



SEQUENTIAL LOGIC

CP9945

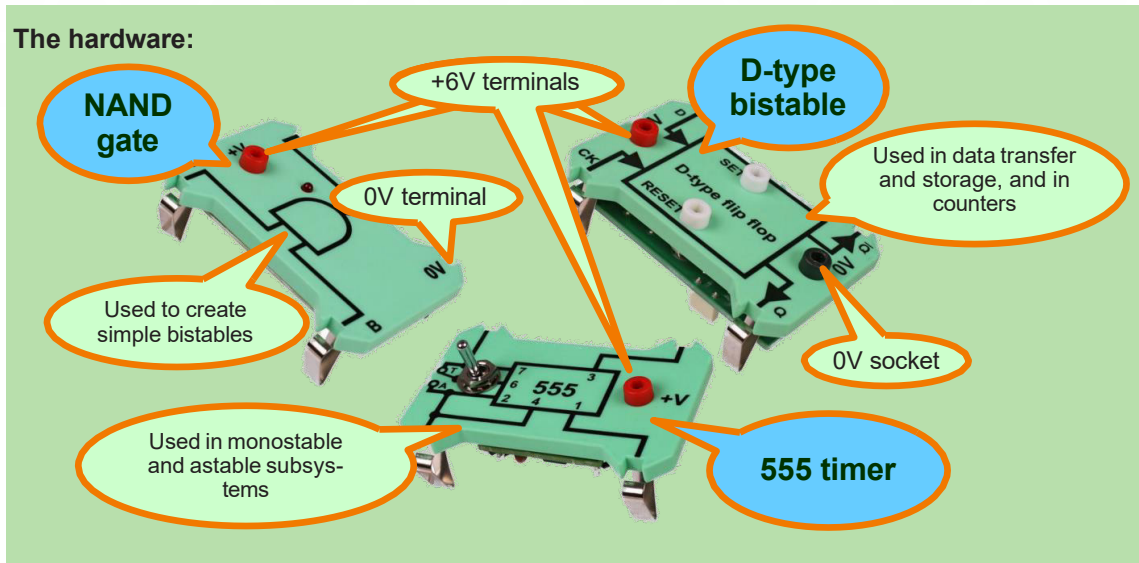
Inspiring The Next
Generation Of Engineers

www.matrixtsl.com

Contents

Introduction	3
Worksheets	5
Worksheet 1 - The astable	6
Worksheet 2 - The monostable	8
Worksheet 3 - The simple bistable	10
Worksheet 4 - The D-type bistable	12
Worksheet 5 - The 1-bit counter	14
Worksheet 6 - Debouncing with a monostable	16
Worksheet 7 - Debouncing with a bistable	18
Worksheet 7A - The bistable debouncer	20
Worksheet 8 - The 3-bit counter	21
Worksheet 9 - The 'modulo-n' counter	23
Worksheet 10 - The 3-stage shift register	25
Worksheet 11 - The R-2R DAC	27

Introduction



Sequential logic

- In combinational logic, the output depends solely on the state of the inputs at the time.
- For example:
 - An AND gate outputs a logic 1 signal only while both inputs are at logic 1.
 - A decoder selects the third subsystem when the input signal is 10.
- In sequential logic, as well as the current state of the inputs, the output depends on the sequence of states that led up to it.
- For example:
 - When both inputs are at logic 1, a bistable may output logic 0, or output logic 1, depending on the previous state of the inputs.
 - A counter may have one of a wide range of outputs when its (clock) input goes to logic 1.
- An astable subsystem:
 - Has no stable output states;
 - Changes the output signal from logic 0 to logic 1 automatically after a set time, and then changes back to logic 0 automatically after a further set time.
- A monostable subsystem:
 - Has one stable output state;
 - Changes the output signal, from logic 0 to logic 1, say, when triggered by an input signal, and then changes back automatically some time later.
- A bistable subsystem:
 - Has two stable output states;
 - Changes the output signal from one to the other when triggered, and then changes back when triggered again.

Introduction

The background:

A monostable sends out a single square pulse when triggered. It is sometimes called a 'one-shot' device. The duration of the pulse depends on component values inside the subsystem, usually a capacitor and a resistor (RC network.)

An **astable** sends out a continuous stream of pulses. Here again, pulse duration depends on components, usually a RC network, inside the subsystem. Both monostable and astable subsystems can be built using a 555 timer chip.

A **bistable** can be used to create a latch, a subsystem that holds the output steady, once triggered, until it receives a second signal, on its Reset input. It is also the basis of a counter. The 'D-type' and the 'J-K' are both bistables.

A **counter** counts square pulses received at its clock input.

A common format is the 4-bit counter.

The Reset input is used to return the count to zero.

Counting in binary:

Most electronic counters count using the binary number system. This uses only two digits, **0 and 1**. (By comparison, the decimal system uses ten - **0, 1, 2, 3, 4, 5, 6, 7, 8 and 9**.)

The table compares these two counting systems.

Notice that the equivalent number in binary contains many more digits. We find it cumbersome to use, but electronic counters work so quickly that they can carry out calculations rapidly using binary.

Decimal	Binary	Decimal	Binary
0	0	8	1000
1	1	9	1001
2	10	10	1010
3	11	11	1011
4	100	12	1100
5	101	13	1101
6	110	14	1110
7	111	15	1111

For your records:

Answer the following questions:

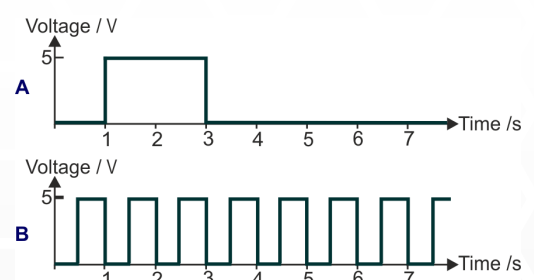
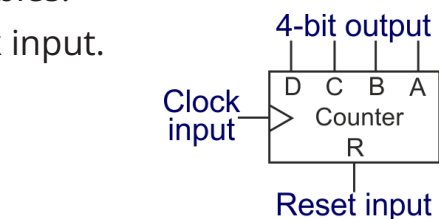
- What is the difference between combinational logic and sequential logic?
- Which two of the following are sequential systems?
OR gate astable bistable counter
- What is the difference between astables, monostables and bistables?

The graphs show two signals, labelled A and B.

Copy them and complete the statements:

Signal shows the output of a monostable system.

Signal shows the output of an astable system.





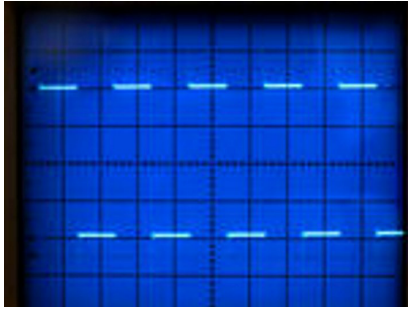
WORKSHEETS



WORKSHEETS

Worksheet 1 - The Astable

The astable is the workhorse of sequential logic systems.



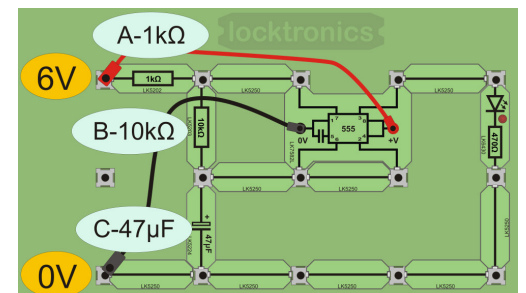
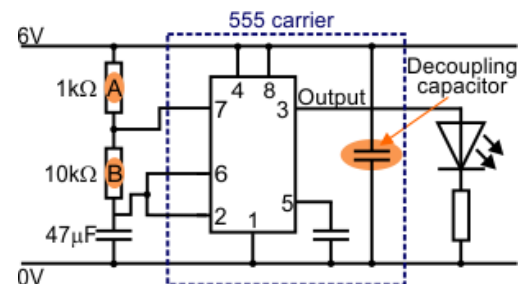
Also known as 'oscillator', 'astable multivibrator', 'flip-flop' or simply 'clock', it generates a series of square pulses used to synchronise the operation of many sequential systems.

The simplest way to create an astable is to use a '555 timer' chip, first manufactured in the 1970's. The Locktronics 555 carrier can be switched between astable and monostable modes.

Over to you:

For astable operation, the 555 carrier requires the addition of three 'timing' components - two resistors and a capacitor, as shown in the circuit diagram. These control the frequency of the output pulses, and the 'mark:space', as explained later on.

- Build the circuit shown opposite.
- Be sure to connect the 555 carrier to the +6V supply, via the '+V' and '0V' sockets.
- Switch on the 6V power supply.
- The LED connected to the output should be flashing on and off. Time how long it takes to produce ten flashes (i.e. ten 'on-and-off's.)
- Enter your measurement in the first row of the table.
- Switch off the power supply. Change the value of resistor 'B' from 10k Ω to 5k Ω .
- Switch on, and time how long it takes to produce ten flashes now. Enter the result in the second row.
- Repeat this procedure for the other combinations of timing components given in the table.



Timing Components			Time for ten flashes
Resistor A	Resistor B	Capacitor	
1k Ω	10k Ω	47 μ f	
1k Ω	5k Ω	47 μ f	
1k Ω	5k Ω	100 μ f	
1k Ω	50k Ω	100 μ f	

(The circuit diagram for the astable includes a decoupling capacitor. This decouples' (isolates) the power supply from the oscillations generated by the 555 chip. Without it, small oscillations could appear on the power rails and these could affect other subsystems in the system.)

WORKSHEETS

Worksheet 1 - The Astable

So What?

Square wave vocabulary:

Amplitude - The maximum voltage in the signal.

Period - The time taken to produce one cycle of the square wave, (i.e. 1 peak plus 1 trough)
- Measured in seconds.

Frequency - The number of cycles of the square wave produced per second;
- Measured in hertz. (1 Hz means one cycle produced every second.)

These are related by the relationship: **Frequency = 1 / period**

Mark - the time spent at the peak of the signal.

Space - the time spent at 0V.

Mark:Space - the time the signal sits at its peak, compared with the time at 0V.

- Use your measurements, and the information above, to complete the period and frequency columns in the table.
- From a theoretical analysis, the period T is given by the formula:

$$T = 0.69(R_A + 2 \times R_B) \times C.$$

This formula, and the one above have been used to give calculated values for the first and third sets of timing components. Do the calculations for the other two sets.

Timing Components			Measured Period in s	Measured Frequency in Hz	Calculated Period in s	Measured Frequency in Hz
Resistor A	Resistor B	Capacitor				
1kΩ	10kΩ	47μf			0.68	1.47
1kΩ	5kΩ	47μf				
1kΩ	5kΩ	100μf			0.76	1.32
1kΩ	50kΩ	100μf				

A challenge -

When $R_B = 100k\Omega$ and $C = 100\mu F$, change R_A to $100k\Omega$.

What effect does this have on the output?

For your records:

- Copy the circuit diagram for the 555 astable, given near the top of the previous page.
- Copy the signal shown below, and give the information requested.

(a) Amplitude =

(b) Period =

(c) Frequency =

(d) Mark:space =



Calculate the period and frequency for this astable when the following timing component values are used:

$$R_A = 5k\Omega$$

$$R_B = 100k\Omega$$

$$C = 22\mu F$$

WORKSHEETS

Worksheet 2 - The Monostable



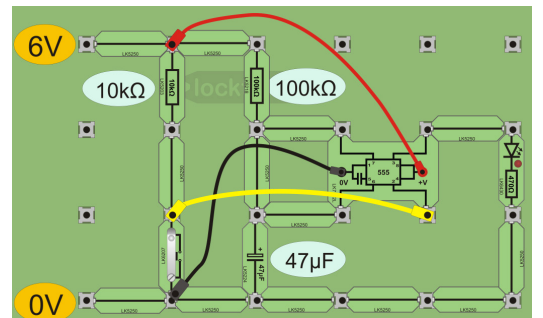
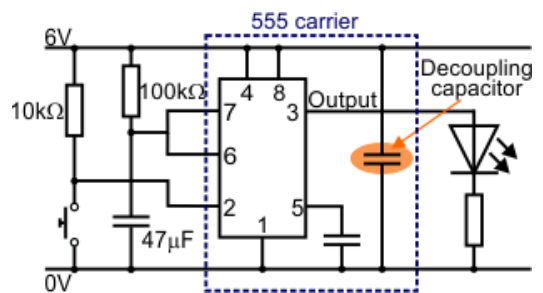
The monostable subsystem has one stable output state. It remains in it until triggered by an external signal. It then jumps into the other output state. (It is digital, and so has only two possible states.) After a time determined by the timing components, a resistor and a capacitor, it reverts automatically to its stable state.

One use is as a time-delay subsystem, used to keep security lights on for a short time once triggered (by a PIR sensor, for example.)

Over to you:

For monostable operation, the 555 carrier requires just two 'timing' components - a resistor and a capacitor, as shown in the circuit diagram.

- Build the circuit shown opposite.
- Notice the orientation of the switch unit - the 10kΩ resistor is in the 'pull-up' position.
- Be sure to connect the 555 carrier to the +6V supply, via the '+V' and '0V' sockets.
- Switch on the 6V power supply.
- Press and release the push-switch. The LED connected to the output should come on for a short time and then go off.
- Press it again, and time how long the LED stays lit.
- Enter your measurement in the first row of the table.
- Switch off the power supply.
- Replace the 10kΩ resistor with a 50kΩ resistor.
- Time how long the LED stays on now, and enter the result in the second row of the table.
- Repeat this procedure for the other combinations of components given in the table.



Timing Components		Time LED stays lit in seconds
Resistor	Capacitor	
100kΩ	47μf	
50kΩ	47μf	
50kΩ	100μf	
100kΩ	100μf	

WORKSHEETS

Worksheet 2 - The Monostable

So What?

The relationship between the time delay, T , and the timing components is:

$$T = 1.1 \times R \times C$$

Here are the options when using it:

- With R in ohms, and C in farads, T will be in seconds.
- With R in megohms ($M\Omega$), and C in microfarads (μF), T will be in seconds.
- With R in kilohms ($k\Omega$), and C in microfarads (μF), T will be in milliseconds (ms).

Remember: $1k\Omega = 1\,000\Omega$

$1M\Omega = 1\,000\,000\Omega$

$1F = 1\,000\,000\mu F$

- Using this information, complete the table by calculating the theoretical values of T .

Timing Components		1.1xRx C
Resistor	Capacitor	
100k Ω	47 μf	
56k Ω	47 μf	
56k Ω	100 μf	
100k Ω	100 μf	

- Compare your results with these theoretical values. Discuss with your colleagues reasons for differences between the theoretical and actual measured values for the time delays.

A challenge - a variation:

Sometimes, a delayed turn on subsystem is needed. In other words, you press the switch, there is a delay and then the device switches on. There are several ways to implement this.

Design a system that does this.

Ask your instructor for permission to build and test your solution.

For your records:

- Copy the circuit diagram for the 555 monostable, given on the previous page.
- Copy the formula linking time delay with the values of the timing components.
- Copy and complete these statements about the performance of the monostable subsystem:

When the value of the resistor is increased, the time delay produced

When the value of the capacitor is increased, the time delay produced

- The formula given above can be re-arranged as:

$$R = T / 1.1 \times C$$

Use this formula to decide what value resistor to use with a 100 μF capacitor to produce a time delay of 11 seconds.

WORKSHEETS

Worksheet 3 - The Simple Bistable



Bistable systems form the basis of electronic memory. Each memory 'cell' is effectively a bistable, storing a single binary digit, either a **0** or a **1**.

The memory modules in the picture each store 8GB of data, which is roughly $8 \times 8 \times 1000 \times 1000000$ **0**'s or **1**'s.

That's a lot of bistables!

Over to you:

The significant element in sequential circuits is the use of feedback, where the output signal, or part of it, is sent back to the input.

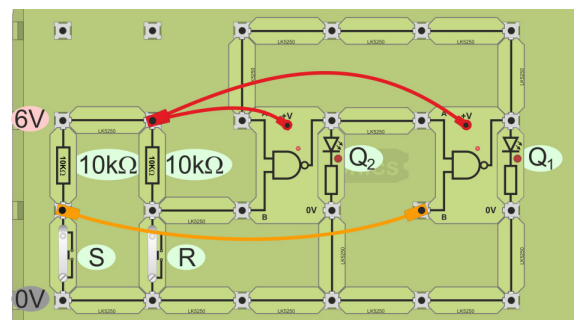
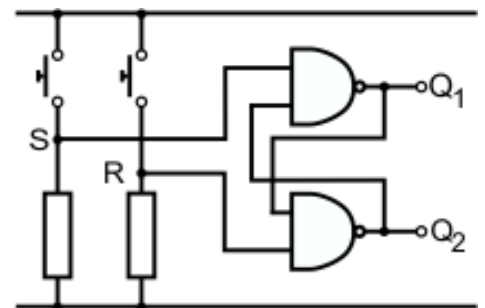
Here, output Q_1 is connected to an input on the lower NAND gate, and Q_2 to an input on the upper gate.

The state of each output is shown on the LED carrier attached to it. When the LED is lit, the output is logic 1, when not lit, logic 0.

- Build the circuit shown opposite.
- Notice the orientation of the switch units - they use pull-up resistors to keep the inputs of the gates at logic 1 until the switch is pressed.

When a switch is pressed, the switch unit sends a logic 0 signal to the gate input. (You can show this by adding LED carriers between the 'B' inputs and 0V.)

- Switch on the 6V power supply. The system starts in any state. Complete the first row of the table with the initial states of Q_1 and Q_2 (before any switches are pressed.)
- Now press switch 'S', to input a logic 0 signal. Use the LEDs to identify the logic states of outputs Q_1 and Q_2 and record them in the second row of the table.
- Release switch 'S', so that both 'S' and 'R' inputs are at logic 1. Record the outputs states in the third row.
- Continue in this way to complete the table.



S(Set)	R(Reset)	Q_1	Q_2
1	1		
0	1		
1	1		
1	0		
1	1		
0	1		
1	1		
1	0		
0	0		

WORKSHEETS

Worksheet 3 - The Simple Bistable

So What?

- Rows 1,3,5 and 7 - Setting both inputs to logic 1 does not fix the state of the outputs! Sometimes $Q_1 = 0$, $Q_2 = 1$ and sometimes it is the other way round. It depends on what happened previously - it just preserves the previous states of the outputs.
- Rows 2 and 6 - Whenever, 'S' is logic 0, Q_1 is logic 1 (set). Hence the label 'S' for the switch.
- Rows 4 and 8 - Whenever, 'R' is logic 0, Q_1 is logic 0 (reset). Hence the other switch is 'R'.

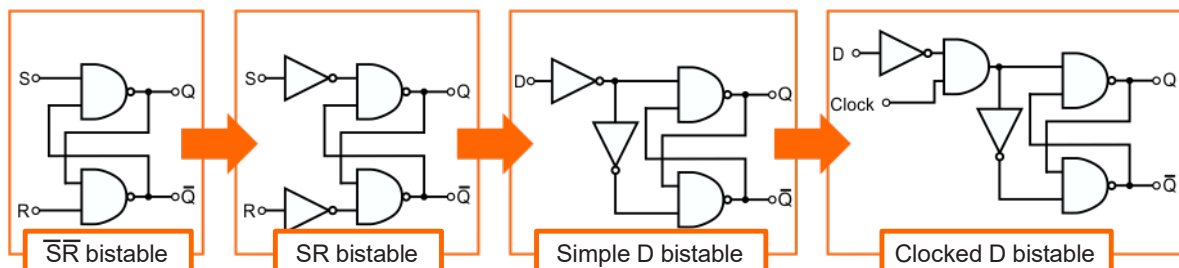
So far everything is civilised - Q_1 and Q_2 sit in opposite states.

As a result, the outputs are normally known as **Q and $\bar{Q}$** . The inputs are 'active low', meaning that they cause changes when they are 'low', logic 0, and so this is known as a **$\bar{S}\bar{R}$ bistable**.

A problem - row 9! The outputs are not opposites - both are at logic 1. The new names fail!

The shortcomings of the basic bistable are overcome through a series of modifications.

1. Illogical! - to make the output logic 1, input a logic 0! Add NOT gates to each input, making them 'active high'. ($S = 1$, $R = 1$ is now the forbidden combination.) This is now the **SR bistable**.
2. There is still a forbidden combination! Connect a NOT gate between the 'S' and 'R' inputs. Now the two inputs cannot be the same. We have only one input, however, the D (data) input. It is the **simple D bistable**. Whenever $D = 1$, $Q = 1$ ($\bar{Q} = 0$). Whenever $D = 0$, $Q = 0$ ($\bar{Q} = 1$).
3. The outputs are not protected. Whenever the input changes, the output changes! Add an AND gate. The second input is called the clock input. The outputs can change now only when the clock input is raised to logic 1. This is the **clocked D bistable**.



This clocked D bistable is 'level-triggered' - the output can change whenever the clock input is at logic 1. The commercial D-type bistable has 'edge triggering' added. This is the focus of worksheet 4.

For your records:

- Copy the circuit diagram for the **$\bar{S}\bar{R}$ bistable**, given near the top of the previous page.
- Why are the outputs known as **Q and $\bar{Q}$** ? Why are the inputs called **$\bar{S}$ and $\bar{R}$** ?
- Describe three problems associated with this bistable, and a modification that will cure each. Include the circuit diagram for each modification with your explanation.

WORKSHEETS

Worksheet 4 - The D-Type Bistable



One use of a D-type bistable is to create an electronic latch - a subsystem which, once triggered, stays on until a reset signal is received.

An alarm, such as a car alarm, uses this approach.

When someone breaks in, the alarm sounds until the car owner turns it off, often by pressing a switch on the remote control fob.

This worksheet looks at the behaviour of a D-type bistable, including its ability to act as a latch.

Over to you:

1 - D-type principles:

- Set up the circuit shown opposite. We are not using the 'Set' and 'Reset' inputs here, so they are connected to 0V.
- Switch on the 6V power supply. If the LED is lit, press and release the 'Ck' switch.
- Now work through the sequence that follows, completing the table with the state of the LED each time:

- Press and hold down the 'D' switch.
- Now press, hold down and release the 'Ck' switch.
- Release the 'D' switch.
- Press, hold down and release the 'Ck' switch again.

Step	'D' switch	'Ck' switch	LED
(a)	On	Off	
(b)	On	On / Off	
(c)	Off	Off	
(d)	Off	On / Off	

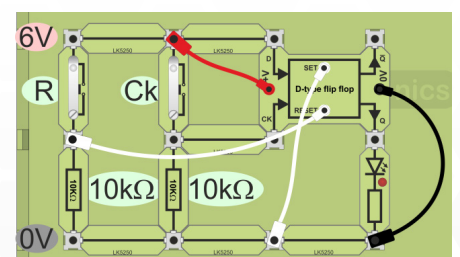
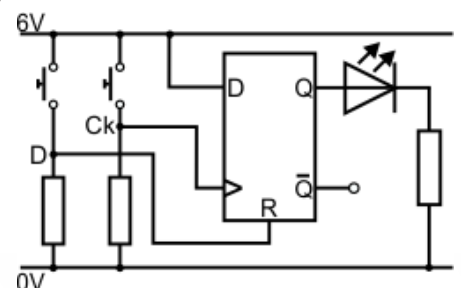
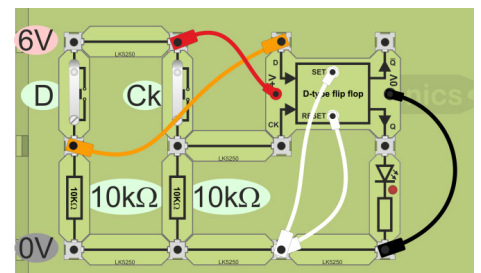
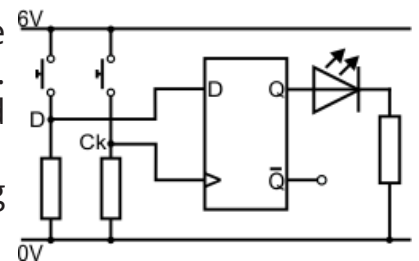
- With the circuit in front of you, answer Q1 in the 'For your records' section.

2 - The latch:

- Modify the circuit as shown.
- The 'D' input is connected directly to +6V.
- The left-hand switch unit controls the 'Reset' input on the D-type bistable carrier.
- 'Set' is still not used, and remains connected to 0V.

- Observe what happens when you:
 - Press and then release the 'Ck' switch.
 - Press and release the 'R' switch.

- Use these observations to answer Q2 in the 'For your records' section.



WORKSHEETS

Worksheet 4 - The D-Type Bistable

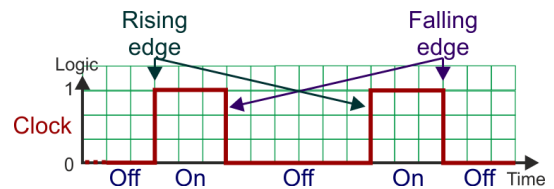
So What?

1 - D-type principles:

Switch behaviour:

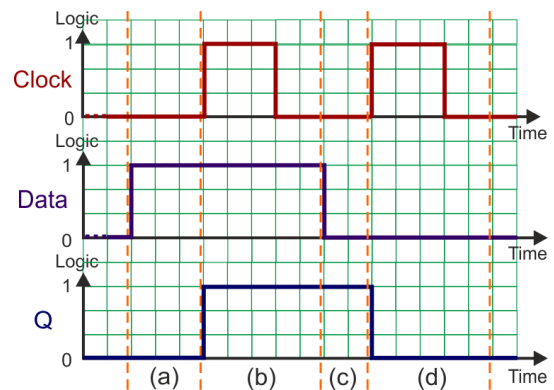
- Switch off - the 'pull-down' resistor, between the input and 0V, holds the input at logic 0.
- Switch on - the input is connected to +6V, inputting a logic 1 signal.

This behaviour is shown in the first timing diagram.



D-type bistable behaviour:

- The bistable is 'rising-edge' triggered. (The first timing diagram illustrates the meaning of 'rising-edge' and 'falling-edge'.)
- On the rising-edge of a clock pulse, the Q output of the D-type copies the logic level present on the D input of the bistable.

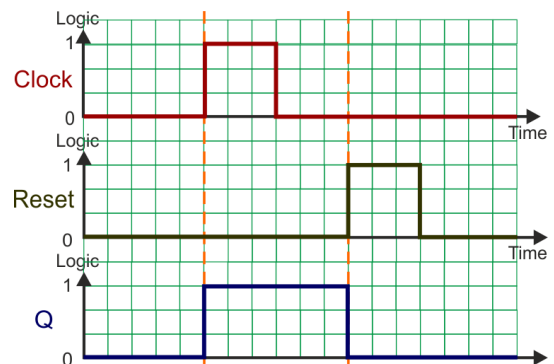


The timing diagram opposite is one way to express your observations from parts (a) to (d) earlier. Check that you understand and agree with it!

2 - D-type latch behaviour:

Now, the Data input ('D') is permanently connected to logic 1, (+6V).

- On the first rising-edge of the clock pulse, the Q output copies it and goes to logic 1.
- When the Reset input goes high, however, the Q output is reset to logic 0.



A challenge -

Add a second LED carrier to show the state of the Q output in the two circuits.

Add a new graph to the two timing diagrams to show your results.

For your records:

- Answer the following questions:
 1. In the first circuit, what is the purpose of the 'D' switch and of the 'Ck' switch?
 2. What is the meaning of 'latch'? (Use your observations of the second circuit in your answer.)
- Copy the three timing diagrams. Add labels to each to explain what is happening at different stages.

WORKSHEETS

Worksheet 5 - The 1-Bit Counter



Electronic counting systems have countless uses in our lives. They count people entering lifts, or train carriages, count eggs in a packing factory, count coins in a bank, pills dispensed to pharmacies and even count votes in political elections.

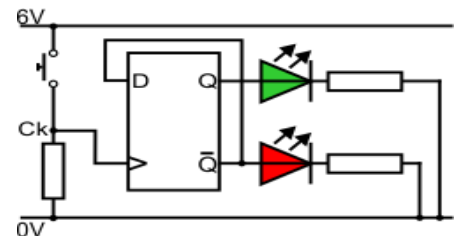
They count using the binary number system, discussed in the Introduction, but may display the result in decimal numbers, more understandable to humans.

This worksheet looks at the basic unit of counters, the 1-bit counter.

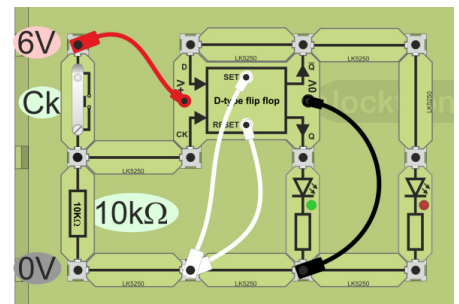
Over to you:

1 - One-bit counter

- Set up the circuit shown. Notice that the Data input is connected to the $\bar{Q}$ output. A second LED carrier shows the state of the $\bar{Q}$ output.
- Switch on the 6V power supply. If the LED is lit, press the 'Ck' switch once.
- Press and release the 'Ck' switch a number of times, to send in clock pulses and observe the effect on the LED.
- Record your results in the table that follows.

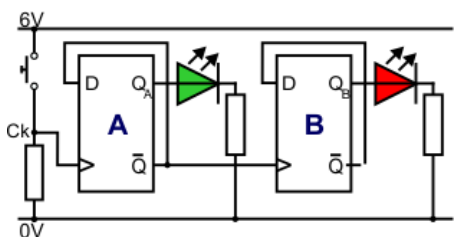


Clock pulse	State of LED
0	Off
1	
2	
3	
4	
5	

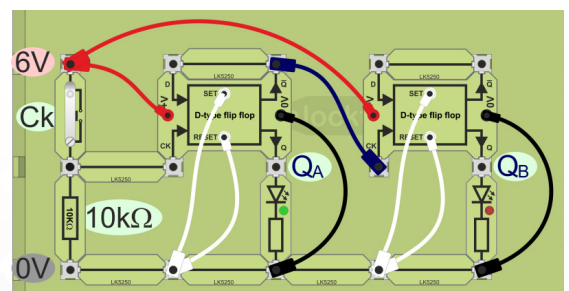


2 - Two-bit counter

- Add a second D-type bistable, as shown. Again, the Data input is connected to the Q output. The red LED now shows the state of the second bistable output.
- Switch on the 6V power supply. If any LED is lit, press the 'Ck' switch until both are off.
- Generate clock pulses, as before and watch the effect on the two LEDs.
- Record your results in the table that follows.



Clock pulse	QA	QB
0	Off	Off
1		
2		
3		
4		
5		



WORKSHEETS

Worksheet 5 - The 1-Bit Counter

So What?

The nub of this behaviour is the connection between $\bar{Q}$ and the Data input. The sequence of events, illustrated in the table, is:

- $\bar{Q}$ is always in the opposite state to Q.
- $\bar{Q}$ is connected to the Data input.
- Hence the Data input is always in the opposite state to Q.
- On the next clock pulse, Q copies D and moves to the opposite state.

Clock pulse	Q Output	$\bar{Q}$ Output	Data input
1	0	1	1
2	1	0	0
3	0	1	1
4	1	0	0
5	0	1	1
6	1	0	0

The table opposite repeats the comparison of binary and decimal number systems, with the binary numbers written over four columns.

Decimal number	Equivalent binary number DCBA			
	D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0

For example, '5' = '0101'

Look at the 'A' column.

It follows the same pattern as the Q output of the bistable. As a result, this subsystem can be called a 1-bit counter.

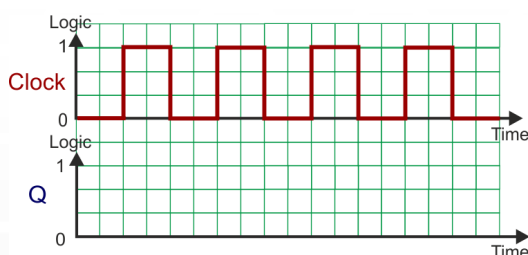
In circuit 2, the second bistable acts in the same way as the first. The Q output changes state every time it receives a clock pulse. However, here, the Q output of the first bistable provides these clock pulses. As your results show, it takes two pulses from the switch to create one of these clock pulses. In other words, its behaviour looks like column B of the table above. In summary, this circuit behaves as a 2-bit binary counter.

A challenge -

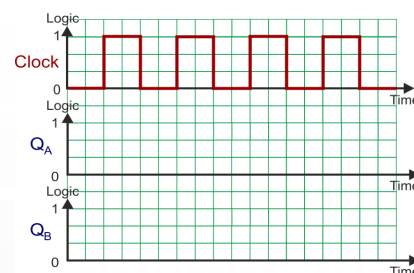
- Modify the 2-bit counter so that the second bistable receives clock pulses from the $\bar{Q}$, not Q, output of the first. Spot the difference!

For your records:

- Copy the circuit diagrams and test results for the 1-bit and 2-bit counters.
- Use your results to complete the timing diagrams given below.



1-bit counter



2-bit counter

WORKSHEETS

Worksheet 6 - Debouncing With a Monostable



Counting the number of times a switch is pressed poses a significant problem - switch bounce.

Switches are designed using springy materials - brass, steel etc. A switch must close quickly, to avoid 'arcing'.

In the process, the two metal contacts come together very quickly - and often bounce off again, and again.

What is meant to be one press of the switch looks to the electronics as a number of presses.

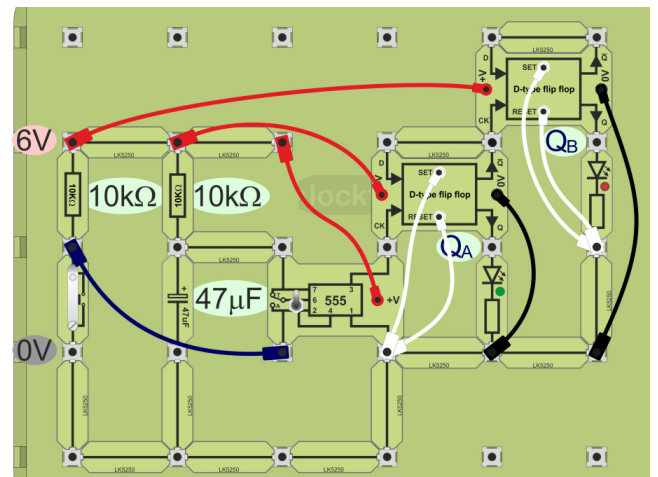
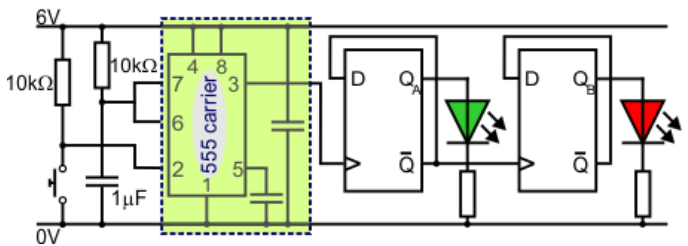
The solution is to add a debouncing circuit, to remove the surplus pulses. As these worksheets shows, there are a number of ways to do this.

Over to you:

- Set up the circuit shown. Notice that $\bar{Q}_A$ is connected to the clock input of the second D-type bistable. In other words, it is connected to count up.
- Switch on the 6V power supply.
- Press and release the switch a number of times, (roughly once per second).
- Observe the effect on the LEDs.
- Use the table to record what you see.

Clock pulse	Q _A	Q _B
0	Off	Off
1		
2		
3		
4		
5		

- The aim is to eliminate switch bounce. The proof is a results table that shows an unbroken and correct sequence of outputs.
- Optimise the time delay using different values of timing components. Make a note of the optimum values for your system.
- Another source of unwanted pulses is noise on the power rails. This can be reduced by connecting a large value capacitor between the +6V rail and 0V.



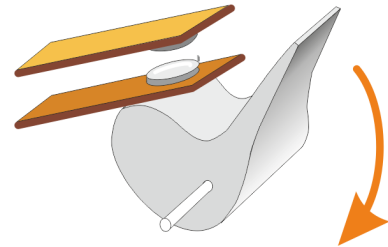
Try this out, but **be very careful to connect the capacitor the right way round**, if it is an electrolytic capacitor. The positive electrode must be connected to the +6V power rail, and the negative electrode to 0V.

WORKSHEETS

Worksheet 6 - Debouncing With a Monostable

So What?

The monostable as a debouncing subsystem:
The diagram illustrates the inner working of a switch.
It has two springy metal contacts.



- When the switch is off, they are not touching.
- When it is turned on, they are forced together.

The changeover must be rapid, as otherwise prolonged sparking can occur when the contacts are almost touching, which will cause wear and corrosion, reducing the lifetime of the switch.

However, as the contacts are springy, this rapid closing can cause them to bounce open again, and bounce several times before finally coming to rest in the closed position. This creates false pulses in a counting system.

You probably observed this in the previous worksheet, where the sequence of output states sometimes jumped one or more states. The first method for tackling the problem is to add a monostable, set up to create a very short delay. For the circuit shown on the previous page, the formula $T = 1.1 \times R \times C$ predicts a delay of 11ms.

During this time, the output of the monostable remains steadily at logic 1, no matter how many times the clock input bounces up and down. Providing all switch bounces cease before the output falls to logic 0, this method will debounce the switch.

The disadvantage of this method is that the counter might miss pulses if they arrive with a very short time interval between them. This is where the need to optimise the time delay is important. A long delay means that it is very unlikely that switch bounce will be a problem, but the system is insensitive during that delay.

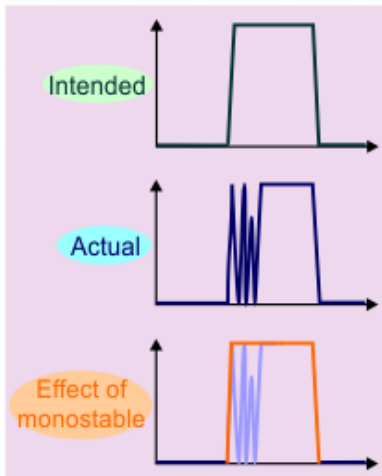
Electrical noise contains a range of high frequencies. Capacitors offer very little impediment to high frequencies, but block low frequencies. A capacitor connected between the power rails offers an easy route to these high frequencies, effectively reducing their effect on the rest of the system.

For your records:

- Copy the circuit diagram for the system, showing the use of the monostable to debounce the switch.
- Add your best set of timing component values to the diagram.
- Explain the principle behind the method, in terms that one of your fellow students would understand.
- Explain how to reduce noise in power rails, using a decoupling capacitor.

WORKSHEETS

Worksheet 7 - Debouncing with a Bistable



The previous worksheet showed that a monostable subsystem can mask the unwanted pulses produced by switch bounce.

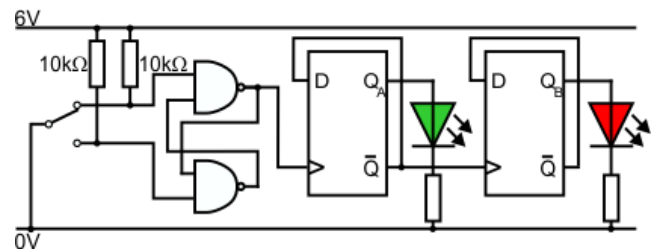
The diagram illustrates this principle. The monostable output, (orange,) stays high until the bouncing has stopped. Overall, the counting system receives a single pulse, just as intended.

There are side-effects! The counter may not see some valid pulses, or may see spurious pulses created in the power supply.

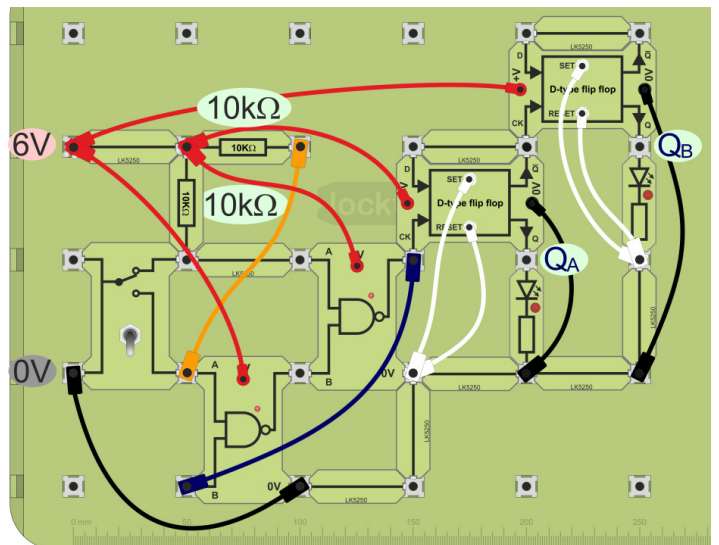
The next solution has none of these disadvantages. It relies on the latching property of a bistable.

Over to you:

- Set up the circuit shown. Once again, $\bar{Q}_A$ delivers clock pulses to the second bistable, to make it count up.
- Switch on the 6V power supply.
- Move the switch from one position to the other and observe the LEDs as you do so.
- Continue until both LEDs are off, (i.e. Q_A and Q_B are both at logic 0.)
- Now operate the switch five times, to send in five clock pulses. Observe the state of the LEDs attached to Q_A and Q_B as you do so.
- Complete the table with your results.



Clock pulse	Q_A	Q_B
0	Off	Off
1		
2		
3		
4		
5		



- The operation of this system should be flawless.

The main drawback is its dependence on a single-pole-double-throw switch, which is fine for this application, but not always suitable when using other sensing subsystems.

WORKSHEETS

Worksheet 7 - Debouncing with a Bistable

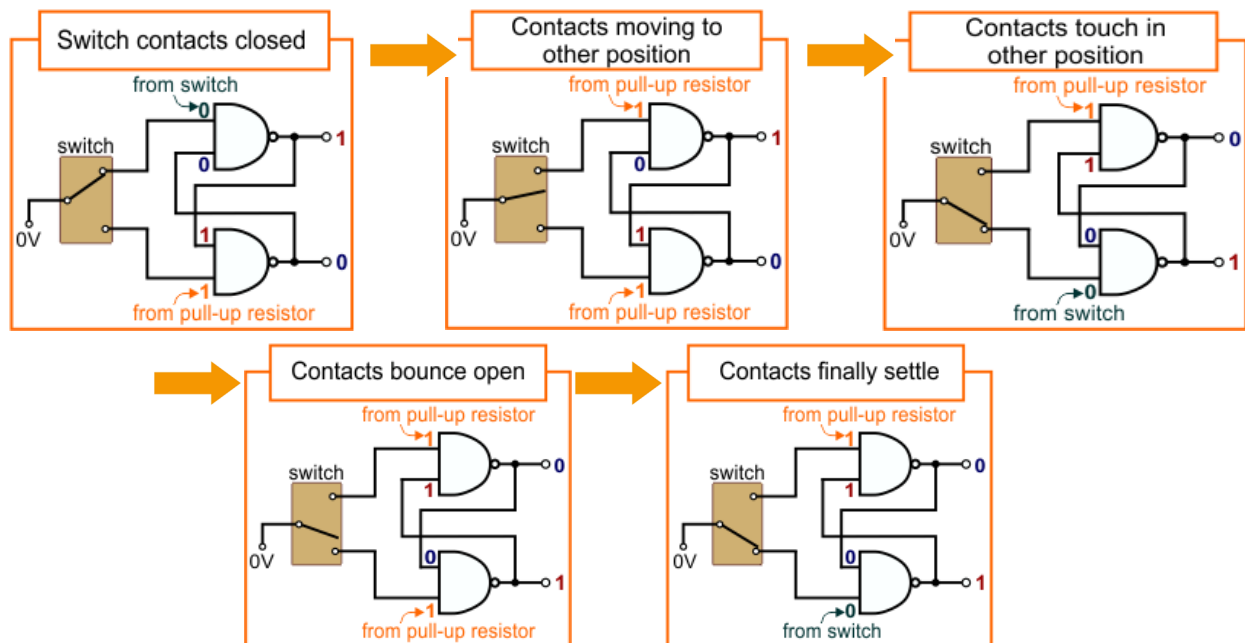
So What?

The bistable as a debouncing subsystem:

You may need to look back at worksheet 3 to appreciate how debouncing works here.

- When the switch contacts are closed, one of the NAND gate inputs is connected to 0V, (logic 0). As a result, the output of that NAND gate is logic 1.
- When the switch is moved to the other position, the contacts separate. No NAND gate is connected to 0V. Instead, the 10kΩ pull-up resistors hold both inputs at logic 1. This is the latching condition, and the output of the gates does not change.
- When the switch contacts meet again, the other NAND gate input is connected to logic 0, and its output goes to logic 1. The contacts may bounce open again, making both inputs return to logic 1, but the outputs do not change. This is the debouncing action.

This sequence is illustrated in the following diagrams.



A challenge -

Connect an oscilloscope to the output of a switch unit, to obtain a trace showing contact bounce. Suggested settings for the oscilloscope are:

Timebase = 1ms/div.

Voltage sensitivity = 1V/div

For your records:

- Copy the circuit diagram for the system, showing the use of the bistable to debounce the switch.
- Explain the principle behind the method to one of your fellow students, and then ask her/him to explain it back to you to test understanding.

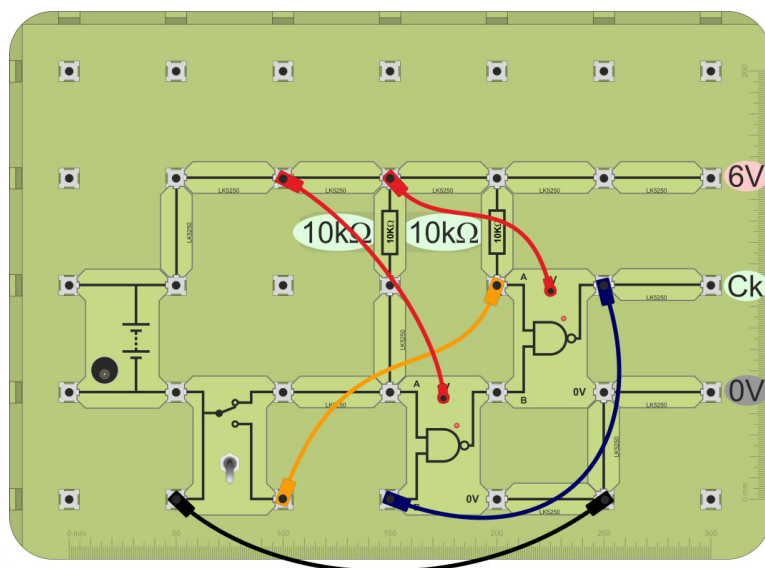
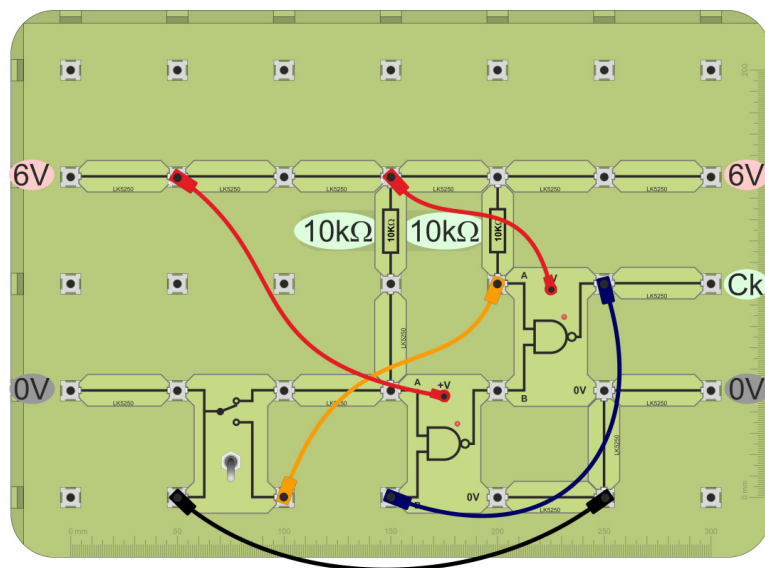
WORKSHEETS

Worksheet 7A - The Bistable Debouncer

The next three worksheets require a reliable debounced switch. To streamline the process, you are going to assemble one on a separate baseboard, which you can then connect to other subsystems, in the following worksheets.

Over to you:

- Set up one of the debouncing circuits shown below. The second one is for plug-top power supply users.
- To test it:
- Connect a LED carrier between the 'Ck' output and 0V;
- Operate the switch a number of times;
- Check the performance with that seen in the previous worksheet.



- Leave it assembled so that you can use it with the subsystems described in the following three worksheets.

WORKSHEETS

Worksheet 8 - The 3 Bit Counter



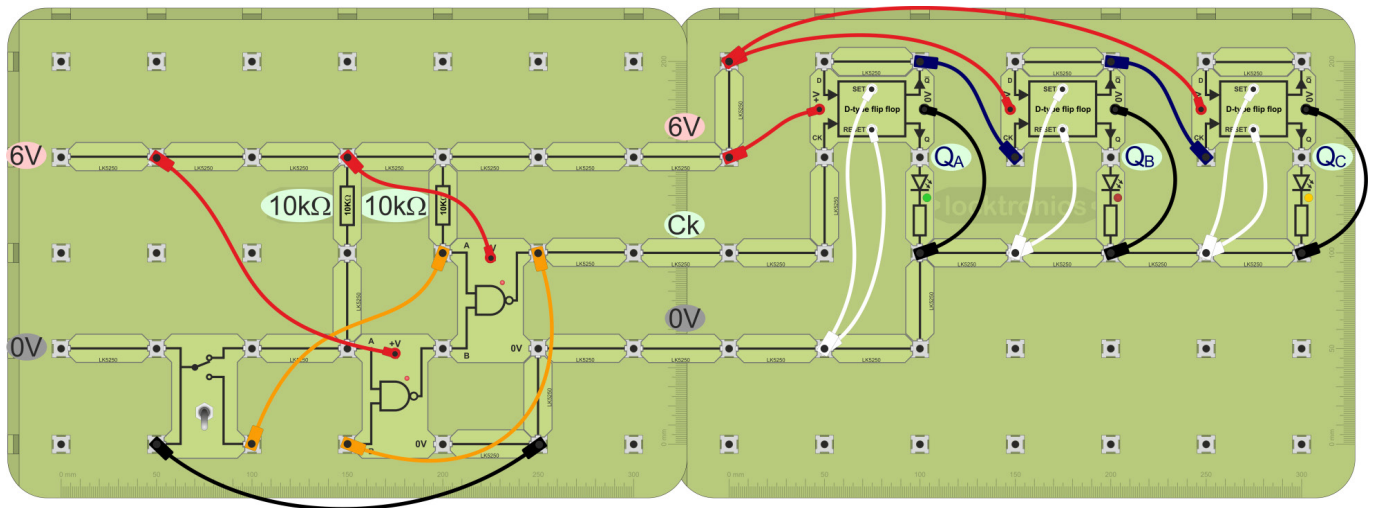
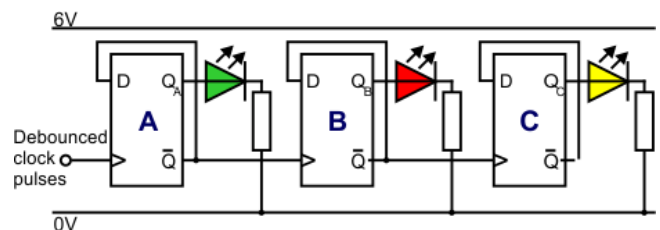
Worksheet 5 studied a 2-bit counter, built from D-type bistables. This one adds a third counting stage, and the next develops it into the 'modulo-n' counter, which resets on the 'nth' clock pulse.

This worksheet shows that the performance of the counter depends on where you take the clock pulse from.

Over to you:

This exercise investigates a 2:4 decoder, where a two-bit number at the input selects one of four outputs.

- Build the three-bit counter, and attach the debounced switch baseboard. Again, the Q outputs deliver clock pulses to the next stage, to make the system count up.



- Switch on the 6V power supply.
- Use the debounced switch to generate clock pulses and observe the effect on the LEDs.
- Continue until all three LEDs are off, (i.e. Q_A, Q_B and Q_C are both at logic 0.)
- Now operate the switch ten times, to create ten clock pulses. Observe the output LEDs as you do so.
- Complete the table with your results.

Clock pulse	Q _A	Q _B	Q _C
0	Off	Off	Off
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

- Do not dismantle this circuit!**

It forms the basis for the modulo-n counter on the next worksheet.

WORKSHEETS

Worksheet 8 - The 3 Bit Counter

So What?

- First of all, convert your results showing whether the LEDs were on or off into logic levels. Remember - when an LED is lit, the output connected to it is at logic 1.

Clock pulse	Q _A	Q _B	Q _C
0	0	0	0
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

- Compare your results with the table, given in the introduction, which relates the binary and decimal number systems.

Decimal	Binary	Decimal	Binary
0	0	8	1000
1	1	9	1001
2	10	10	1010
3	11	11	1011
4	100	12	1100
5	101	13	1101
6	110	14	1110
7	111	15	1111

A challenge -

Re-arrange the wiring so that the Q outputs, not Q outputs, provide clock pulses for the following bistable.

What effect does this have on the performance of the counting system?

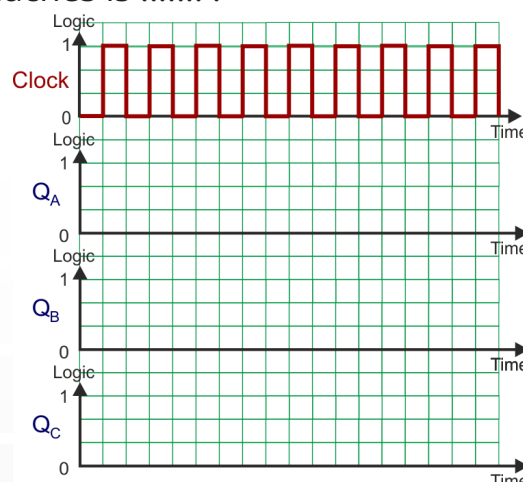
For your records:

- Copy the circuit diagrams and test results for the 3-bit counter.
- Copy and complete the sentence:

The biggest number, in decimal, that this counter reaches is

- Use your results to complete the timing diagram given opposite.
- Draw the circuit diagram for a 4-bit up-counter based on D-type bistables.
- Copy and complete the sentence:

The biggest number, in decimal, that a 4-bit counter can reach is



WORKSHEETS

Worksheet 9 - The 'Modulo-n' Counter



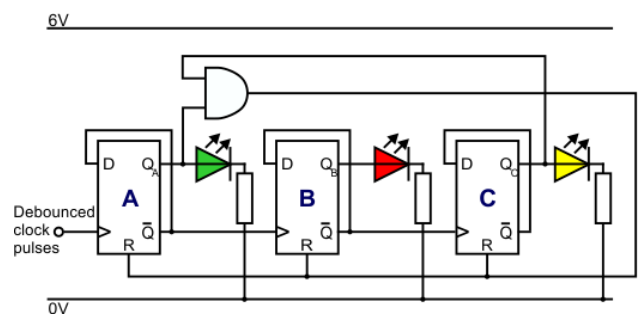
Worksheet 5 studied a 2-bit counter. Worksheet 8 added a third counting stage. This one creates a 'modulo-n' counter, which resets on the 'nth' clock pulse. The 7-segment display is added to show the count as a decimal number.

The 3-bit counter resets once a particular number is reached. This is used in applications where a maximum count is required. For example, when packing eggs, the counter may need to reset on reaching '6' (known as a 'modulo-7' counter). The system closes the box and starts again.

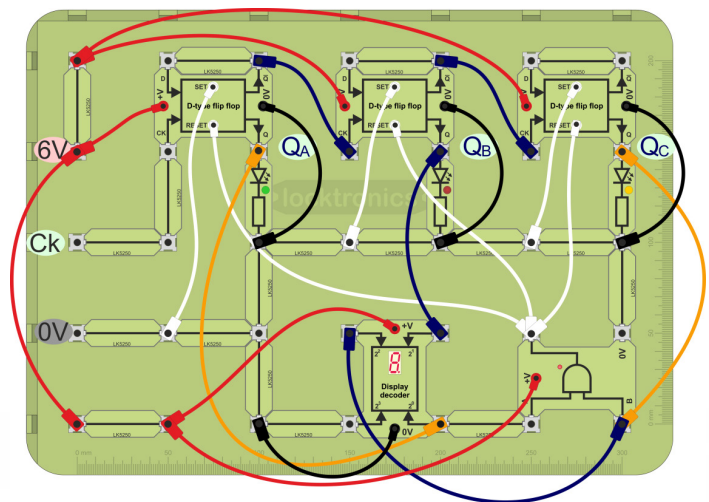
Over to you:

Set up the circuit including the debounced switch baseboard to deliver clock pulses via the 'Ck' connection. (The circuit from the previous worksheet, can be modified by adding the AND gate, and 7-segment display.)

- The procedure is the same as in worksheet 8. Switch on the 6V power supply.
- Use the switch on the debounced switch baseboard to generate clock pulses and observe the LEDs as you do so.
- Continue until all three LEDs are off.
- Now operate the switch ten times, to send in ten clock pulses. Observe the state of the outputs LEDs as you do so.
- Complete the table with your results.



Clock	Q _A	Q _B	Q _C	Display
0	Off	Off	Off	0
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				



WORKSHEETS

Worksheet 9 - The 'Modulo-n' Counter

So What?

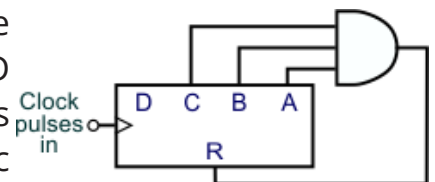
- As before, convert your results into logic levels. ('Off' = '0', 'On' = '1'.)
- Compare them with the results for worksheet 8. There, the maximum binary number was '111', on the seventh clock pulse. The counter then reset on the eighth pulse.
- This system relies on the 'Reset' inputs, which reset the bistable (make the Q output logic 0,) when a logic 1 signal is sent to the Reset input.
- On your system, the Reset inputs of all three bistables connect to the output of a 2-input AND gate.

Clock	Q _A	Q _B	Q _C	Display
0	0	0	0	0
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Its input signals come from Q_A and Q_C. As a result, they reset when Q_A = 1 AND Q_C = 1, i.e. on the fifth clock pulse. The reset process takes only microseconds and so we do not see this state on the LEDs. Instead, the LEDs turn off.

- In electronics jargon, it is a modulo-5 counter:
- It resets on the 5th clock pulse.
- The outputs have 5 possible states - 000, 001, 010, 011, 100. (Notice - the biggest is '4' in decimal, because '0' is included as one of the 5 states.)

The counter can be contained within a single chip. The diagram shows a 4-bit counter, connected to a 3-bit AND gate as a modulo-7 counter. The biggest number it outputs is '110' ('6' in decimal,) making it suitable for an electronic dice, or the 'tens of minutes' digit in a clock.



For your records:

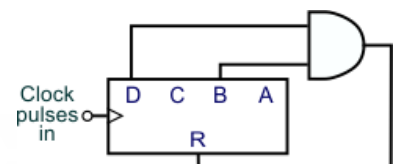
- Copy and complete the statement:

A modulo-5 counter resets on the th clock pulse, and outputs numbers up to

- Copy the diagram and complete the statements:

The diagram shows a modulo-..... counter.

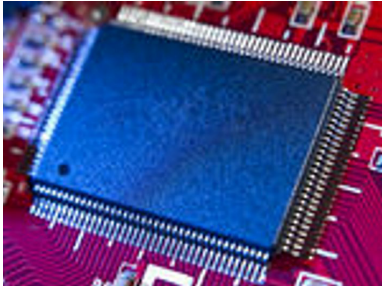
The biggest number it outputs is



- Copy the diagram for the modulo-7 counter, given above, and explain why it is suitable for the 'tens of minutes' display in an electronic clock.

WORKSHEETS

Worksheet 10 - The 3 Stage Shift Register



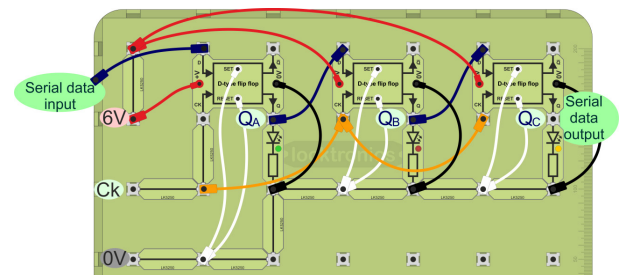
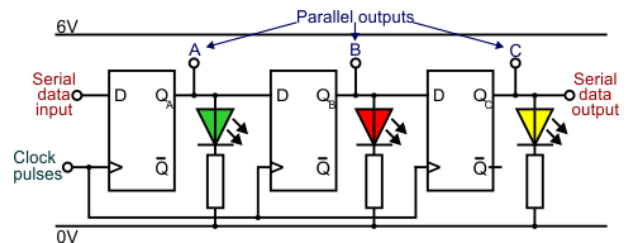
In electronics, the word 'register' often means a data store, a location where a binary number can be saved for later use.

Electronic memory is an array of these registers. Questions that arise include "How do I find it again later?" and "How big a number can I store?"

This worksheet begins that study with a look at how a series of bistables can be used to store data, and how that data can be shifted in and out of that register.

Over to you:

- Set up the shift register circuit, and attach the debounced switch baseboard to deliver clock pulses via the 'Ck' connection
- Connect the serial data input to 6V (logic 1).
- Switch on the 6V power supply.
- Use the debounced switch to generate clock pulses and observe the LEDs as you do so.
- Once all three LEDs are on, connect the serial data input to 0V (logic 0).
- Now send in further clock pulses, and notice the effect on the LEDs.
- By changing the serial data input connection, store the number '101' in the shift register. It will stay there indefinitely as long as the power supply remains active, and providing no further clock pulses are applied to the input.
- The stored data is extracted from the serial data output, by sending in sufficient clock pulses, in this case three.



Notice the differences between the counter and shift register circuits:

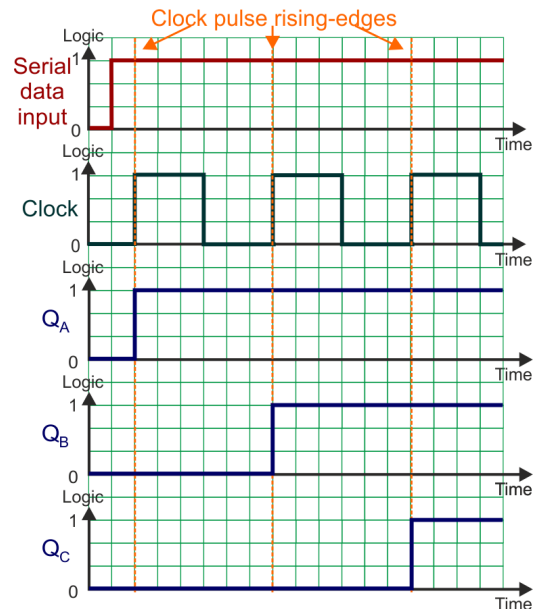
- Clock pulses from the debounced switch are fed to all bistables simultaneously;
- The Data input of each bistable is connected to the previous Q output, not $\bar{Q}$ output.

WORKSHEETS

Worksheet 10 - The 3 Stage Shift Register

So What?

- Each D-type bistable copies the state of its Data input to its Q output on the rising-edge of a clock pulse.
- Here, apart from the first bistable, each Data input is connected to the previous Q output.
- The combined effect is that the data bit applied at the Data input of the first bistable is passed along to the following bistables, one step at a time, by sending in a series of clock pulses.
- To load a 8-bit number, it is applied in serial fashion, i.e. one bit at a time, to the first Data input, and then eight clock pulses move the data to the eight bistables needed to store it.
- A further eight clock pulses will push the data out of the serial data output.
- The timing diagram opposite shows this. To make this more understandable, remember that changes in the outputs occur extremely rapidly. It may work better to think that the Q output of each bistable copies the data input logic level just before the rising-edge of the clock pulse.



Types of shift register:

The circuit you built allowed data to be fed in, one bit at a time, via the serial data input. Eventually, that data could be extracted from the serial data output. This is known as a 'serial-in-serial-out' (SISO) shift register.

However, once the data is stored as described above, it can be extracted simultaneously from each Q output. This is a 'serial-in-parallel-out' (SIPO) shift register.

Other variations allow all data bits to be inserted and extracted simultaneously - a 'parallel-in-parallel-out' (PIPO) shift register, or inserted simultaneously, but extracted one bit at a time from the serial data output - a 'parallel-in-serial-out' (PISO) shift register.

For your records:

- Copy the circuit diagram for the shift register given on the previous page.
- Write a bullet list of the steps needed to store the number '101' in this shift register.
- Draw a timing diagram for the process you just described.
- Describe the differences between SISO, SIPO, PISO and PIPO shift registers.

WORKSHEETS

Worksheet 11 - The R-2R DAC



Digital processing has a number of advantages:

- Regeneration - errors and noise can be eliminated;
- Added functionality - effects like compression and encryption;
- Increased security.

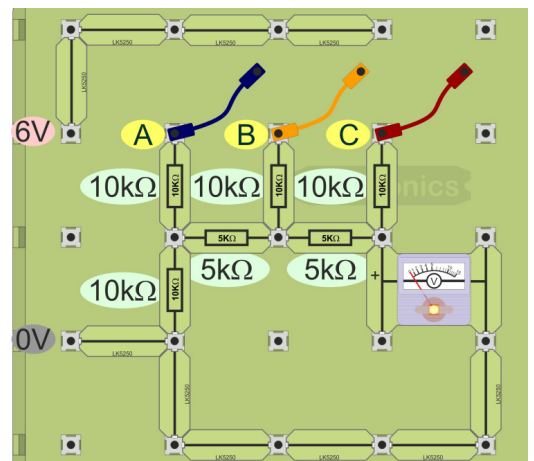
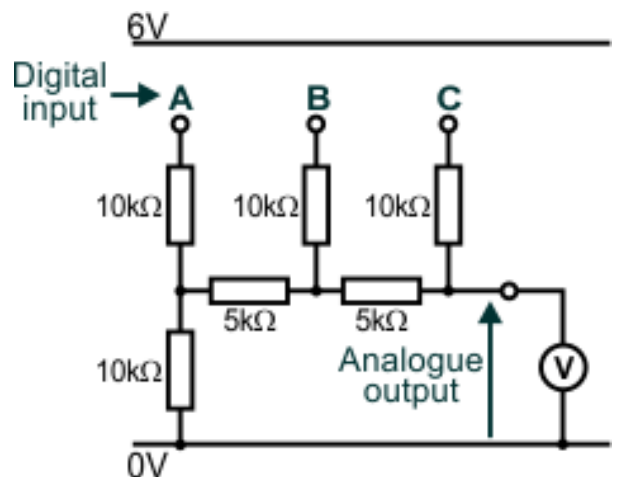
Audio signal processing has embraced digital technology in the form of CDs, DVDs and media streaming over the internet.

However, ultimately, our ears are analogue devices and so digital audio signals must be converted into analogue signals, if we are to hear them correctly. This requires a digital-to-analogue converter (DAC).

There are a number of ways to implement the DAC circuit. This worksheet looks at one known as the R-2R ladder.

Over to you:

- Set up the circuit, leaving inputs A, B and C unconnected.
- The input, the digital number CBA, is created by attaching individual bits (A, B etc.):
- To the 6V rail, to set them to logic 1,
- To the 0V rail, to set them to logic 0.
- Switch on the 6V power supply.
- Connect inputs A, B and C to 0V, creating the binary number 000. Note down the initial voltmeter reading in the first row of the table.
- Next, connect input A to 6V, and B and C to 0V, (binary number 001), and record the new reading.
- Now, connect input B to 6V, and A and C to 0V, (binary number 010). Again, note down the resulting reading.
- Continue in this way to input the other binary numbers, and complete the table with the corresponding output voltages.



Digital input	Analogue output
0	V
001	V
010	V
011	V
100	V
101	V
110	V
111	V

WORKSHEETS

Worksheet 11 - The R-2R DAC

So What?

- In a simple Digital-to-Analogue converter, the analogue output voltage should copy changes in the digital input number.
- As the number gets bigger, the output voltage gets bigger.
- When the number doubles, the output voltage should double, and so on.
- Look at your results table! Can you see this behaviour?

Remember that all electronic components have manufacturing tolerances. Even though the resistors are high quality, they have a tolerance of 5%. The 10k Ω resistor may legitimately be anywhere between 9.5k Ω and 10.5k Ω .

- With this type of DAC, only two values of resistor are needed, one twice as big as the other. The accuracy of the conversion depends on having precise resistor values. The theory behind this circuit is quite complex, and is best tackled through Thevenin's theorem, which is beyond the scope of this course. Never mind the theory - it works!

Try this:

- Make a 4-bit R-2R ladder, and test it. (Your results table will need sixteen rows.)
- Replace the analogue voltmeter with a multimeter set onto the 20V range. The output voltage readings may rise by 30%! Why is this?

Any device connected directly to the ladder acts like another resistor and affects the output voltage.

An ideal voltmeter has infinite resistance, and draw no current from the R-2R ladder. The analogue voltmeter takes more current from the ladder circuit than the multimeter, reducing the voltage that appears at the output.

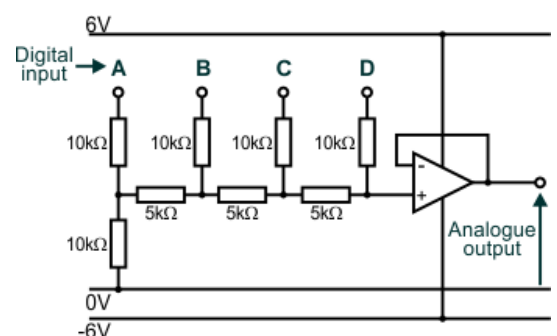
- Add a 'buffer' to the output of the R-2R ladder.

In its simplest form, this produces an output voltage that copies the input. However, any current drawn from the output comes from the power supply, not from the ladder.

An op-amp 'voltage follower' can be used for the job, as the circuit shows, but it will need a 'split' power supply, with an additional -6V supply.

For your records:

- Copy the circuit diagram for the 3-bit (unbuffered) DAC, given on the previous page.
- Copy your completed results table for the system.
- Add an explanation for why the output is bigger when a multimeter is used.
- Draw the circuit diagram for the 4-bit buffered DAC, using an op-amp voltage follower sub-system.



[illegible]

[illegible]

[illegible]

SEQUENTIAL LOGIC

CP9945

www.matrixtsl.com

Matrix Technology Solutions
The Factory, 33 Gibbet Street, Halifax, HX1 5BA, United Kingdom
t: +44 (0) 1422 252380 e: sales@matrixtsl.com