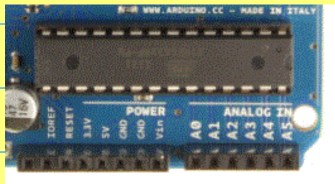
Lecture 5 CS3514 26/9/17

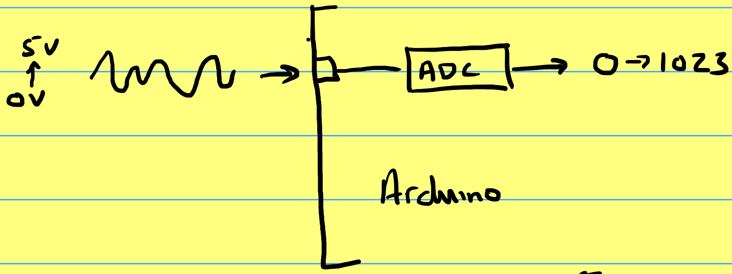
Inputting and processing Analog signals

We appear to live in an analog world. Light, Sound, touch, taste, smell are all perceived as continuous and varying in degree. Our digital computer, in contrast, models the world more coarsely — in discrete units. Converting an analog signal to a range of discrete numbers is the job of the Analog to Digital Converter (ADC).

The Arduino has 6 Analog Inputs $A_0 - A_5$. Attached to each of these pins is an ADC



Each ADC can accept a Signal voltage, varying from 0v to 5v. The o/p of an ADC is a number in the range 0 to 1023



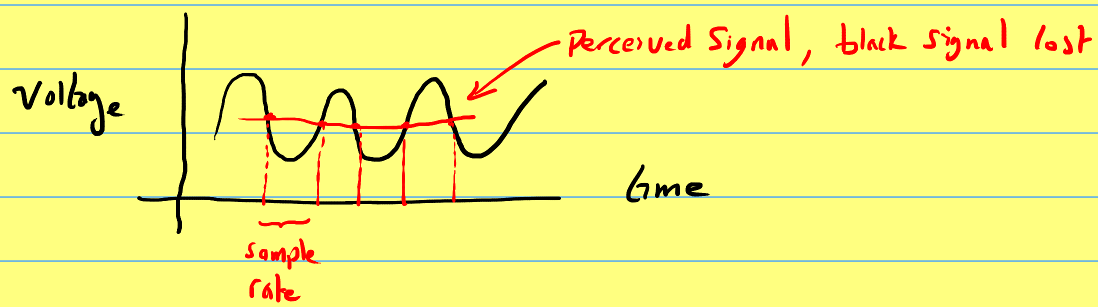
The resolution of the ADC is $\frac{5}{1023}$ = 0.0048 Volts per division
= 4.8 mV / division

That is, the difference between two successive ADC output values corresponds to a change of 4.8 mV in the Input Signal. There are techniques for improving this resolution. We will examine these later.

The Arduino ADCs use 10 bits to encode the input Signal thus we have 1024 (2^{10}) values over the input voltage range. Other ADCs may use more, or less, bits

Note: Our Input Signal is a varying voltage, not a varying current. If we have a sensor that varys its o/p current in response to sensed changes in its environment, then we must provide external circuitry to convert the varying current into a varying voltage before connecting it to the ADC.

The sample rate of an ADC can be important for high frequency input signals. If the input signal varys more quickly than it can be sampled, information will be lost:



To read an analog input, we use the <u>analogRead()</u> function

This function takes one parameter, the input pin to be read, and returns a number between 0 and 1023.

Note: No voltage greater than 5v should be applied to an input pin. If necessary, external circuitry needs to be constructed to enforce this constraint.

<u>Different Sensors exhibit different Interfaces</u>

* Some sensors (usually powered) produce a varying voltage in an appropriate range and can be directly connected to the input pin of an ADC.
* Some sensors change resistance change resistance under certain conditions eg., a light dependant Resistor (LDR), a flex resistor, or a

pressure sensitive resistor.

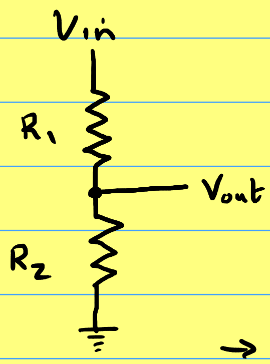
* other sensors react by varying their o/p current.

To connect sensor that change their resistance under changing conditions, we typically use a voltage divider

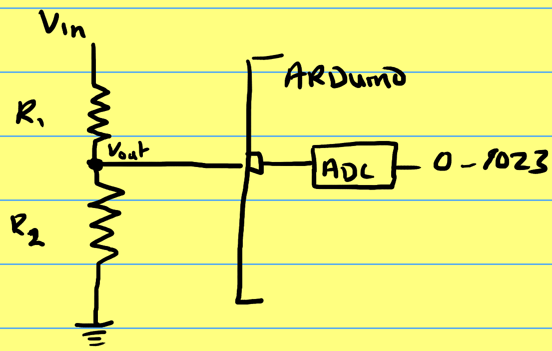
Voltage Divider

A voltage divider divides the voltage in a circuit by causing it to drop across a number (usually 2) series resistors. The voltage drops across these resistors in proportion to their relative values.

Thus:



V_{out} is the voltage at the junction between R_1 & R_2. We label it as V_{out} because this is typically the voltage that we are sampling with our microcontroller
→ it is the voltage out of the voltage divider but it is the voltage in to the ADC



V_{in} can be 'any' voltage value (in our cases it will be 5v) but R_1 & R_2 should be chosen appropriately so that V_{out} is never greater than 5v.

It can be shown that

$$V_{out} = V_{in} * \frac{R_2}{R_1 + R_2}$$

This follow directly from ohm's Law :

$$V_{in} = I . (R_1 + R_2)$$
$$V_{out} = I . R_2$$

Since I is the same everywhere in the circuit

$$\frac{V_{in}}{R_1 + R_2} = \frac{V_{out}}{R_2}$$

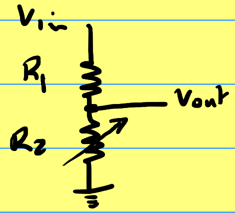
$$\therefore \quad V_{out} = \frac{R_2 * V_{in}}{R_1 + R_2} = V_{in} * \frac{R_2}{R_1 + R_2}$$

Note : * if $R_2 = 0$, $V_{out} = 0$

* if $R_2 = R_1 \Rightarrow V_{out} = \frac{1}{2} V_{in}$

* if $R_2 >> R_1$ then $\frac{R_2}{R_1 + R_2} \rightarrow 1$ and $V_{out} \rightarrow V_{in}$

To vary V_{out}, we typically make R_2 a variable resistor or a sensor whose resistance changes under changing environmental conditions. In that case we draw R_2 as follows:



The arrow through the resistor indicates that its value is variable.