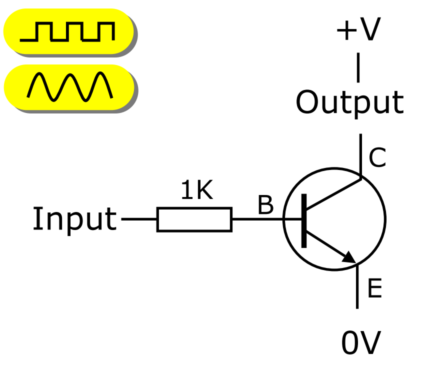
PROCESS

NPN Transistor

Diagram:



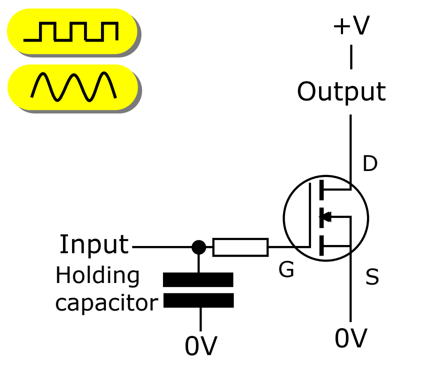
Function:
When the input is less than 0.7V the output is off. The output is on when the input is greater than 0.7V.

The max current that flows is determined by the transistor type. **BC548B** is 300mA, **BC337** is 800mA

PROCESS

N MOSFET with memory

Diagram:



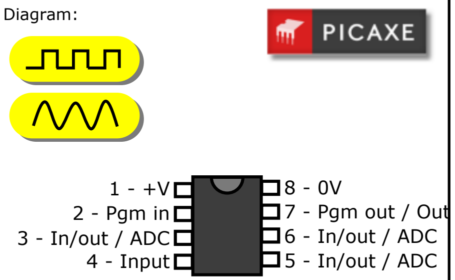
Function:
When the input is less than 3V the output is off. The output is on when the input is greater than 3V, and **stays on** until it is made 0V.

The max current that flows is determined by the transistor type. **BS170** is 500mA, **ZVN4206A** is 1A

PROCESS

8 pin Microcontroller

Diagram:



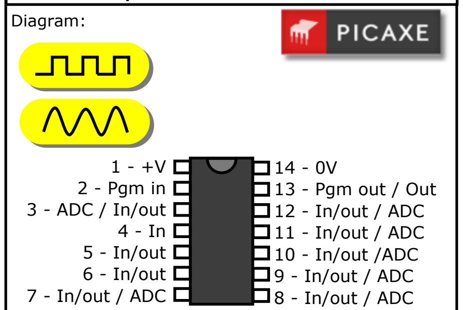
Function:
A programmable component with a number of inputs (analogue & digital) and outputs.

Each output can supply up to 20mA, if more current is needed use either a transistor driver or interface driver.

PROCESS

14 pin Microcontroller

Diagram:



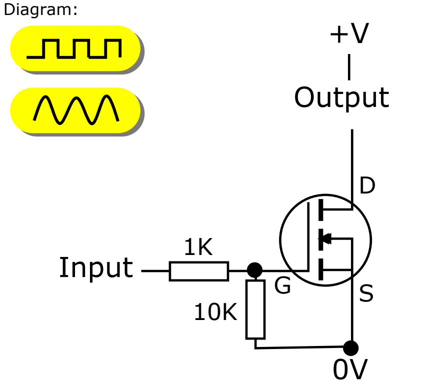
Function:
A programmable component with a number of inputs (analogue & digital) and outputs.

Each output can supply up to 20mA, if more current is needed use either a transistor driver or interface driver.

PROCESS

N MOSFET Transistor

Diagram:



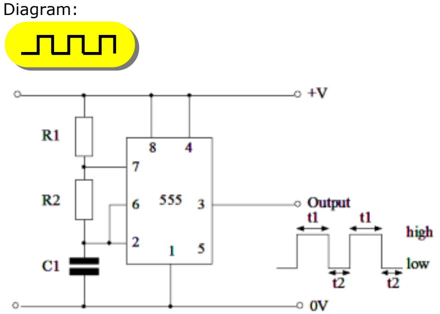
Function:
When the input is less than 3V the output is off. The output is on when the input is greater than 3V.

The max current that flows is determined by the transistor type. **BS170** is 500mA, **ZVN4206A** is 1A

PROCESS

555 Timer Astable

Diagram:



High time, T1 = 0.693 x (R1 + R2) x C
Low time, T2 = 0.693 x R2 x C

Function:
The 555 is a general purpose timer, which can be configured to operate as an astable.

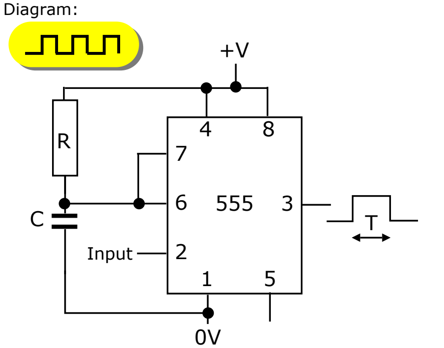
The output frequency is calculated using:

F = 1.44 / C x (R1 + R2 + R2)

PROCESS

555 Timer Monostable

Diagram:



Pin 2 is the input which is normally high and goes low to trigger the pulse

Function:
The 555 is a general purpose timer, which can be configured to operate as a monostable.

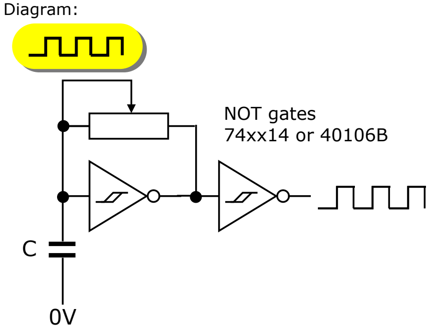
The output pulse length is calculated using:

T = 1.1 x R x C

PROCESS

NOT gate Astable

Diagram:



Note: Most chips have at least 4 gates on them, and always tie unused inputs to 0V or +V

Function:
NOT logic gates can be used to produce simple astables as shown above. The variable resistor is used to 'trim' the frequency of operation.

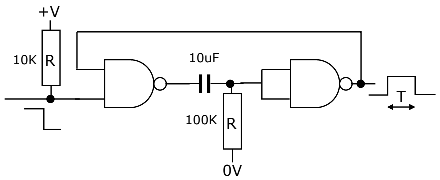
The output frequency is calculated using:

F = 1.44 / R x C

PROCESS

NAND gate Monostable

Diagram:



Typical NAND Gate 74xx00, 4011, 4093

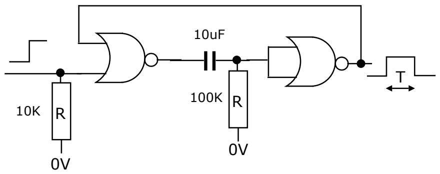
Function:
A monostable can be made from NAND gates in the following manner. A low pulse at the input will trigger a pulse, whose length is approx:

T = R x C

PROCESS

NOR gate Monostable

Diagram:



Typical NOR Gate 74xx02, 4001

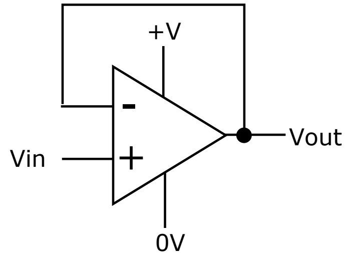
Function:
A monostable can be made from NOR gates in the following manner. A high pulse at the input will trigger a pulse, whose length is approx:

T = R x C

PROCESS

Op-Amp Buffer

Diagram:



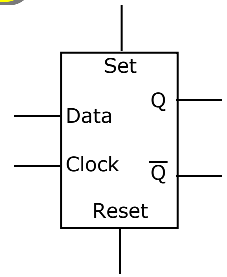
Typical Op-Amp LM324N

Function:
This Op-Amp circuit is known as a **Unity Gain Buffer Amp**, its output is the same voltage as the input, but the current the Op-amp provides is 50mA. This circuit is used to provide higher output currents from circuits that provide low output current.

PROCESS

Logic gate Flip Flop

Diagram:



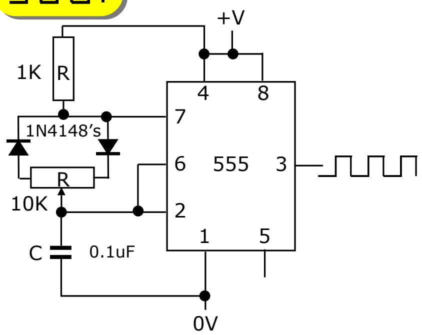
4013 Dual D Flip Flop

Function:
A Flip flop is a simple memory, the logic value at the Data input is stored at the output Q (Q-bar is the opposite Q), when the clock input is pulsed.
The set and reset inputs are normally connect to 0V, they can be used independently if required - use a pull down resistor with a switch for manual operation.

PROCESS

555 Timer PWM

Diagram:



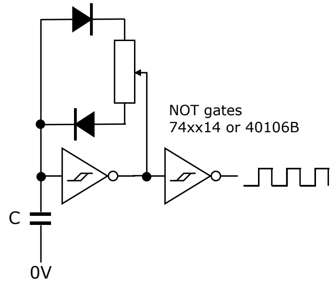
Function:
A 555 timer PWM circuit, this configuration gives variable mark/space ratio at approx 1kHz, use 0.01uF (10nF) for approx 10KHz.

Note the use of diodes to separate the charge/discharge paths to produce the PWM effect.

PROCESS

NOT gate Astable PWM

Diagram:



Note: R>20K for correct operation

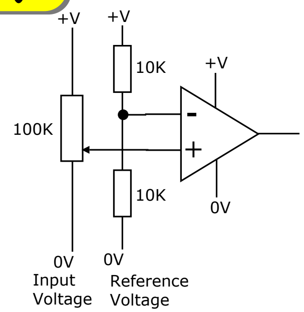
Function:
To control the speed of motors or brightness of LEDs, we use PWM (Pulse Width Modulation). The use of diodes separate the charge/discharge paths giving unequal times for the mark to space ratio.

Mark = ON, Space = OFF

PROCESS

Analogue Comparator

Diagram:



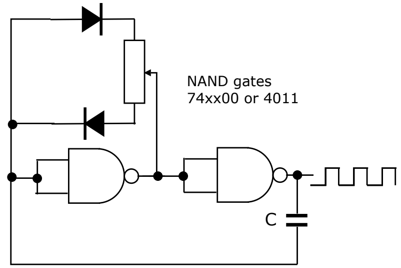
LM339 Quad Comparator

Function:
It compares the input voltage to the reference voltage, if input voltage falls below the reference voltage the output goes low, and high if its above. To make the output work the opposite way around use: - input for the input voltage and + input for the reference voltage.

PROCESS

NAND gate Astable PWM

Diagram:



Note: R>20K for correct operation

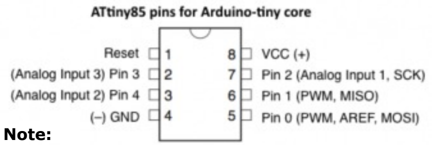
Function:
To control the speed of motors or brightness of LEDs, we use PWM (Pulse Width Modulation). The use of diodes separate the charge/discharge paths giving unequal times for the mark to space ratio.

Mark = ON, Space = OFF

PROCESS

8 pin Microcontroller

Diagram:



Note:

Can be programmed via the Arduino IDE, most standard commands supported and a simple connection circuit.

The ATtiny85 has 8K ROM, 512B RAM, 512B EEPROM

Function:

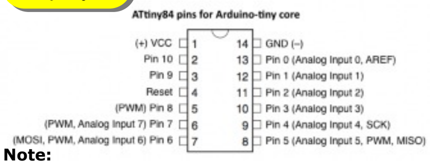
A programmable component with a number of inputs (analogue & digital) and outputs.

Each output can supply up to 40mA, if more current is needed use either a transistor driver or interface driver.

PROCESS

14 pin Microcontroller

Diagram:



Note:

Can be programmed via the Arduino IDE, most standard commands supported and a simple connection circuit.

The ATtiny84 has 8K ROM, 512B RAM, 512B EEPROM

Function:

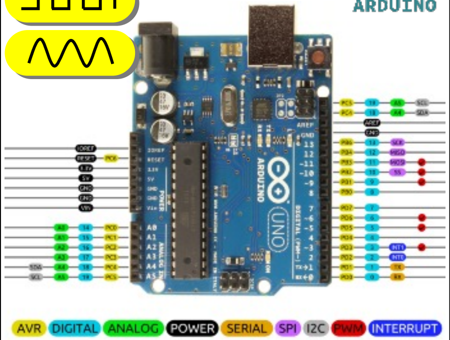
A programmable component with a number of inputs (analogue & digital) and outputs, and is Arduino compatible.

Each output can supply up to 40mA, if more current is needed use either a transistor driver or interface driver.

PROCESS

Arduino Uno

Diagram:



Function:

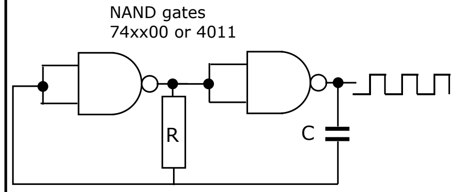
A programmable module with a number of inputs (analogue & digital) and outputs, and has a wide range of add-on Shields to extend its capacity.

Each output can supply up to 40mA, if more current is needed use either a transistor driver or interface driver.

PROCESS

NAND gate Astable

Diagram:



Note: Most chips have at least 4 gates on them, and always tie unused inputs to 0V or +V

Function:

NAND logic gates can be used to produce simple astables as shown above. The variable resistor is used to 'trim' the frequency of operation.

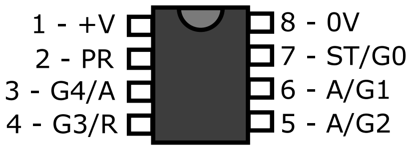
The output frequency is calculated using:

F = 1.44 / R x C

PROCESS

8 pin Microcontroller

Diagram:



Function:

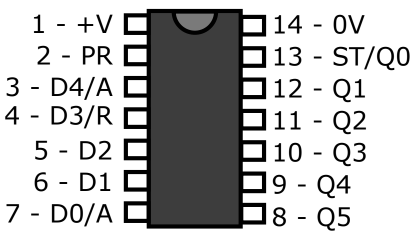
A programmable component with a number of inputs (analogue & digital) and outputs.

Each output can supply up to 20mA, if more current is needed use either a transistor driver or interface driver.

PROCESS

14 pin Microcontroller

Diagram:



Function:

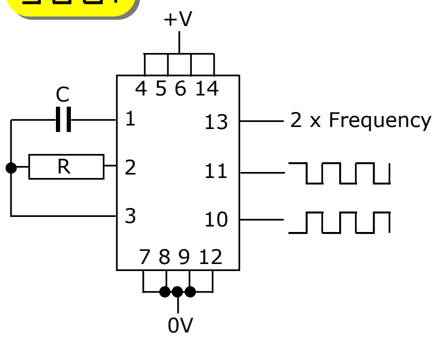
A programmable component with a number of inputs (analogue & digital) and outputs.

Each output can supply up to 20mA, if more current is needed use either a transistor driver or interface driver.

PROCESS

Digital Astable

Diagram:



Function:

The **4047** is a general purpose logic based timer, configured to operate as an astable. Output 1 and 2 are opposite to each other.

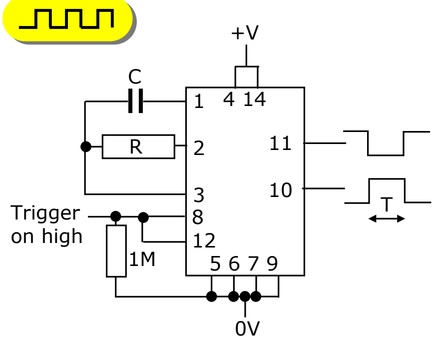
The frequency of the pulses is given by:

f = 1 / 4.4 x R x C

PROCESS

Digital Monostable

Diagram:



Function:

The **4047** is a general purpose logic based timer configured to operate as an monostable. Output 1 and 2 are opposite to each other. R =10K - 10M

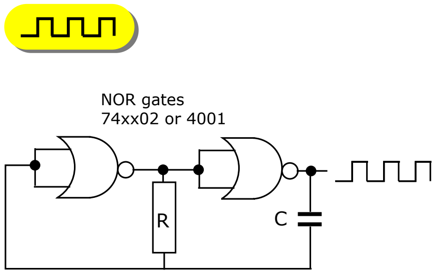
The output pulse length is given by:

T = 2.48 x R x C

PROCESS

NOR gate Astable

Diagram:



Note: Most chips have at least 4 gates on them, and always tie unused inputs to 0V or +V

Function:

NOR logic gates can be used to produce simple astables as shown above. The variable resistor is used to 'trim' the frequency of operation.

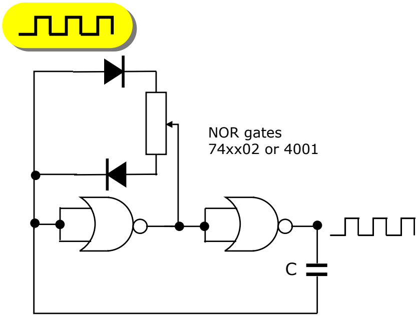
The output frequency is calculated using:

F = 1.44 / R x C

PROCESS

NOR gate Astable PWM

Diagram:



Note: Most chips have at least 4 gates on them, and always tie unused inputs to 0V or +V

Function:

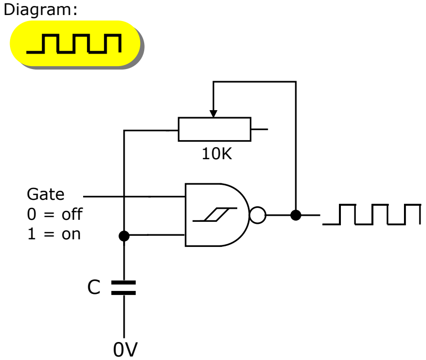
To control the speed of motors or brightness of LEDs, we use PWM (Pulse Width Modulation). The use of diodes separate the charge/discharge paths giving unequal times for the mark to space ratio.

Mark = ON, Space = OFF

PROCESS

NAND gate Gated Astable

Diagram:



74xx132 NAND Schmitt trigger

Function:

This astable is controlled by the gate input. If the gate is low NO pulses are produced. Making the gate high pulses are produced.

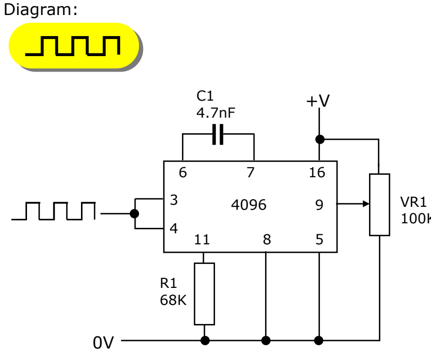
The output frequency is calculated using:

F = 1.44 / R x C

PROCESS

Full range VCO

Diagram:



0 - 5KHz
Max freq = R1 x C1

Function:

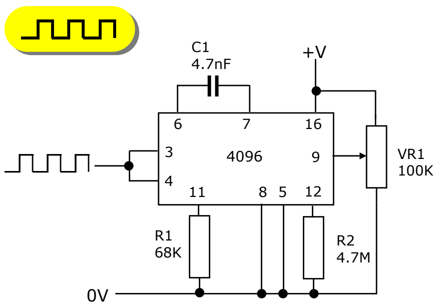
The 4046 is general purpose logic pulse generator, whose frequency is controlled a voltage, hence a Voltage Controlled Oscillator, VCO.

The input potentiometer could be a light or heat sensor see the Input cards.

PROCESS

Restricted range VCO

Diagram:



Fmin = C1 x R2
Fmax = C1 x (1/R1 x R2)

Function:

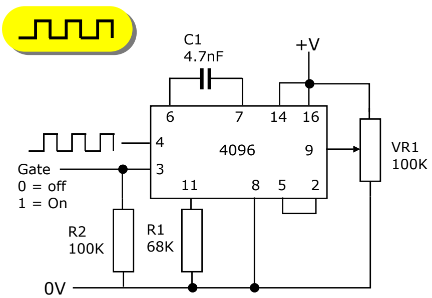
This 4046 VCO has a restricted range between *fmin* and *fmax*, the range is calculated as shown above.

The input potentiometer could be a light or heat sensor see the Input cards.

PROCESS

Gated VCO

Diagram:



0 - 5KHz
Max freq = R1 x C1

Function:

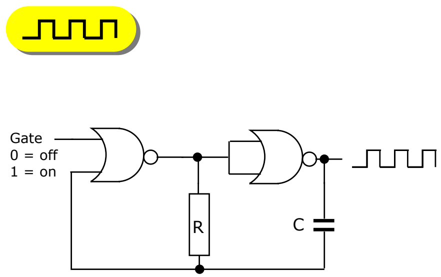
The 4046 is general purpose logic pulse generator, whose frequency is controlled a voltage, hence a Voltage Controlled Oscillator, VCO.

The input potentiometer could be a light or heat sensor see the Input cards.

PROCESS

NOR gate Gated Astable

Diagram:



4001 NOR gate

Function:

This astable is controlled by the gate input. If the gate is low NO pulses are produced. Making the gate high pulses are produced.

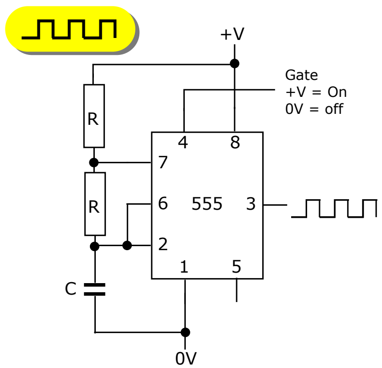
The output frequency is calculated using:

F = 1 / 2.2 x R x C

PROCESS

555 Gated Astable

Diagram:



Function:

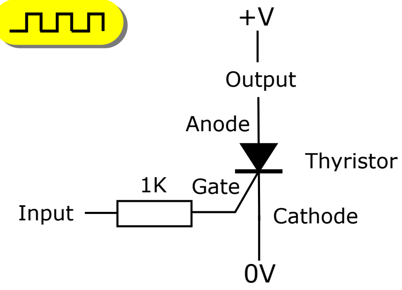
This is a variation of the 555 Timer astable, pin 4 is the gate input. If the gate is set to +V it will produce pulses, when at 0V **NO** pulses will be produced.

The output frequency is calculated using:

F = 1.44 / C x (R1+ R2 + R2)

PROCESS
Thyristor switching

Diagram:

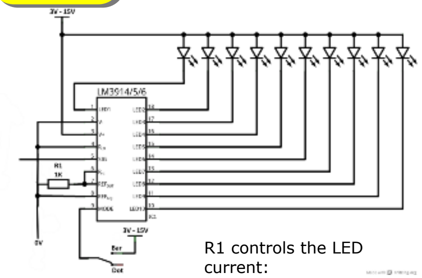


The input to the gate needs to be +V

Function:
A Thyristor is a "switched diode", when a one-off voltage is applied to the gate, the diode conducts. It will continue to conduct until the current between the anode and cathode is turned off.
The max current that flows is controlled by the type used: **2N5060g** is 800mA, **C106D** is 4A.

PROCESS
LM3916 Dot / Bar Driver

Diagram:



R1 controls the LED current:

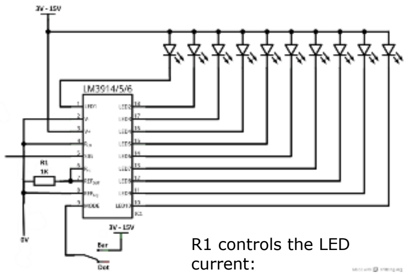
I_{LED} = 12.5/R1

Function:
A 10 LED driver with a **Vu meter response** to the input voltage at pin 5. When using 3V - 15V, the readout range is scaled to -3dB steps, +3dB to -20dB. Pin 9 controls the mode dot or Bar, Pin 9 = +V = bargraph, Pin 9 left open = Dotgraph.

Note: The input can withstand ±35V.

PROCESS
LM3914 Dot / Bar Driver

Diagram:



R1 controls the LED current:

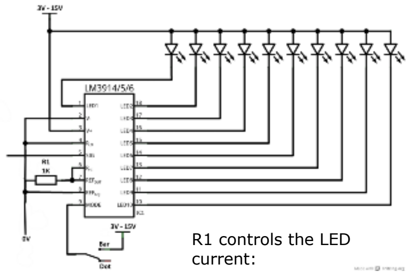
I_{LED} = 12.5/R1

Function:
A 10 LED driver with a **linear response** to the input voltage at pin 5. When using 3V - 15V, the readout range is 0.13V to 1.30V. Pin 9 controls the mode dot or Bar, Pin 9 = +V = bargraph, Pin 9 left open = Dotgraph.

Note: The input can withstand ±35V.

PROCESS
LM3915 Dot / Bar Driver

Diagram:



R1 controls the LED current:

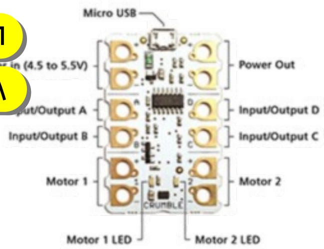
I_{LED} = 12.5/R1

Function:
A 10 LED driver with a **logarithmic response** to the input voltage at pin 5. When using 3V - 15V, the readout range is scaled to -3dB steps, 0dB to -27dB. Pin 9 controls the mode dot or Bar, Pin 9 = +V = bargraph, Pin 9 left open = Dotgraph.

Note: The input can withstand ±35V.

PROCESS
Crumble microcontroller

Diagram:



Note:

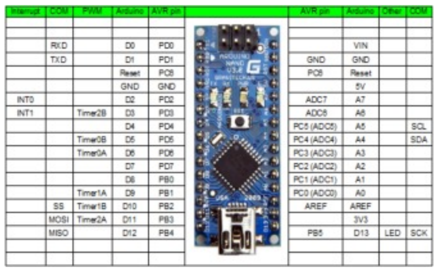
Pads allow terminal blocks to be fitted or a direct solder connection made, while the larger 4mm holes make an easy target for conductive thread and needle.

Function:
The Crumble programmable controller, can drive 2 motors forwards and backwards at variable speeds. It has 4 IO (Input/Output) pads which allow connections to switches, LDRs, low power LEDs and so on. Using 'croc leads' for quick and simple connections.

PROCESS
Arduino Nano

Diagram:

ARDUINO

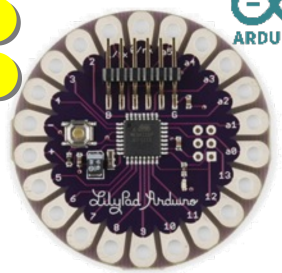


Function:
A programmable module with a number of inputs (analogue & digital) and outputs, the Nano is much smaller than the Uno, as a result can be built into small products.
Each output can supply up to 40mA, if more current is needed use either a transistor driver or interface

PROCESS
Arduino Lilypad

Diagram:

ARDUINO



Note:

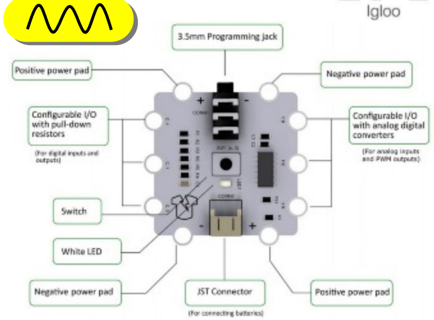
Pads allow terminal blocks to be fitted or a direct solder connection made, while the larger 4mm holes make an easy target for conductive thread and needle.

Function:
The Arduino Lilypad is specifically design for creating wearable products. It has all the capacity of an Uno but with a form factor that allows it to be easy stitched into products and to other e-textile components using conductive thread.

PROCESS
Igloo wearable module

Diagram:

Igloo

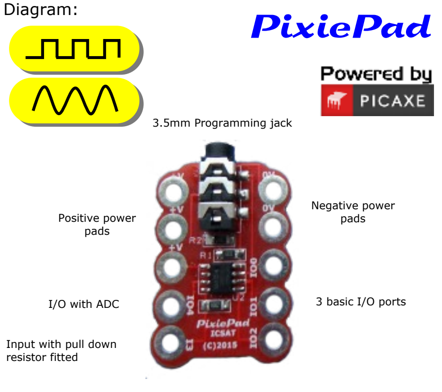


Function:
The Igloo is a programmable control board based on the PICAXE system, the wearable module offers an easy introduction to the world of wearables. The board can be used to read inputs such as switches and sensors. You can also control outputs like LEDs, buzzers.

PROCESS

Pixie Pad wearable module

Diagram:

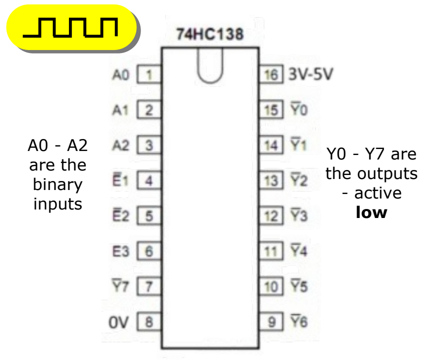


Function:
The Pixie Pad is a programmable control board based on the PICAXE system, the wearable module offers an easy introduction to the world of wearables. The board can be used to read inputs such as switches and sensors. You can also control outputs like LEDs, buzzers.

PROCESS

Binary to 8 Line Decoder

Diagram:

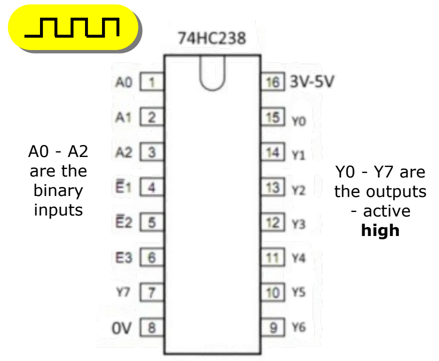


Function:
The 74HC138 is a BCD to 8 line driver it converts a binary number 0-7 into one of 8 outputs which goes **low**, which matches the binary number at the inputs. It is useful in scanning or multiplexing applications. To operate E1 & E2 must be **low** and E3 **high**.

PROCESS

Binary to 8 Line Decoder

Diagram:

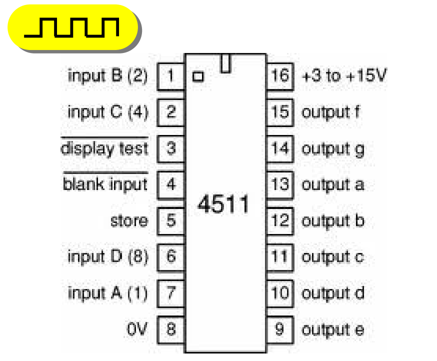


Function:
The 74HC238 is a BCD to 8 line driver it converts a binary number 0-7 into one of 8 outputs which goes **high**, which matches the binary number at the inputs. It is useful in scanning or multiplexing applications. To operate E1 & E2 must be **low** and E3 **high**.

PROCESS

Binary to 7 Segment Decoder

Diagram:

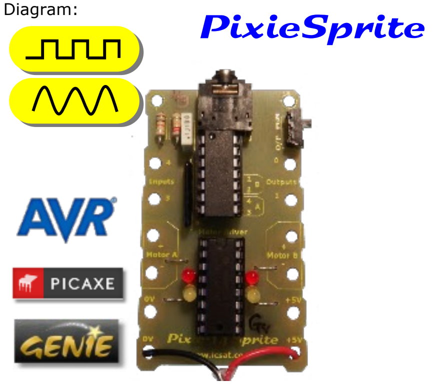


Function:
The 4511 is a BCD to 7-segment decoder driver it converts a binary coded decimal, into signals which will drive a 7-segment display. The display shows the decimal numbers 0-9 and is easily understood.

PROCESS

Pixie Sprite

Diagram:

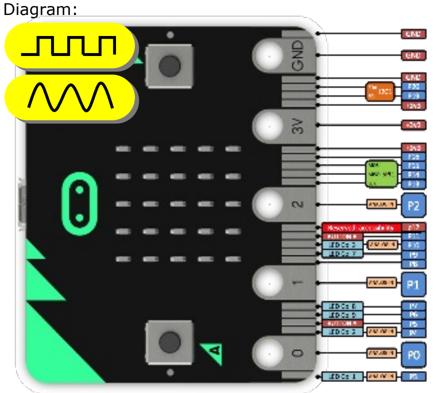


Function:
The Pixie Sprite is a PICAXE, Genie or AVR based programmable controller, can drive 2 motors forwards and backwards at variable speeds. It has 4 IO (Input/Output) pads which allow connections to switches, LDRs, low power LEDs and so on. Using 'croc leads' for quick and simple connections.

PROCESS

BBC Micro:bit

Diagram:

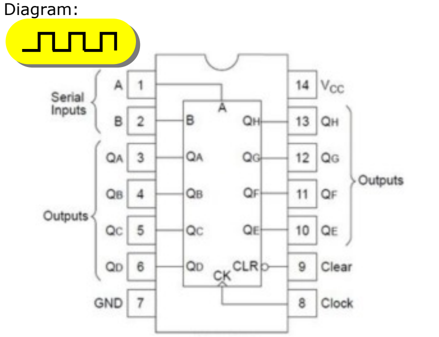


Function:
The Micro:bit is a small all-in-one computer, which has a range of inputs & outputs, more connections are available by using a connector. It has a 5x5 set of LEDs and two switches, along with built-in Bluetooth, accelerometer & compass. **Important it uses a 3V power supply.**

PROCESS

Shift Register

Diagram:



74HC164

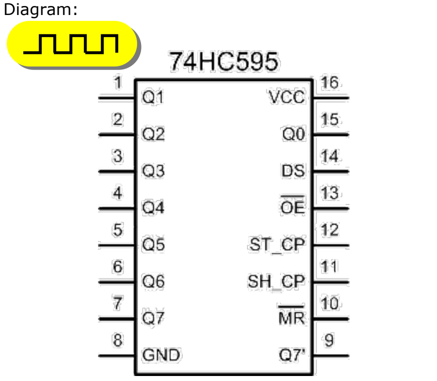
In use the serial inputs A and B are connected together.

Function:
The '164 is an 8 bit shift register, data at the serial input is shifted into the register 1 bit at a time, it takes 8 clock pulses to transfer one byte (8 bits) into all the correct outputs. Great for making 8 bit or 7 segment displays using a small number of pins. The clear pin is held high unless you control it for normal operation.

PROCESS

Shift Register

Diagram:



Function:
The data input is pin 14, clock input is pin 11, pin 12 transfers the data to the outputs, pin 10 is the clear pin which is held low unless you control it for normal operations. Pin 13 controls the outputs, which is normally held low unless you control it. Great for making 8 bit or 7 segment displays.

PROCESS

Logic Gate Families

Diagram:

	TTL 74LSxx	CMOS 74HCxx	CMOS 74HCTxx	CMOS 4xxx
Supply voltage	5V	2V-6V	5V	3V-15V
Supply current	5mA	<1mA	<1mA	<1mA
Output current	8mA	4mA	5mA	8mA
Battery suitable	N	Y	Y	Y
Available functions	V high	High	High	High

Function:

All logic gates, and other functions including microcontrollers are now CMOS type devices. Care needs to be taken when handling them - **static can damage** them, also you must ensure the **correct voltage** is used, always check the Datasheet.

PROCESS

NOT gate

Diagram:

NOT gate truth table

Input	Output
0	1
1	0

Typical chips:

TTL	CMOS
74XX04, 74XX14	4049UB, 40106B

Function:

A **NOT** gate always produces an output that is opposite to input, it is sometimes called an **INVERTER**, due to the way it functions.

A **truth table** is used to summarise it's function.

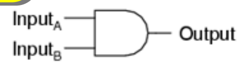
Off = logic 0=0V & On = Logic 1=+V

PROCESS

AND gate

Diagram:

AND gate



A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

Typical chips:

TTL	CMOS
74XX08	4081B

Function:

The **AND** gate only gives a logic 1 when both it's inputs are at logic 1. Used in logic circuits to trigger an output if both inputs are ON.

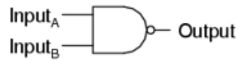
AND gates with more 2 inputs are available if needed, or can be made up from more gates.

PROCESS

NAND gate

Diagram:

NAND gate



A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

Typical chips:

TTL	CMOS
74XX00	4011B

Function:

The **NAND** gate only gives a logic 0 when both it's inputs are at logic 0. It is the opposite function to an **AND** gate, the bobble on the output indicates the **NOT** gate.

NAND gates with more 2 inputs are available if needed, or can be made up from more gates.

PROCESS

OR gate

Diagram:

OR gate

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

Typical chips:

TTL	CMOS
74XX32	4071B

Function:

The **OR** gate only gives a logic 1 when any of it's inputs are at logic 1. It is used in circuits where the output is triggered by any logic 1 on any of it's inputs.

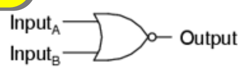
OR gates with more 2 inputs are available if needed, or can be made up from more gates.

PROCESS

NOR gate

Diagram:

NOR gate



A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0

Typical chips:

TTL	CMOS
74XX32	4001B

Function:

The **NOR** gate only gives a logic 1 when both it's inputs are at logic 0. It is the opposite function to an **OR** gate, the bobble on the output indicates the **NOT** gate.

NOR gates with more 2 inputs are available if needed, or can be made up from more gates.

PROCESS

XOR gate

Diagram:

Exclusive-OR gate

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

Typical chips:

TTL	CMOS
74XX86	4070B

Function:

The **XOR** gate only gives a logic 1 when both it's inputs are different. It is used to detect when 2 inputs are **NOT** the same.

PROCESS

XNOR gate

Diagram:

Exclusive-NOR gate

A	B	Output
0	0	1
0	1	0
1	0	0
1	1	1

Typical chips:

TTL	CMOS
N/A	N/A

Function:

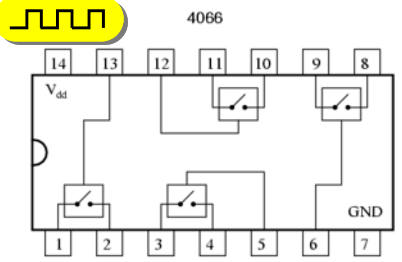
The **XNOR** gate only gives a logic 1 when both it's inputs are the same. It is used to detect when 2 inputs are the **SAME**.

This function must be made using an **XOR** gate and a **NOT** gate, as no chip is available for this function.

PROCESS

Bilateral Switch

Diagram:



Typical chips:

TTL	CMOS
N/A	4066B

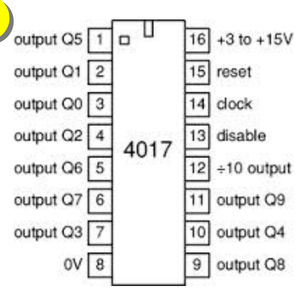
Function:

The bilateral switches can be open and closed using logic levels, **0 = open**, **1 = closed**. The switches can carry either digital or analogue signals so long as they **don't exceed** the supply voltage and have small currents <10mA. Useful for selecting different signals / voltages for input circuits.

PROCESS

Decade counter

Diagram:



Typical chips:

TTL	CMOS
N/A	4017B

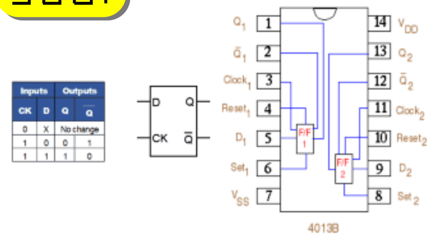
Function:

Decade counter that makes each output logic 1(others are logic 0) in response to the pulses at the clock input. Pins 13 and 15 must be logic 0 for it to count. It can be made to reset at any number by connecting the required output pin to pin 15. The clock input pin 14 can be driven by **any** pulse producing circuit.

PROCESS

D Type Flip flop

Diagram:



Typical chips:

TTL	CMOS
74HCT74, 74HC74	4013B

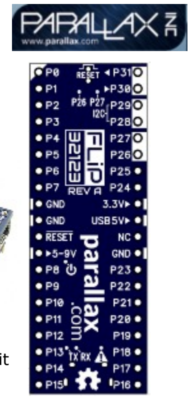
Function:

The D(ata) flip flop is the basic digital data latch, it remembers the state of the input. When it is clocked (pulsed) at the CLK input it will store the value at it's D input at the Q output. The ***Q-bar*** output is always the opposite of Q.

PROCESS

Parallax FLIP microcontroller

Diagram:



The FLIP module is the same size as a 40 pin chip and will fit a chip holder or fit on a small

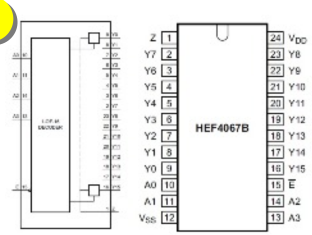
Function:

The Parallax FLIP is a high performance microcontroller, that offers a wide range of pins and functions. It is programmed using BlocklyProp, which is similar to Scratch and Blockly on the PICAXE as a graphical coding method along with C and Parallax's Spin language.

PROCESS

Analog Demultiplxer/Multiplexer

Diagram:



Note: Useful to read more than 1 analog input with only 1 ADC input

Typical chips:

TTL	CMOS
74HC4067, 74HCT4067	4067B

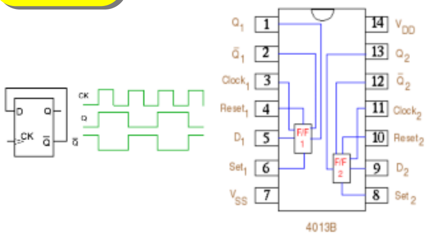
Function:

This is a 16-channel analog multiplexer demultiplexer. It's an IC that can direct an analog signal either from one pin to any one of sixteen pins or the reverse. It can be considered as a digital replacement to those rotary switches that allow you to select one of sixteen positions.

PROCESS

T Type Flip flop

Diagram:



Typical chips:

TTL	CMOS
74HCT74, 74HC74	4013B

Function:

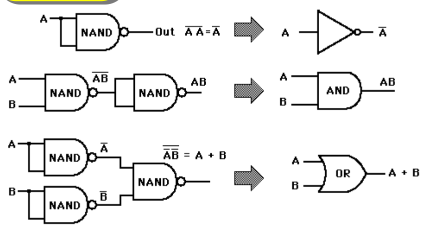
A T(oggle) flip flop is whose output toggles between logic values each time it is clocked (pulsed). It can be created using D type flip flops and is the basic building block of counters.

Useful to create once **ON** and once **OFF** digital toggle switches.

PROCESS

Gates from NAND gates

Diagram:



Typical chips:

TTL	CMOS
74XX00	4011B

Function:

The NAND gate is the universal logic gate, this is because it can be used to make all the other gates.

It also helps in minimizing the number of gates, if a logic circuit is converted into NAND gates, often this leads to the removal of duplicated gates & functions.

PROCESS

N MOSFET Power Driver

Diagram:

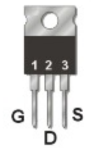
Note

IRF510 5.6A

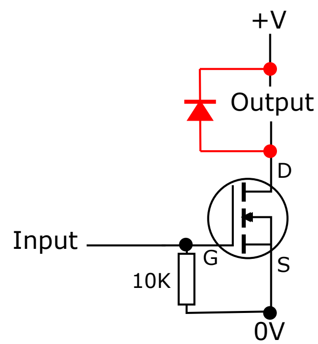
IRF530 8A

IRF540 14A

Pinout



Same for all 3



Flyback diode in red

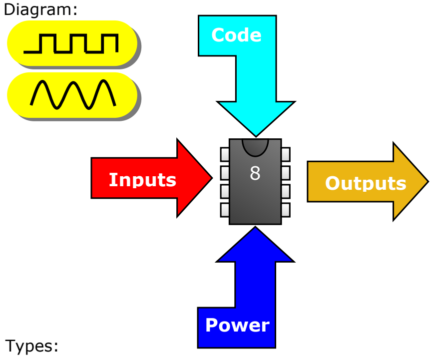
Function:

To drive high current output devices the best method is to use a high power N MOSFET. If your output device is electromagnetic, such as a motor you will need to fit the 'flyback' diode to protect the transistor against the large back emf voltage produced when the motor etc is turned off.

PROCESS

Basic 8 pin microcontroller

Diagram:



Types:

PICAXE, Genie, ATTiny

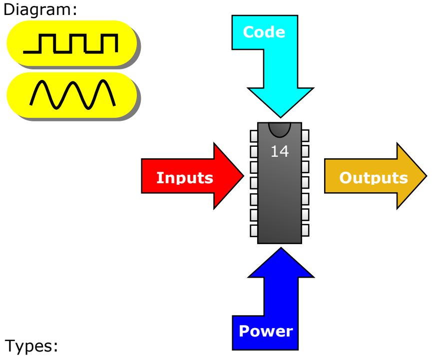
Function:

A basic 8 microcontroller is a key **Programmable Component**. It has a number of input (digital & analogue) and output pins. The actions of these pins is controlled by the program uploaded to it, which you have to design/code.

PROCESS

Basic 14 pin microcontroller

Diagram:



Types:

PICAXE, Genie, ATTiny

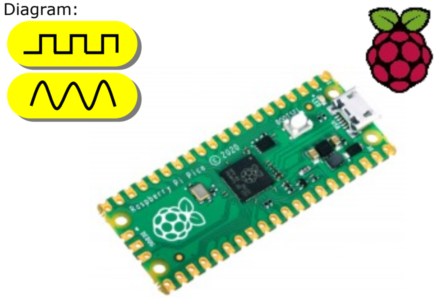
Function:

A basic 14 microcontroller is a key **Programmable Component**. It has a larger number of input (digital & analogue) and output pins. The actions of these pins is controlled by the program uploaded to it, which you have to design/code.

PROCESS

Raspberry Pi Pico

Diagram:



Note: The Pico is a 3V only device, like the Micro:Bit.

Function:

The Raspberry Pi Pico is a high performance microcontroller, that offers a wide range of pins and functions, GPIO, ADC, UART, I2C and SPI. It is programmed using MicroPython, CircuitPython or C/C++, and has a wide range of libraries for add-on's like the Arduino.