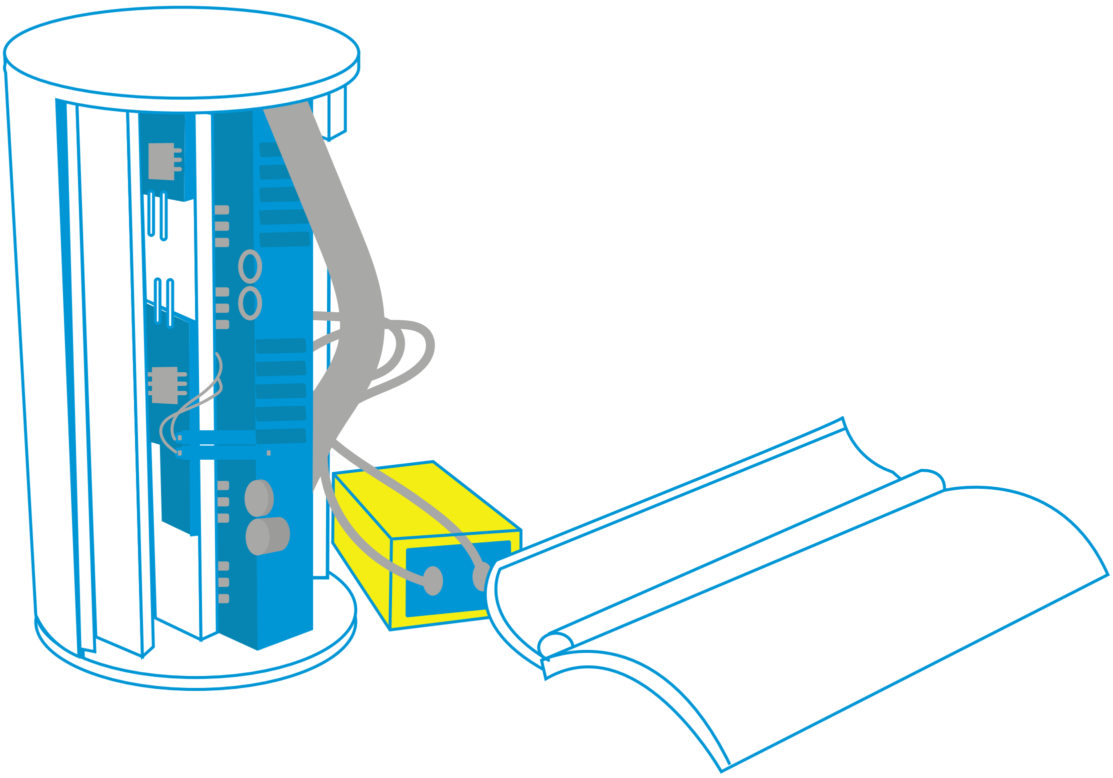
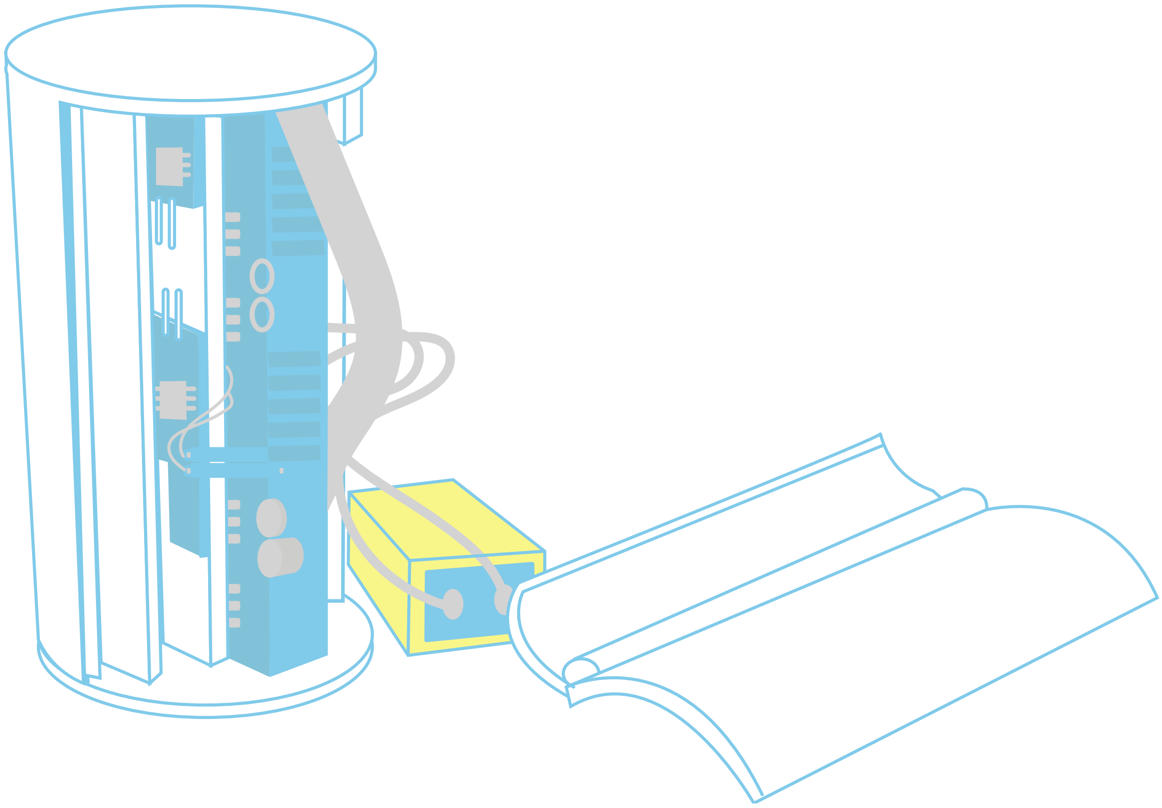
teach with space

→ GETTING STARTED WITH CANSAT

A guide to the Primary Mission





teach with space – getting started with cansat | T08
www.esa.int/education

The ESA Education Office welcomes feedback and comments
cansat@esa.int

An ESA Education production
in collaboration with ESERO Ireland and ESERO Belgium

→ GETTING STARTED WITH CANSAT

A guide to the Primary Mission

Fast facts

Age range: 14 - 20 years old
Curriculum links: electronics, programming, mathematics
Complexity: Medium
Lesson time required: 90 minutes
Methodology: Project based learning
Supporting resources: Meet Arduino! Radio communication, Parachute design
Keywords: Sensors, Resistor, Radio, Communication, Protocols, Soldering, CanSat

Outline

This module outlines the main features of the Primary Mission for CanSat. In the Primary Mission, teams must measure the temperature and pressure and send the information to their ground station. Students will learn about the differences between the sensors they can use and about the challenges associated with completing the Primary Mission. This module is designed in line with a range of resources to support the entire CanSat mission.

Students will learn

- The basic knowledge required in order to assemble and perform CanSat's Primary Mission
- How sensors work: thermistor & atmospheric pressure sensor
- Basic electronics: Ohm's law
- How to collect data from a resistance based sensor – using a voltage divider circuit
- Soldering

→ Summary of activities

Summary of activities					
	Title	Description	Outcome	Requirements	Time
1	The basic components	Students are introduced to all the essential components of a CanSat.	Students will be able to choose which sensors are most suitable for a CanSat and explain why.	None	30 minutes
2	Basic electronics	Students are guided through the application of Ohm's law to resistors and a voltage divider circuit.	Students will become familiar with how resistance-based sensors work and their purpose in a CanSat.	Previous activities	15 minutes
3	Communicating with your CanSat	This activity includes information on how a CanSat communicates with a ground station, and communication protocols for electronics.	Students will learn the basics of wireless communication and how circuit components communicate with one another.	Previous activities	20 minutes
4	Putting it all together	This activity summarises the assembly of a CanSat: fitting components together, soldering, power, casing.	Students will appreciate the importance of a good technique when soldering, and be able to select all the components to suit the Primary Mission.	Previous activities	25 minutes

Introduction

The CanSat competition has two main challenges: the Primary Mission and the Secondary Mission. In the Primary Mission, CanSat teams must record the air temperature and pressure with the CanSat and send the data to the ground based station. The Secondary Mission is an open challenge for which teams are asked to design their own investigation using the CanSat. To successfully complete the Primary Mission, teams need to understand the basic electronics involved and how they can use sensors to measure temperature and pressure. The main objective of this guide is to provide this ground knowledge.

By following this resource and identifying the elements needed to complete the Primary Mission your team will have all the information they need to get started in the ESA CanSat challenge!

CanSat Primary Mission

The team must build a CanSat and program it to accomplish the following compulsory primary mission:

After release and during descent, the CanSat must measure the following parameters and transmit the following data as telemetry to the ground station at least once every second:

- Air temperature
- Air pressure

In your final CanSat this will likely be part of a much more complex circuit, with components also related to your secondary mission.

It is best to first set up circuits using a solderless breadboard. After checking the circuit and code to make sure it works, the components can be soldered in place on the sensor board (Arduino shield).

At the end of this resource you can find several links to websites that stock the various components we will discuss.

Activity 1: The basic components

This activity gives students an overview of the key components required for the CanSat Primary Mission. This allows students to appreciate the complexity of the CanSat mission, by considering the different options available for each component.

Exercise 1

1. **Can you think of any additional problems with using a thermistor to measure the air temperature?**

   When a current flows through a resistor, heat is generated. This means that the temperature being measured will be higher than the ambient temperature of the surroundings, due to self-heating of the resistor. This can be even more important if the temperature sensor is placed near other components, such as the CPU, as they also generate heat.

Exercise 2

Students are asked to complete a table with information about various sensors, including the BMP280. They should be encouraged to conduct their own independent research, for example using the internet and datasheets. They could explore different types (pressure, temperature) and different models of the same type, such as comparing two temperature sensors.

Activity 2: Basic electronics

Now that students are familiar with the key components of the CanSat Primary Mission they are ready to learn about how these components work. This activity provides an introduction to Ohm's law as well as information on how to calculate the resistance of a resistor and set up a voltage divider circuit.

Exercise

1. **What is the resistance of the resistor below?**

   Using the chart, this means the resistance is 15 x 100Ω or 1500Ω

Bonus Exercise

In this exercise students must make use of their mathematics skills to rearrange and combine the two equations given below to give an expression for $\mathbf{V_{out}}$.

$$V_{in} = I(R1+R2) \quad \text{and} \quad V_{out} = I(R2)$$

The first step is to make **I** the subject of both equations:

$$I = \frac{V_{in}}{(R1+R2)} \quad \text{and} \quad I = \frac{V_{out}}{R2}$$

Now, we can replace the I from one equation with the expression from another, like so:

$$\frac{V_{in}}{(R1+R2)} = \frac{V_{out}}{R2}$$

Finally, we can rearrange the equation to make $\mathbf{V_{out}}$ the subject (by multiplying by **R2**). This gives us:

$$V_{out} = \frac{V_{in}R2}{(R1+R2)}$$

This equation allows us to calculate the output voltage of a voltage divider circuit, provided we know the input voltage and the value of the two resistors. This is the basic principle on which many sensors are based.

Activity 3: Communicating with your CanSat

This activity brings together the previous work by looking at how we communicate with our CanSats. The students should be ready to prepare the electronics needed to carry out the Primary Mission, but there is one vital step missing! The information that the CanSat collects must be sent to a ground station. To be able to do this, we need to understand how electronics communicate and take a look at the components we can use to communicate with them.

Activity 4: Putting it all together

In activity 4, students learn how to fit the components of the CanSat Primary Mission together, using solder boards and soldering. An introduction to the soldering technique is given. Students are made aware of the safety precautions that should be followed when soldering. Information on powering the CanSat is also presented, and the important considerations when deciding how to power the CanSat are briefly discussed.

Exercise 1

1. **Why are solar cells the preferred option for satellites, and why might they be less useful for your CanSat?**

   Satellites remain in orbit for long periods of time and therefore require an indefinite power source. The Sun is a great source for this. But powering your CanSat in this way is problematic. Firstly, there is a size and weight restriction, which makes building a large enough solar panel difficult. Secondly, since much of the radiation from the Sun is absorbed by the atmosphere, solar panels on the ground are much less effective than those in orbit.

Exercise 2

Students are asked to complete a table with their chosen components, a reason for their choice and a back-up option. The final exercise should motivate students to discuss the advantages and disadvantages of the components, as individual parts, and justify their decisions on using them in a CanSat.

Discussion

The main aim of these activities is for students to familiarise themselves with the basic components and sensors that can be used to build a CanSat. They should appreciate the variety of sensors that are available for each purpose, and how each one has its advantages and disadvantages. As a result, they should be able to make informed decisions on which components would be suitable for their CanSats.

You can develop the discussion at the end of Activity 4 to prompt the students to consider whether they would change any of their choices, or use their next best alternative, if they take into account the fact that all of these components need to work and as one whole integrated system in a CanSat. There are additional factors, such as the overall size and weight restrictions for the European CanSat Challenge that may require compromises to be made. At this point, you could make links to the mission objectives in the CanSat guidelines and in particular, discuss which objectives should be prioritised (e.g. the compulsory primary mission versus your own secondary missions).

→ GETTING STARTED WITH CANSAT

A guide to the Primary Mission

→ Activity 1: The basic components

Introduction

The essential components needed to complete the CanSat Primary Mission are divided between your CanSat and your ground station. Whilst your CanSat will fly on a rocket, balloon or drone, your ground station will be installed on ground. This is where you will receive the data from your CanSat using an antenna.

CanSat components:

- A suitable microprocessor or computer (such as an Arduino or Raspberry Pi)
- A temperature sensor (such as a thermistor)
- An atmospheric pressure sensor (such as an MPX4115A)
- A wireless transceiver (such as an APC220, X-Bee or LoRa)
- An antenna (normally a quarter wave antenna)
- A parachute or similar device to provide a safe, controlled landing
- A power source

Ground Station components:

- A wireless transceiver
- An antenna (normally a Yagi antenna)
- A suitable microprocessor or computer (such as an Arduino, Raspberry Pi or your laptop)

Except for the parachute, the components of your CanSat must fit inside the volume of a 330ml soft-drink can (max diameter = 66mm; max height = 115mm; mass = 300-350g) until after launch. GPS modules and radio antenna can be mounted externally to the top or bottom of the can, providing size restrictions are not exceeded.

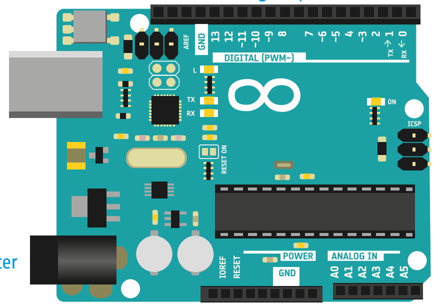
Note: Have a look at the guidelines for all the requirements.

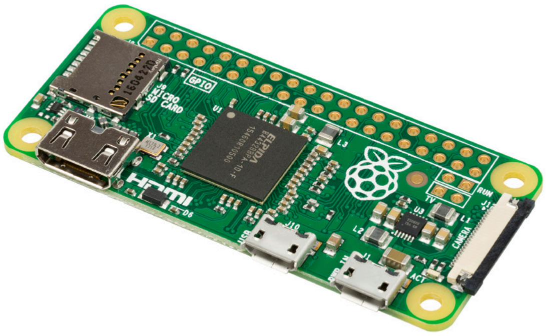
Microprocessors and computers

Microprocessors have a wide range of functions. A microprocessor differs from other products that you might be familiar with (such as a Raspberry Pi) as it requires an input from a computer before it can function and is not a self-contained device. A popular type of microprocessor is an Arduino. Once you have uploaded your code and powered the microprocessor it can function independently of a computer!

A Raspberry Pi differs from a microprocessor as it is a computer. On the Raspberry Pi board is everything it needs to function and run code. An on-board CPU means that Raspberry Pi's can offer greater computational power than a microprocessor. A Raspberry Pi Zero is a miniature, low cost Raspberry Pi and is a common choice for CanSat projects. The choice of microprocessor or miniature computer is again up to you! You will have to consider the compatibility with the sensors you want to use as well as the programming language you are most confident with. At the end of this resource are links to the Arduino and Raspberry Pi websites, where you can find out more about both options.

Shown below are an Arduino Uno and a Raspberry Pi Zero, both typical boards used in CanSats.





The Arduino Uno board can be split into five major components:

1. **Digital pins** – There are 13 pins configured for digital inputs*. 6 of these (3, 5, 6, 9, 10 and 11) are PWM pins. More information about PWM can be found in the Radio Communication Resource
2. **USB input** – used to connect the Arduino to a computer
3. **AC adaptor** – To connect greater than 5V to the Arduino
4. **Power pins** – These pins can be used to supply up to 5V to the Arduino
5. **Analog pins** – There are 6 pins configured for analog inputs*

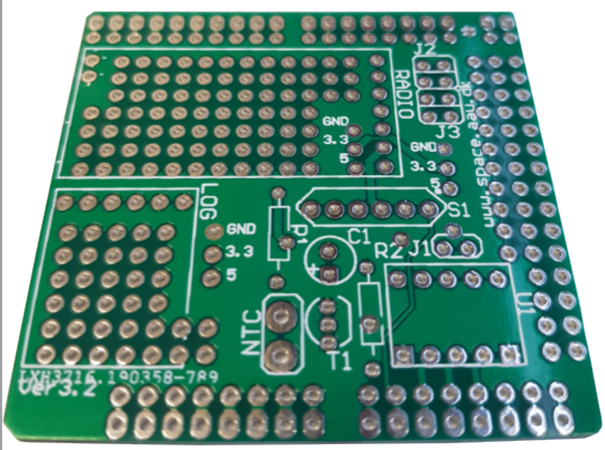
*By default they are inputs but can also be defined to be an output; for more information, see: https://www.arduino.cc/en/Tutorial/DigitalPins

The breadboard

Whilst you are learning the basics of Arduino and sensors it will be best to use a solderless breadboard, as any mistakes you make building your circuit can be easily changed. A breadboard is a simple tool that can be used to wire electrical components together.

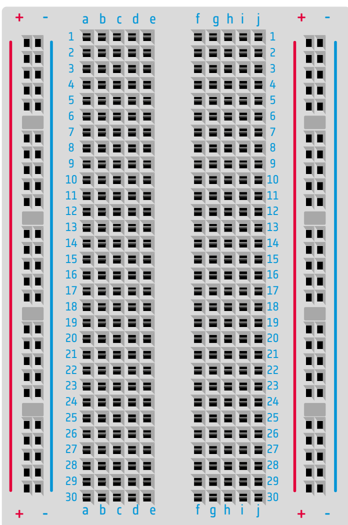
Note: Have a look the guidelines for all the requirements.

Figure A1



↑ A solder breadboard

Figure A2



↑ A solderless breadboard

Pins on electrical components can be placed into the terminals on the board. Centrally, rows are connected. This means for example, that the two pins of a resistor should be placed in different rows, otherwise it will form a closed circuit with itself.

It's very important to make a sketch of your circuit before connecting and powering the circuit, because you will risk breaking the components. The outer columns of the board are connected in columns, rather than rows. Typically, these are used to provide ground and voltage connections to reduce the complexity of the setup.

When you assemble the final version of your CanSat you will need to use a typical solder breadboard. We will look into this in Activity 4!

The temperature sensor

Temperature sensors can be divided into these main categories:

- Thermistors
- Analog sensors
- Thermocouples

A typical two legged thermistor is a negative temperature coefficient thermistor (NTC). It works on the principle that a change in temperature will cause the electrical resistance of the thermistor to change – the NTC part means that as the temperature increases, the electrical resistance decreases (and vice versa). The change in resistance can be directly measured with a multi-meter, but to be input to the Arduino we must convert the change in electrical resistance to a change in voltage – we will look into this later.

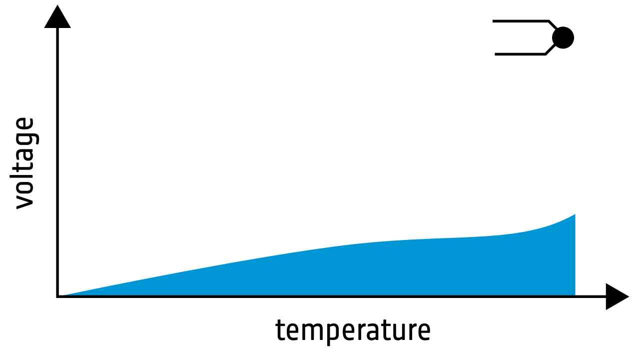
Analog sensors are cheap and require no calibration. They use solid-state techniques to determine the temperature, rather than temperature sensitive resistors. We don't need to delve into the physics of how these sensors work, but if you're interested you can find out more here[1].

Thermocouples measure the temperature by using the thermoelectric effect present between two different metals. This effect was discovered by Thomas Seebeck. Again, the underlying physics here isn't important, but you can find out more here.

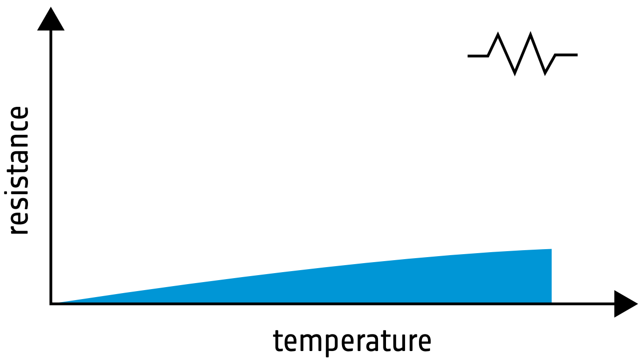
[1] You can find all the links at the end of this resource

The graphs show how the temperature affects the voltage and resistance in the three types of temperature sensors we've discussed. A further comparison of NTC thermistors and analog temperature sensors can be found here.

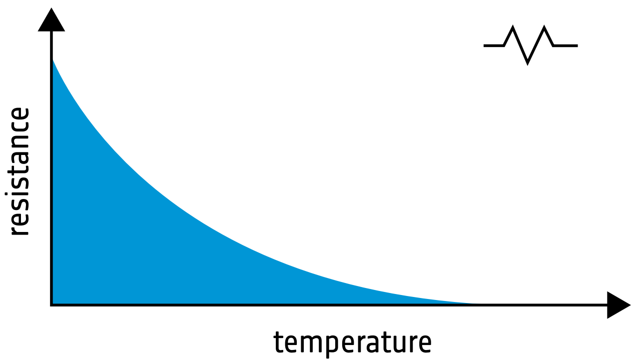
THERMOCOUPLES



RTD



THERMISTOR



When deciding which sensor to use in your CanSat you should compare the technical specifications using the manufacturers data sheet. The sensor you choose should best fit your mission aims. You might also have to consider the number of available pins. For example, an Arduino Uno board has more digital pins than analog pins. Depending on your secondary mission you might have to prioritise these pins.

Exercise 1

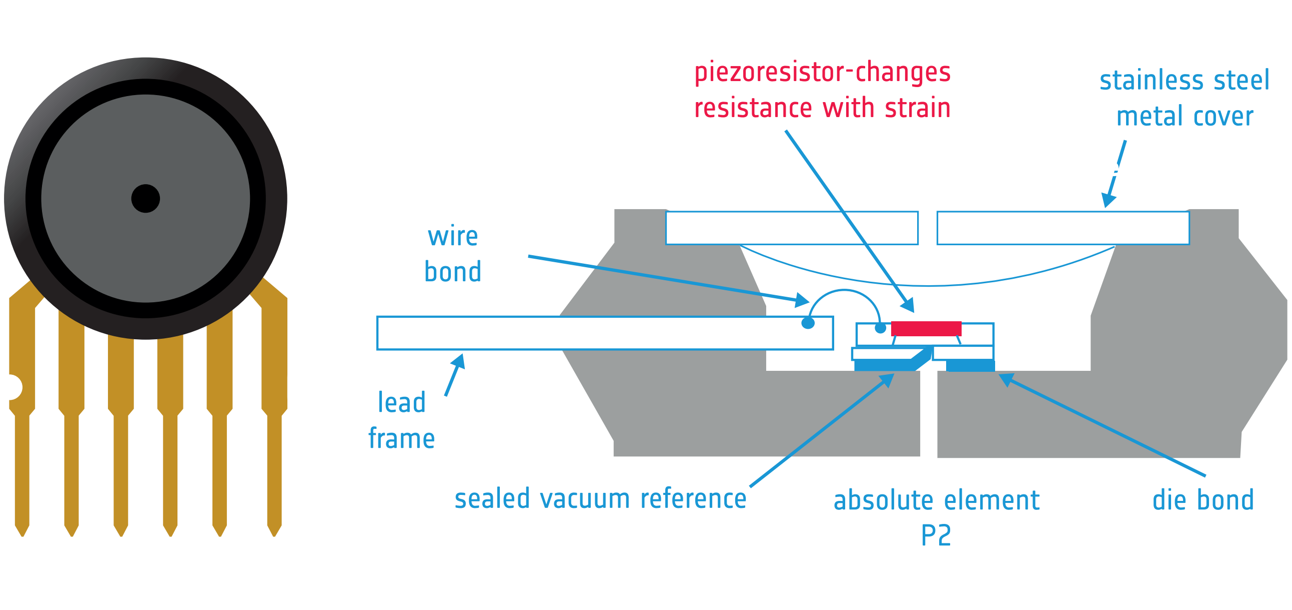
1. Can you think of any additional problems with using a thermistor to measure the air temperature?

Hint: What is generated when a current flows through a material (resistive or not)?

Atmospheric Pressure Sensor

The MPX4115A atmospheric pressure sensor is a 6-legged component, of which only 3 legs are used for measurements. This sensor is commonly used in CanSats. The sensor is made by having a small **piezoresistor** seal a cavity. If you are using a different sensor you should look at the data sheet provided by the manufacturer for further information.

Figure A3



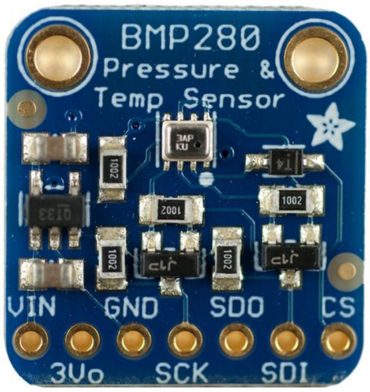
↑ The MPX4115A atmospheric pressure sensor and a schematic of the internals of the sensor, showing the piezoresistive elements

But...What is a Piezoresistor?

A piezoresistor is a resistor whose resistance varies when it is under a mechanical strain, such as stretching or bending. As atmospheric pressure changes the piezoresistor bends slightly. This 'bending' causes strain in the piezoresistor which causes it to change its electrical resistance.

In the MPX4115A sensor, the change in resistance is converted into a voltage within the sensor package itself. This voltage can then be inputted by the user in to the Arduino board. Again, there are many different types of pressure sensor available. Whilst the MPX4115A is perhaps the most common, it might not be the best fit for your project!

Other sensors



A very popular alternative to the temperature and pressure sensors discussed is the BMP280 digital pressure sensor. This sensor combines a pressure and temperature sensor into one unit, making it an ideal solution for the CanSat Primary Mission. The BMP280 sensor is cheap and very small. This means you can complete the Primary Mission whilst using very little of the available budget and size for your CanSat, making more exciting secondary missions possible!

The sensor is typically capable of measuring to 1 hPa of pressure and 1ºC of temperature. Perhaps the greatest advantage of using such a popular sensor is that you are well supported should you run into trouble! The temperature sensor on the BMP280 is close to other electrical components that will carry currents. As you know, anything that carries a current will have a resistance, and therefore generate some heat, so the reported temperature can be higher than the ambient temperature – you should account for this in your measurements!

Important considerations when choosing a sensor:

Sensitivity: what is the minimum change that can be measured by the sensor?

Response time: how quickly does the sensor respond to a changing environment?

Linearity: is the response linear (over the range required for measurements)?

Range: what is the min/max value that can be measured by the sensor?

Hysteresis: does the sensor have the same output for the same ambient conditions; e.g. would a temperature sensor output the same values for temperature measured, regardless of whether the temperature was rising or falling? This is a phenomenon that you may have already come across when studying magnetism.

Exercise 2

Research the datasheet of BMP280 , a pressure sensor (e.g. the MPX4115A) and a temperature sensor (e.g. a thermistor) and write down their features in this table below. There are also two empty columns to add more sensors.

	BMP280	MPX4115A	Thermistor		
Working range					
Accuracy					
Response time					
Cost					
Required Power					

→ Activity 2: Basic electronics

Introduction

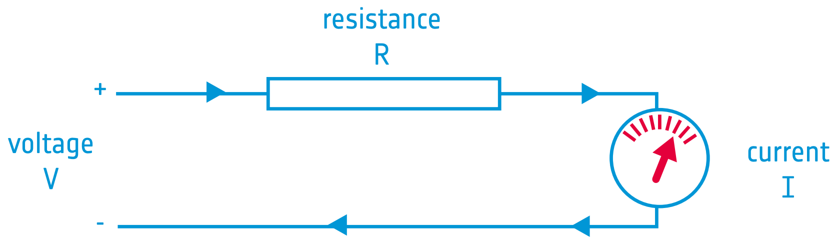
Now that we have an idea of the different types of components available for the Primary Mission, let's look at how the components work. The following section will look at the basic equations used in electronics that the sensors make use of.

Voltage & Current: Ohm's Law

Before you start the Primary Mission it is a good idea to understand the fundamental concepts of electricity. Ohm's law describes how the main features of an electronic circuit are related to one another: voltage, current and resistance.

In normal electrically conducting materials such as copper, gold, silver etc., electrons are able to flow through the material easily – a voltage V (providing potential energy) provides the 'push' needed to keep electrons flowing (flow rate = current I) through a circuit.

Ohm's law allows us to predict the current flowing (I) through a resistor (R) when a voltage (V) is applied to a circuit as in the diagram.



Ohm's law:

$$V = IR$$

which can be rearranged to give:

$$I = \frac{V}{R}$$

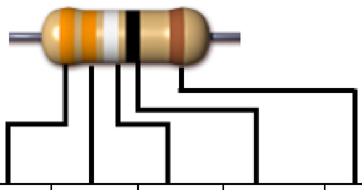
So, if the voltage we apply to the circuit is constant, we can adjust the current flowing by changing electrical resistance.

Resistors

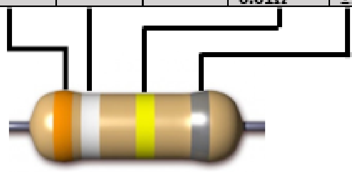
Resistors are a vital piece of any CanSat setup. We have already discussed a special type of resistor, the thermistor. Normal resistors, while providing no useful measurement themselves, can be used to control the voltage and current flowing through the components in your circuit. The resistance of the resistor you use, measured in Ohms, must be suitable. But how do you know what the resistance is? For example, for a typical thermistor, you might use a low value resistor, such as 220kΩ, whilst a typical LED will require a much bigger value resistor, such as 10kΩ. If you don't select the correct resistor, the circuit might not behave as you expect it to.

A simple solution is to use an Ohmeter. This will give you an instant reading. However, you could also calculate the resistance yourself by looking at the coloured bands on the resistor.

The graphic below shows how to read a 4 or 5 band resistor. The 'tolerance' gives an indication of the range you can expect the actual value of resistance to be from the supposed resistance.



Colour	1st Band	2nd Band	3rd Band	Multiplier	Tolerance
Black	0	0	0	1Ω	
Brown	1	1	1	10Ω	±1%
Red	2	2	2	100Ω	±2%
Orange	3	3	3	1KΩ	
Yellow	4	4	4	10KΩ	
Green	5	5	5	100KΩ	±0.5%
Blue	6	6	6	1MΩ	±0.25%
Violet	7	7	7	10MΩ	±0.10%
Grey	8	8	8		±0.05%
White	9	9	9		
Gold				0.1Ω	±5%
Silver				0.01Ω	±10%



For example, the two resistors in the graphic have resistances of 339Ω ±1% (top) and 390,000Ω (390MΩ) ±10% (bottom).

Exercise

1) What is the resistance of the resistor below?

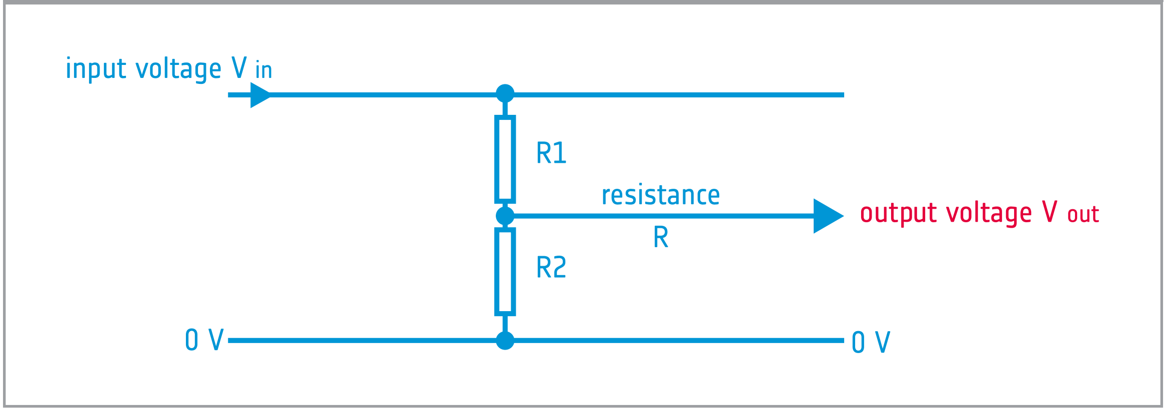
The Voltage divider circuit

Many sensors show a change in electrical resistance proportional to the change in the parameter being measured. For example, the resistance of a thermistor changes in response to a change in ambient temperature. However, the Arduino is not able to directly measure a changing resistance, only a changing voltage. Therefore, for a thermistor to be able to communicate information about the temperature to the Arduino, we need to change the output to a changing voltage instead of a changing resistance.

This is done using a voltage divider circuit. By keeping one resistor a fixed value, the change in voltage that is measured (V_{out}) must be caused by the thermistor. This voltage increase or decrease can then be mapped to an increase or decrease in temperature, using a transfer function. A transfer function is a simple equation that tells us the relationship between the measured voltage and the temperature.

In order to calculate the transfer function, we first need to analyse the circuit, and eventually put the temperature as a function of the voltage. Remember to check the datasheet of the sensor that you are using in order to gather all the information that you need to get your transfer function.

Figure A4



↑ A voltage divider circuit diagram

Either R1 or R2 could be replaced by a thermistor in this circuit.

Bonus Exercise

The current flowing (**I**) through resistors **R1** & **R2** can be calculated using Ohm's law (**V=IR**). Have a go at combining and rearranging the equations below, to give a final equation for V_{out}.

If:

$$V_{in} = I\,(R1 + R2) \quad \text{and} \quad V_{out} = I(R2)$$

I= ____________ and **I=** ____________

Then we can combine the equations to remove **I**:

=

But we want to know how V_{out} depends on the other parameters, so rearranging the equation gives us:

V_{out} = ________________

As V_{in} is fixed (along with **R1**), we can convert a **changing resistance (R2)** to a **changing voltage** using the voltage divider circuit.

Tip: a flowing electric current generates heat in the material it passes through (e.g. thermistor material) and this in turn changes its electrical resistance. To minimise the effect of self-heating it's a good idea to switch on the circuit just before taking a measurement. Take the measurement quickly and then switch off the circuit again after the measurement. This minimises the amount of time that current is flowing so less heat is generated.

→ Activity 3: Communicating with your CanSat

Introduction

You should be ready to prepare the electronics needed to carry out the Primary Mission, but there is one vital step missing! The information that the CanSat collects must be sent to a ground station. To be able to do this, we need to take a look at the components we can use to communicate and how electronics communicate.

Transceivers (or Radio Modules)

We know how to use an Arduino and sensors to collect temperature and pressure data. But how do we receive the information our CanSat collects? We could of course save the data and collect the information when we collect the CanSat. However, in the Primary Mission your CanSat must transmit information to your ground station every second. This is a requirement for two reasons.

Firstly, it gives you a taste of a real satellite mission! Secondly, we can't control all the conditions of a launch, and sometimes this means we can't recover every CanSat. By transmitting information, you can still complete the analysis stage of your project.

Wireless transceivers are used to relay information between a CanSat and ground station. They work in pairs, in a similar way to how you might have used walkie-talkies when you were younger (or now!). Both the CanSat and ground station are fitted with an antenna. The CanSat antenna transmits information and the ground station antenna receives it. In order to avoid disturbance and interference each team in a competition is given their own frequency – just like the channels on a walkie-talkie. This means you only receive the information from your own CanSat, and no other sources. Actually, the word transceiver is a composition of two words - transmit and receive, exactly what the transceiver can do.

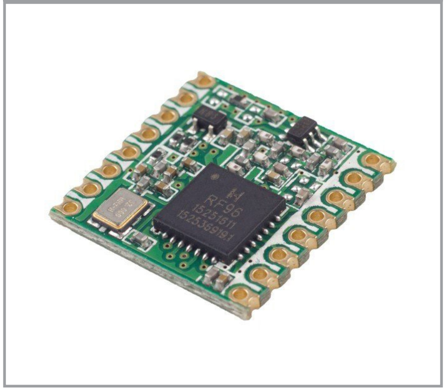
We will focus on the different options you have when choosing a transceiver and what criteria you might want to use to decide which to choose.

When it comes to choosing a transceiver perhaps the most important criteria are the operating frequencies, the power required and the physical size of the transceivers. Of course, you have to also consider the cost of the transceivers. Designing a project often involves some degree of compromise. The perfect components for each job are not necessarily compatible, for one reason or another.

Figure A5

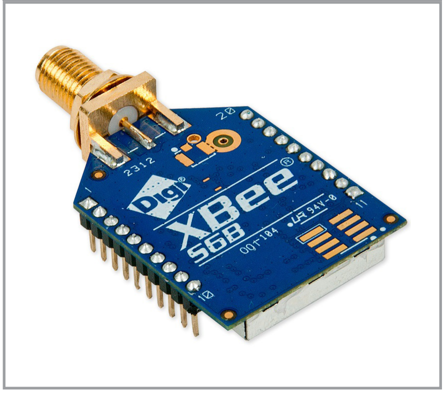
↑The APC220 module

Figure A6



↑ The RFM95 LoRa module

Figure A7

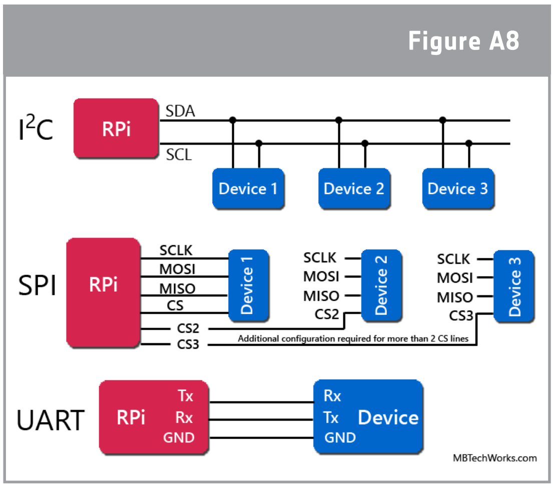


↑ The XBee module

One of the most common choices is the APC220. It is capable of communicating over a distance of 1000m and operates between 418MHz and 455MHz. A popular alternative is a LoRa module (like the RFM95). They generally offer an increased range, up to 2000m, but operate at discrete frequencies rather than over a range like the APC220. The final module that we will briefly discuss is the XBee module. They differ significantly from the APC200 and LoRa modules as they operate in the WiFi range (2.4Ghz) rather than the MHz range. This can bring its own challenges, such as nearby devices causing interference. Depending on the model the range can also vary from 400m to 1600m.

The most appropriate transceiver for your CanSat will vary from team to team. You should study the data sheet of each and use these details as part of your evaluation, whilst also keeping in consideration the rest of your project! For more information, check out our 'Radio Communication' resource.

Communications protocols



↑Schematics of the different communication protocols

All electronics systems use one of several systems to communicate between components. The three main systems we will use in CanSat are UART(Universal Asynchronous Receiver-Transmitter), SPI (serial peripheral interface) and I2C(Inter-Integrated circuit).

UART communication

UART stands for 'universal asynchronous receiver-transmitter'. The main difference between UART and SPI and I2C is the 'asynchronous' part! But, what do we actually mean by asynchronous? As you might have guessed, it is the opposite of synchronous, but let us compare what they both mean for communications.

Synchronous communication is just like calling someone on a telephone. You first dial a number and wait for the other end to pick up. From this moment, when you send data (speak) the receiver is able to immediately receive (hear) and then return their own data (speak) which you can receive (hear). When you want to stop sending data, you send a message to make this clear to the receiver (say goodbye!) and you are then free to complete other tasks.

Asynchronous communication is more like sending a letter. After sending a letter, the transmitter (sender) is able to complete other tasks whilst waiting for a reply. Sometime later, the sender can check his mailbox to see if he has received a reply and act on it accordingly!

UART is widely used and well documented and could therefore be summarised as a simple, easy to use system, but of course, this comes with limitations:

1. One important aspect to note about UART communication is that it is designed for communication between only two devices at a time (not so useful in a complex CanSat). Because the protocol only sends bits indicating the start of a message, the message content, and the end of a message, there is no method of differentiating multiple transmitting and receiving devices on the same line. If more than one device attempts to transmit data on the same line, bus contention occurs, and the receiving devices will most likely receive rubbish unusable data!

2. UART is half-duplex, which means that even though communication can occur bidirectionally, both devices cannot transmit data to each other at the same time. In a project where two Arduinos are communicating with each other via a serial connection, for example, this would mean that at a given time, only one of the Arduinos can be "talking" to the other. For most applications, however, this fact is relatively unimportant and not disadvantageous in any way.

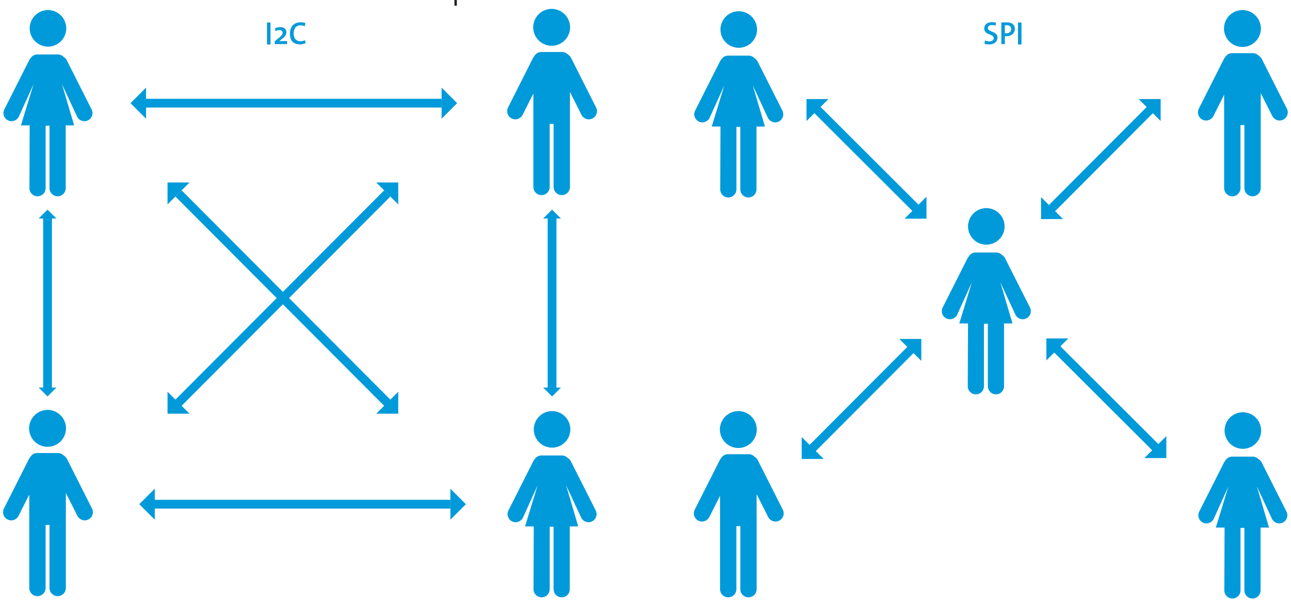
Typical UART CanSat uses are: Sending debug and development messages to a PC, communicating with GPS sensors, communicating with external WiFi and GPRS (3G) modems.

I2C and SPI communication

I2C allows multiple devices (up to 1008!) to be connected to the same I2C interface which is just a pair of wires. It also allows bi-directional communication over these two wires and so is ideal for communicating with many sensors. **Typical I2C CanSat uses are**: "Smarter" sensors (e.g. the BMP 280), Accelerometers, Analog-Digital Converters, Digital-Analog Converters, LCD Screens, Battery Controllers.

On the other hand, **SPI** is the most complex interface that the Arduino hardware supports. As with I2C, it also supports bi-directional communication with several devices but offers a much higher data throughput. This makes it suitable for communicating with the most complex devices that you might connect to the CanSat. **Typical SPI CanSat uses are:** Cameras, storage cards (e.g. SD cards), GPS modules, WiFi Modems.

We are going to use a simple analogy to explain how the different components behave in an SPI system and an I2C system. SPI stands for 'Serial Peripheral Interface'. I2C stands for 'Integer-Integrated Circuit'. We don't need to worry too much about where these names come from, we just need to understand what these protocols mean for our CanSat.



The figure above shows our analogy. On the right is the SPI system, and on the left is the I2C system. In this analogy our electrical components are replaced with people. The arrows show the possible communication between each person. As you can see, in the SPI system there is one component in charge of communications, this is called the master. The other components are called slaves. In the I2C system each component can communicate with one another, such that only one component at any one time can be assigned as the master. The master decides which component to communicate information with. The 'slave' listens to the master and either sends or receives data as requested. In the I2C protocol the situation is dynamic. Any of the components can send the 'listen' command and become the master.

You might think that the I2C protocol sounds much better, but the reality for our circuits is not as simple as the analogy. Whilst I2C is easier to setup, the data transfer is slower than SPI and it generally consumes more power. The choice of communication protocol must be evaluated just like any other part of your project, but you might also find that it is determined by the properties of the sensors you use in your CanSat. The summary table below gives the key points to consider about the protocols.

Table A1

	Protocol		
	I2C	SPI	UART
Setup	Many masters and slaves	1 master, many slaves	1 master, 1 slave
	Simple – 2 pins required	Complex – 4 pins required	Simple – 2 pins required
Data transfer rate	Slow	High	Slow
Energy use	High	Low	High

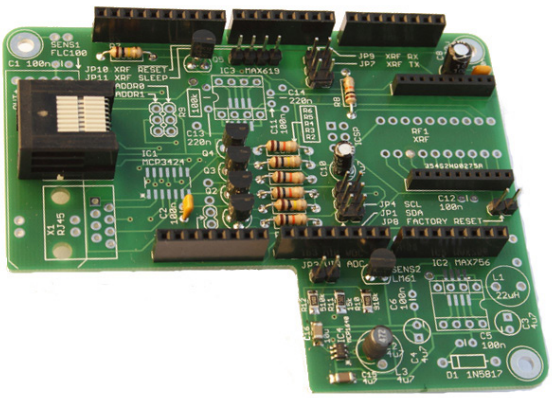
→ Activity 4: Putting it all together

Introduction

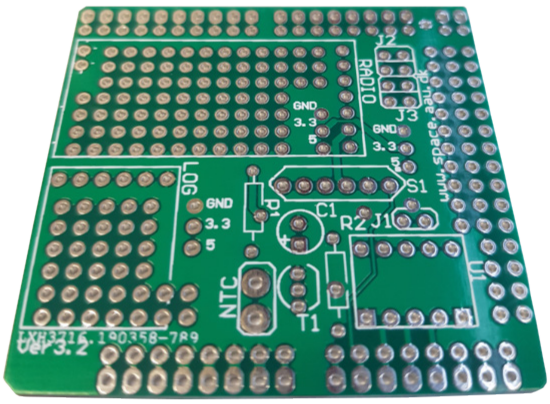
Almost there! With an understanding of all the components that make up a CanSat Primary Mission the only thing that remains is to put it all together! The CanSat competition offers some unique challenges, as you must think carefully about the space your components occupy and how to fit them together. We will now look at some ways to do this and the different ways we can power a CanSat.

The solder breadboard

As you develop your CanSat beyond the Primary Mission you will likely need to connect more and more sensors to your mainboard. This can get messy quickly. Two solutions are to use a breadboard or a sensor shield. A sensor shield acts very much in the same way as a breadboard, but is built specifically to be attached to the main board. You can see an example of a sensor shield below. Sensor shields can however be more expensive than a breadboard, as they are custom made for use with specific microcontrollers.



Breadboards, on the other hand, are very cheap and versatile. They are similar to solderless breadboards that we discussed earlier. The major difference is that electrical connections are formed using solder. Below is an example of a solder breadboard.



Though we are only focusing on the Primary Mission in this resource, you must always keep in mind the direction your project will take in order to make sure what you build is future proof. Choosing a suitable extension for the mainboard is therefore important!

Soldering

The launch of your CanSat involves significant acceleration and force and could cause problems if the connections in your circuitry are weak. To solve this problem, we use soldering. Soldering is used to provide a permanent electrical connection between electrical components. A metal is melted and used to join the connections.

The metal that is used to join the connections is called the solder. It is important that this metal has a lower melting point than the wires or components you are connecting; you don't want these to melt too! Solder is applied to the joint using a soldering iron. It may seem like an unimportant aspect of your project compared to writing the code and choosing your sensors, but a poor solder joint will be one of the first things to fail during a launch, and with an incomplete circuit the entire project could be in jeopardy!

The real advantage of solder joints is that they are much more durable and reliable than a typical breadboard connection. Due to vibrations and g-force shocks during a rocket launch, it is important that the solder joints are of good quality.

Figure A9

↑ An example of a soldering station

Safety:

- Soldering irons typically operate between **300-400°C**.
- **Always wear safety glasses** and make sure not to accidentally place anything near or on the hot iron.
- Un-plug the soldering iron after use and **allow it to cool down** fully before putting it away.

A **good soldering technique** is needed to obtain good solder joints:

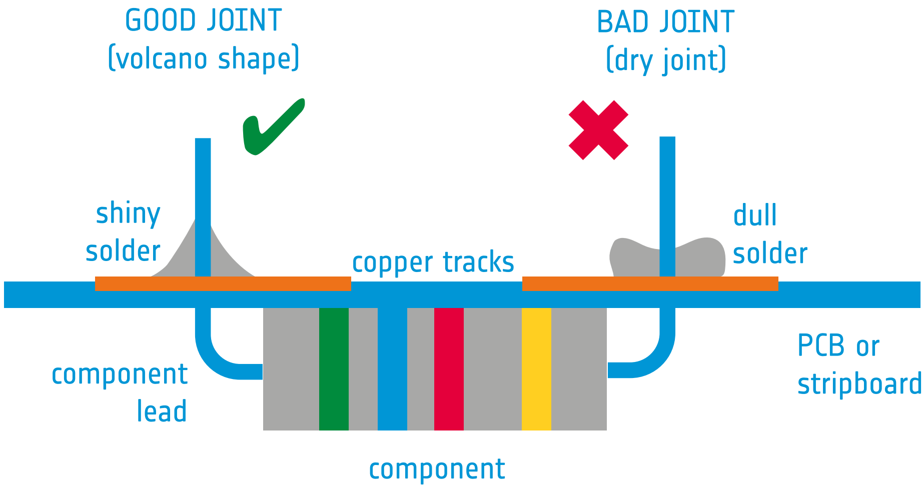
- Bring the soldering iron up to touch the leg of the component to be soldered; wait 2-3 seconds to heat up before applying a small amount of solder.
- Allow the solder to fully melt around the component leg; remove the solder and then the soldering iron.
- Make sure the component cools down before moving on to the next component leg.

A **good solder joint** occurs when the solder 'wets' the surfaces to be joined. The shape should be volcano-like.

Poor soldering joints occur for many reasons including when too much solder is used, the iron heats the solder first (before heating the component leg), or the surfaces are not clean.

Before soldering the components on to the sensor board, it is a good idea to practice soldering using a spare piece of breadboard. A poor solder will be one of the first things to fail during launch and this could stop you collecting any data at all!

You can find more information about the soldering process in the links at the end of this resource.



Powering your CanSat

You now have the basic knowledge you need to start building your CanSat for your primary mission. But there is one important feature of every CanSat we haven't yet discussed, how to power your CanSat! Of course, your CanSat cannot be connected to a USB port during its launch; it must have an internal power source. In a typical satellite, power is provided by solar cells attached to the outside of the satellite. However, this is not possible in the CanSat competition.

Exercise 1

1. **Why are solar cells the preferred option for satellites, and why might they be less useful for your CanSat?**

____

____

____

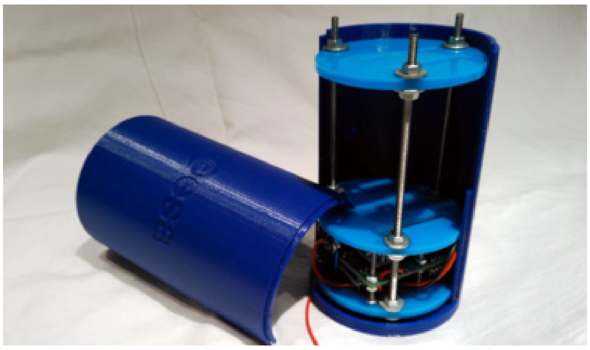
There are a few considerations you need to make when deciding how you will power your CanSat:

- What voltage do you need to supply?
- What battery capacity do you need (mAh)?
- How big (physically) can the battery be?
- How heavy can the battery be?

Many microprocessor boards are capable of delivering either 3.3V or 5V to their connected components, but this does not make it impossible to use a higher voltage, such as a 9V battery, as many boards also have on-board 'regulators' that can drop the voltage to a suitable level. This often comes at a cost however, as energy is wasted in this process. You have to consider if this energy loss is crucial to the success of your mission. In many cases it won't be as the flight time of the CanSat is relatively short. However, it is a calculation you will have to be sure of!

A power bank, such as a portable mobile phone charger, is also a suitable option. They come in all shapes and sizes, and with varying battery capacity. Some also have 'smart' electronics that do not provide power if the power being used by the device is low. Whilst this could be a useful energy saving feature you will have to investigate what the electronics deem to be 'low' and if this is suitable for your CanSat.

Putting the 'Can' in CanSat



The last, but by no means least important, step to building your CanSat is to build a case to house all of the electrical components for your primary and secondary mission. Not only does this protect the components from the large forces it will experience during launch, it can also offer some protection against the environment, such as a light rain or low temperatures.

Using a 3D printer allows you to build the case to your exacting specifications. A layered design is a common approach. This can be used to separate the primary and secondary mission, for example. Using a case like this also allows you to add the function of an easily removable case, making on-the-fly repairs and changes to your CanSat much easier. Here you can download the .stl files for 3D printing the case parts that you see in the picture.

This guide has presented an introduction to electronics and sensors for the Primary Mission of your CanSat project. You can now combine your understanding from our supporting resources to build your primary mission.

Exercise 2

Once you have an understanding of the different types of components available for your primary mission it might be a good idea to plan the production, using something like the table below:

Table A2

Element	Component chosen	Reasons for choice	Next best alternative
Microprocessor			
Temperature sensor			
Pressure sensor			
Transceiver			
Power			

Good luck!

→ Links

Information on how a Thermistor works:
https://en.wikipedia.org/wiki/Thermistor

Information on the principles of a pressure sensor:
https://en.wikipedia.org/wiki/Pressure_sensor

Information on the Piezoresistive effect:
https://en.wikipedia.org/wiki/Piezoresistive_effect

An introduction to the theory of, and building of a voltage divider circuit:
https://learn.sparkfun.com/tutorials/voltage-dividers

Information on the digital pins found on an Arduino:
https://www.arduino.cc/en/Tutorial/DigitalPins

A guide to soldering:
https://learn.sparkfun.com/tutorials/how-to-solder-through-hole-soldering

Data-sheet for the MPX4115A pressure sensor:
http://www.farnell.com/datasheets/8723.pdf

Adafruit and Sparkfun are two websites that provide sensors and components suitable for the CanSat Primary Mission:

https://www.adafruit.com/categories

https://www.sparkfun.com/

The .stl files for 3D printing the Cansat case part:
http://esamultimedia.esa.int/docs/edu/3d_printer_files_for_Cansat_case.zip