO)** feeds **carry in (CI)** of the next counter. **Carry in (CI)** of the first counter should be low for 4029, 4510 and 4516 counters.



4518 dual decade (0-9) counter
4520 dual 4-bit (0-15) counter

These contain two separate **synchronous** counters, one on each side of the IC.

Normally a clock signal is connected to the **clock** input, with the **enable** input held high. Counting advances as the clock signal becomes high (on the rising-edge). Special arrangements are used if the 4518/20 counters are linked in a chain, as explained below.

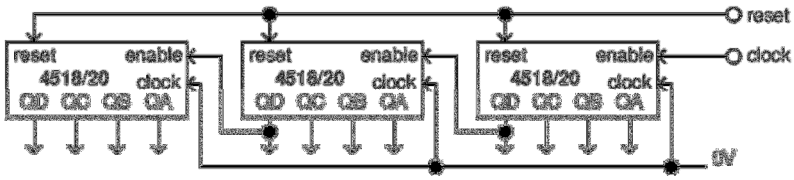


For normal operation the **reset** input should be low, making it high resets the counter to zero (0000, QA-QD low).

Counting to less than the maximum (9 or 15) can be achieved by connecting the appropriate output(s) to the reset input, using an AND gate if necessary. For example to count 0 to 8 connect QA (1) and QD (8) to reset using an AND gate.

Connecting 4518 and 4520 counters in a chain

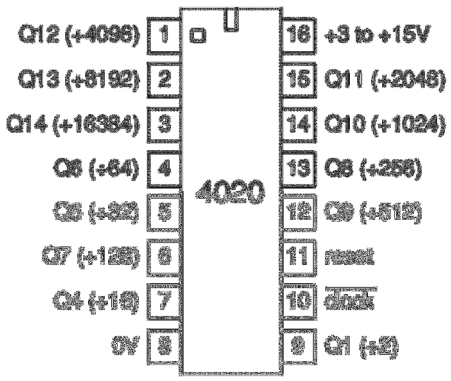
The diagram below shows how to link 4518 and 4520 counters. Notice how the normal **clock** inputs are held low, with the **enable** inputs being used instead. With this arrangement counting advances as the enable input becomes low (on the falling-edge) allowing **output QD** to supply a clock signal to the next counter. The complete chain is a ripple counter, although the individual counters are synchronous! If it is essential to have truly synchronous counting a system of logic gates is required, please see a 4518/20 datasheet for further details.



7-bit, 12-bit and 14-bit counters

4020 14-bit (÷16,384) ripple counter

The 4020 is a **ripple** counter so beware that glitches may occur in any logic gate systems connected to its outputs due to the slight delay before the later counter outputs respond to a clock pulse.

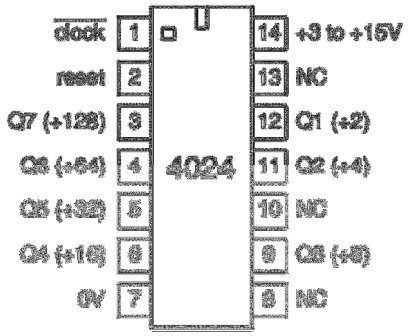


The count advances as the **clock** input becomes low (on the falling-edge), this is indicated by the bar over the clock label. This is the usual clock behaviour of ripple counters and it means a counter output can directly drive the clock input of the next counter in a chain.

Output Qn is the nth stage of the counter, representing 2^n, for example Q4 is $2^4 = 16$ ($^1/_{16}$ of clock frequency) and Q14 is $2^{14} = 16384$ ($^1/_{16384}$ of clock frequency). Note that Q2 and Q3 are not available.

The **reset** input should be low for normal operation (counting). When high it resets the count to zero (all outputs low).

4024 7-bit (÷128) ripple counter



The 4024 is a **ripple** counter so beware that glitches may occur in any logic gate systems connected to its outputs due to the slight delay before the later counter outputs respond to a clock pulse.

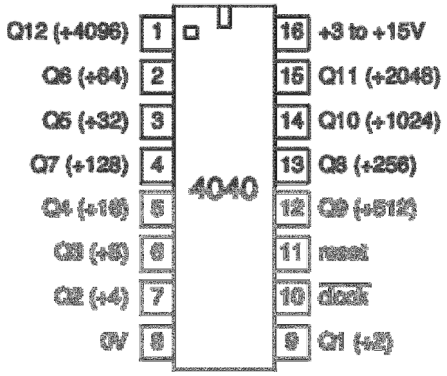
The count advances as the **clock** input becomes low (on the falling-edge), this is indicated by the bar over the clock label. This is the usual clock behaviour of ripple counters and it means a counter output can directly drive the clock input of the next counter in a chain.

Output Qn is the nth stage of the counter, representing 2^n, for example Q4 is $2^4 = 16$ ($^1/_{16}$ of clock frequency) and Q7 is $2^7 = 128$ ($^1/_{128}$ of clock frequency).

The **reset** input should be low for normal operation (counting). When high it resets the count to zero (all outputs low).

4040 12-bit (÷4096) ripple counter

The 4040 is a **ripple** counter so beware that glitches may occur in any logic gate systems connected to its outputs due to the slight delay before the later counter outputs respond to a clock pulse.

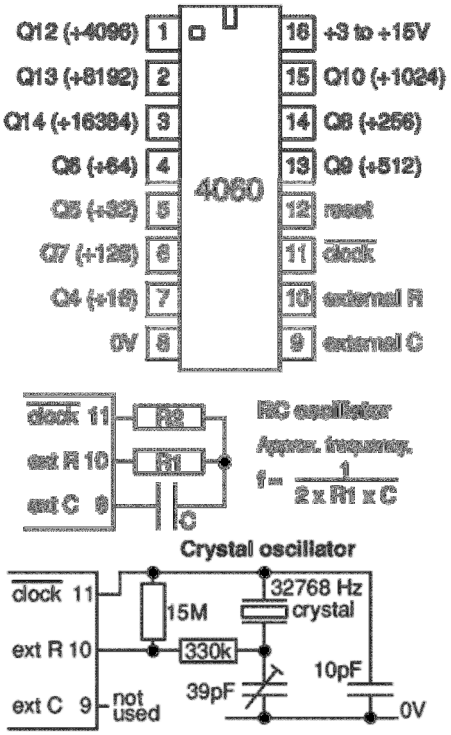


The count advances as the **clock** input becomes low (on the falling-edge), this is indicated by the bar over the clock label. This is the usual clock behaviour of ripple counters and it means a counter output can directly drive the clock input of the next counter in a chain.

Output Qn is the nth stage of the counter, representing 2^n, for example Q4 is $2^4 = 16$ ($^1/_{16}$ of clock frequency) and Q12 is $2^{12} = 4096$ ($^1/_{4096}$ of clock frequency).

The **reset** input should be low for normal operation (counting). When high it resets the count to zero (all outputs low).

4060 14-bit (÷16,384) ripple counter with internal oscillator



The 4060 is a **ripple** counter so beware that glitches may occur in any logic gate systems connected to its outputs due to the slight delay before the later counter outputs respond to a clock pulse.

The count advances as the **clock** input becomes low (on the falling-edge), this is indicated by the bar over the clock label. This is the usual clock behaviour of ripple counters and it means a counter output can directly drive the clock input of the next counter in a chain. The clock can be driven directly, or connected to the **internal oscillator** (see below).

Output Qn is the nth stage of the counter, representing 2^n, for example Q4 is $2^4 = 16$ ($^1/_{16}$ of clock frequency) and Q14 is $2^{14} = 16384$ ($^1/_{16384}$ of clock frequency). Note that Q1-3 and

Q11 are not available.

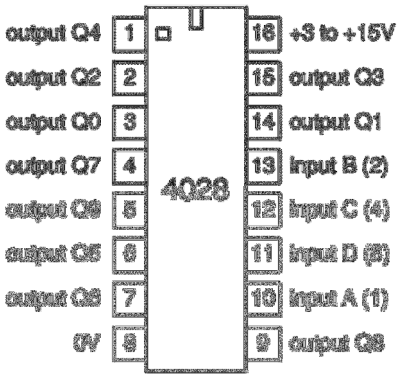
The **reset** input should be low for normal operation (counting). When high it resets the count to zero (all outputs low).

The 4060 includes an **internal oscillator**. The **clock** signal may be supplied in three ways:

- **From an external source** to the clock input, as for a normal counter. In this case there should be no connections to **external C** and **external R** (pins 9 and 10).
- **RC oscillator** as shown in the diagram. The oscillator drives the clock input with an approximate frequency f = $^1/_{(2\times R1\times C)}$ (it partly depends on the supply voltage). R1 should be at least 50k if the supply voltage is less than 7V. R2 should be between 2 and 10 times R1.
- **Crystal oscillator** as shown in the diagram, note that there is no connection to pin 9. The 32768 Hz crystal will give a 2Hz signal at the last output, Q14.

Decoders

4028 BCD to decimal (1 of 10) decoder



The appropriate **output Q0-9** becomes high in response to the BCD (binary coded decimal) input. For example an input of binary 0101 (=5) will make output Q5 high and all other outputs low.

The 4028 is a BCD (binary coded decimal) decoder intended for input values 0 to 9 (0000 to 1001 in binary). With inputs from 10 to 15 (1010 to 1111 in binary) all outputs are low.

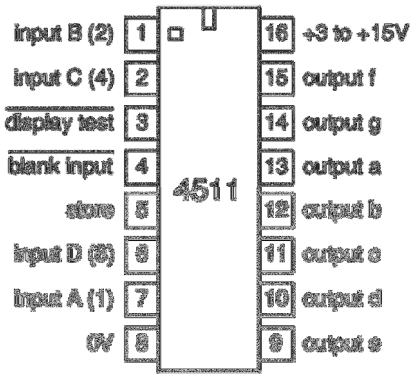
Note that the 4028 can be used as a **1-of-8 decoder** if input D is held low.

7-segment Display Drivers

4511 BCD to 7-segment display driver

The appropriate **outputs a-g** become high to display the BCD (binary coded decimal) number supplied on **inputs A-D**. The **outputs a-g** can source up to 25mA. The 7-segment display segments must be connected between the outputs and 0V with a resistor in series (330R with a 5V supply). A **common cathode** display is required.

Display test and **blank input** are active-low so they should be high for normal operation. When **display test** is low all the display segments should light (showing number 8). When **blank input** is low the display will be blank (all segments off).



The **store** input should be low for normal operation. When **store** is high the displayed number is stored internally to give a constant display regardless of any changes which may occur to the **inputs A-D**.

The 4511 is intended for BCD (binary coded decimal). Inputs values from 10 to 15 (1010 to 1111 in binary) will give a blank display (all segments off).