Chapter 56 - Simulation

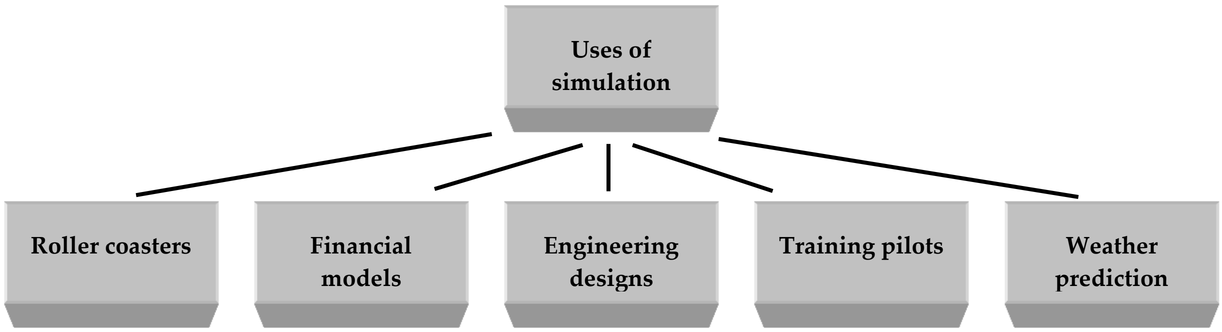
56.1 Examples of simulation and reasons for using them

According to the British Computing Society's "A Glossary of Computing Terms", simulation is "the use of a computer program to predict the likely behaviour of a real-life system". A system is described using mathematical rules and then tested and the rules improved until they describe the system well. You then program these rules into a computer program. Instead of running the actual system, you can now run the program and get the 'same' result. You can also look at different situations and see what the outcome would be.

56.2 The difference between a model and a simulation

When you have a problem, you can construct a model. A model is a set of rules that define how data changes. When you use the model by feeding values into it and watching what happens, then you have a simulation. For example, you could have a model that is made up of the following rule: "The fox population doubles when the rabbit population trebles". This is a model because it is made up of a rule that describes how data changes. The simulation is when you actually enter different numbers of rabbits and watch what happens to the fox population.

56.3 Some uses of simulation and reasons for their use



56.3.1 Roller coasters

Roller coasters are designed using complex simulations to make them ever faster, with greater G-Force sections and improved thrill factors! It is not practical to build each and every idea that a designer has. It would cost too much and take up too much time. Engineering rules, effects and dynamics are well known. These can be used to provide a working model of a known roller coaster. A simulation can then be run for a given set of data and the roller coaster observed. Different layouts can quickly be tried out. If there is a crash, no one gets hurt!

56.3.2 Financial models

A business is very complex. It has money coming into the business from a variety of sources and money leaving the business from a variety of sources. If the managers are looking to increase profits, they could set up a model in a spreadsheet. The model could have lots of variables. Variables are items that vary! In this situation, the business model variables might be the gas/electric bill, the salaries, the Christmas bonus, the money for materials to make products, the sales in a month and the cost of running the company cars, for example. They are all variables because they are values in the business that might change. The model would also contain rules about how those variables affect each other and how the total money coming into the business and leaving the business is calculated. A simulation could then be run by actually putting some numbers into the model and seeing what happens to the business. Different business situations could be tried out without the danger of making some serious financial blunders! Spreadsheets are often used for these kinds of models. Spreadsheet software is widely available, you can use 'what-if' calculations and goal-setting functions easily, graphs can be used to show the results and these can update themselves automatically as data changes, calculations can be done automatically and they are also much easier and quicker to carry out using the software than doing them by hand.

56.3.3 Engineering designs

If you wanted to build a new bridge, you could design it on paper, build a prototype and then build the bridge itself. The problem with building prototypes is that they can be both expensive and time-consuming. It would be much better if you could create a model on a computer that includes variables such as likely traffic, wind speeds and temperature changes, for example, and how those variables relate to each other. Then you could run a simulation. You could try out lots of different designs, exploring new ideas. Your model, however, may not be perfect! It may not represent every single variable that changes the way a bridge behaves in real life. Models should constantly be improved, therefore. For example, in this bridge example, the model would be used to design a bridge. The bridge would be built. The way the actual bridge behaves would be compared to the model. Any differences would be noted and the rules of the model 'tweaked' so it takes into account the new data from the new bridge. The next time a bridge is being designed, the model will (hopefully) be a little more accurate. All that said, it is still possible to get things badly wrong! Witness the Millennium Bridge that connects St Paul's to the Tate Modern in London. It is very difficult to produce really good models sometimes. Even if your model is good, it can never take into account totally random events or events that haven't been thought of and designed into the model!

56.3.4 Training pilots

Training pilots is expensive, time-consuming and uses valuable resources. Flight simulators have been built that allow pilots to do much of their training in safety. The simulators can be used to provide a whole array of different situations, something that would be impractical in a real plane. Different weather conditions or different emergency situations could be set up for pilots to try out. Their reactions can be saved, replayed and discussed after the simulation has been done with the aim of helping them to improve their performance. Pilots are not endangering their own lives or their trainers' by making mistakes in real planes. They can build up their confidence. They are not taking up planes that could be used for other things that make money for the company. Flight simulators are **very expensive** to set up. Compared to training in planes, the benefits outweigh the expense. Simulations, however, are just that, simulations. There is no substitute for hands-on experience. You cannot factor into a simulation the adrenaline that might be pumping around a pilot's body in a real emergency situation! Pilots still need to get real plane experience. Would you like to go in a car driven by someone who has only played car racing games and never had any practical experience?

56.3.5 Predicting the weather

A huge amount of data has been collected about the weather. The relationships between a set of environmental circumstances, the temperature, wind speed, phase of the moon and so on and subsequent weather conditions have been noted for many years. The variables are now part of a weather forecasting program that can be used to predict weather conditions. This is very important to flying planes, preventing disasters from hurricanes and floods, the military, to name but a few. Predicting the weather requires a huge amount of processing because it involves working with large amounts of data. Because of this, computers that use parallel processing are used.

56.4 Reasons for using simulation

In the previous section, examples of models and simulations were given and the reasons for using them. You should also have done some background research and found your own examples. Broadly, though, the following points can be made:

1) Simulations allow you to explore different approaches.
2) Simulations can be expensive to set up but over a period of time can save you money. (Think of a flight simulator).
3) Simulations may allow you to replay runs through the simulation so that they can be discussed. This might be important in a flight simulator or for the army. They use simulators to simulate how a battle would progress for a given set of circumstances.
4) Simulators allow you to try out roller coaster designs, for example, without endangering life.
5) Simulators might be the only realistic way of testing theories without potentially causing havoc! For example, the Chancellor of the Exchequer may have a model of the economy to help him decide what variables to change each budget. He may explore lots of different scenarios. He couldn't explore them in practice!
6) There may be situations where it is too dangerous to carry out some actions, for example, design changes to a nuclear plant or chemical experiments. Simulations provide a safe alternative approach.

56.5 Models must be constantly refined

Simulations are only as good as a model's rules. A model needs constantly to be refined. Each time some new real-life data becomes available, it should be compared to the results the model would have given. If any areas are identified where the real-life situation differs from the model's predictions, then the rules of the model need to be examined and 'tweaked'. This is a never-ending process.

56.6 Simulations are never perfect

Models are based on rules. When you make a rule, you usually have to make assumptions. For example, suppose you had a model that predicted the number of foxes and rabbits in a wooded area. The variables include the number of rabbits in the area at the start of the simulation, the breeding rate of the rabbits, the number of foxes at the start of the simulation, the breeding rate of the foxes and how many rabbits a fox eats in a day.

The rules that you define in your model are based on assumptions. For example, you might assume that the breeding rate of both foxes and rabbits remain constant from month to month, regardless of the time of year. Clearly this isn't true. You might assume that the eating rate of foxes remains constant. You might assume that there isn't anything else killing rabbits (or foxes) such as disease, other predators such as man, the weather or old age. You might assume that there aren't any random events, e.g. fire or flood. There are many reasons for using a simulation. The models that simulations are based on, however, are rarely perfect. Models can always be improved by looking at the assumptions and then trying to incorporate them into the model rather than glossing over them and by comparing new real-life data when it becomes available with the way the model predicts things would happen. The rules are then adjusted so the model is a little more 'accurate'.

56.7 How do you describe a simulation?

When describing a simulation, you need to do a number of things. You need to describe what information is being displayed and the different methods of outputting information effectively. You need to describe the inputs that can be put into the system, the variables that can be changed. You need to describe any assumptions that are being made, why the simulation might not represent the real life situation. After looking at some of the train simulations and the lemonade game, and also some simulations you have found on your own such as a simulation of the queues at a supermarket's checkouts, what might you say in response to 1, 2 and 3 above? Here are some possible answers.

56.7.1 Examples of outputting information effectively

An animation is used to see the changing situation over time, for example, the position of trains over time. A digital clock is used to show time passing. The number of events occurring (e.g. number of customer's in a queue, number of customers who have bought lemonade and so on) is displayed as a number on the screen. Some data is shown as a chart or graph. For example, customer satisfaction is shown as a partially completed bar. Statistics throughout an animation or at the end of a cycle of animation are produced. These are in the form of ratios or percentages or integers/real numbers and displayed on the screen. Events are colour-coded to help the user. For example, critical events in the simulation are highlighted in red and some words flash to draw the attention of the user.

56.7.2 Describe the inputs (variables) that you the user can enter into the system and change

In the lemonade game, you can change the price of lemonade and the number of ice cubes per glass, for example, by typing in a new number. You can toggle the rate that time passes by clicking on a button. You can adjust the number of people per hour arriving at a queue by adjusting a slider on the screen. A number next to the slider shows you what has been selected.
You can click on a button to input a 'random' event into the simulation.

56.7.3 Describe assumptions and why simulations might not represent real-life situations

You're assuming that the train timetables in the train simulations don't change over the period of time. You're assuming that the weather in the Lemonade Game stays constant all day. You're assuming in a supermarket queue that a constant rate of customers arrive and leave a counter. You're assuming that there are no random events, such as a gale blowing down the lemonade stand! You're assuming that trains don't break down and block tracks.

56.8 Simulation and the role of parallel architecture

When you have huge amounts of data to process in simulations and the variables in a simulation are related in complex ways, then a large amount of computer processing will need to take place. A standard Von Neumann architecture computer could provide this. But if the output was required quickly then a different set-up might be needed. Parallel processing can reduce considerably the time that lots of calculations are done in. Consider having to add 1000 numbers using a Von Neumann architecture computer. If each addition took one cycle, then you would need 1000 clock cycles to produce a result. If you have 500 processors in a parallel processing computer, then you could add 500 pairs of numbers in one clock cycle to produce 500 results. One clock cycle later and you can add those 500 results in pairs to produce 250 results. After only 10 cycles, you will have added up all 1000 numbers! This is much faster than before.

Q1. Search for the 'Lemonade Game', 'Java simulations' and 'train simulations' and try some of them out.
Q2. What is a model?
Q3. What is a simulation?
Q4. Using an example, show your understanding of what an 'assumption' is.
Q5. What are the advantages and disadvantages of using a simulation to train a pilot?
Q6. Why are simulations useful in bridge design?
Q7. How can models be improved?
Q8. Describe different methods used in simulations to display output data, with examples.
Q9. Describe what is meant by a variable in a simulation, with examples.
Q10. Explain the role of parallel processing in simulations.

Chapter 57 - More about real-time applications

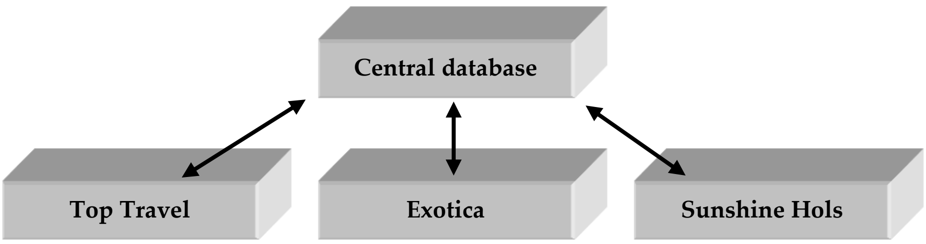
57.1 Introduction
According to the British Computing Society's 'A Glossary of Computing Terms', a real-time system is "one which can react fast enough to influence behaviour in the outside world". A key part of the above definition is the phrase 'fast enough'. Many real-time systems have outputs that change in fractions of a second as a result of changes to input information combined with the processing of that information. Examples of this include missile guidance systems and plane control systems. Some real-time systems, however, do not happen in fractions of a second. A plane ticket booking systems is an example of one such system.

Another key part of the definition is the phrase 'influence behaviour'. It is important in real-time systems that the processing of data from the input sensors of a system environment actually influences the outputs before the environment in which the system operates has changed again of its own accord! For example, suppose you have a greenhouse with a ventilation window that can be opened and closed by various amounts, depending upon the temperature in the greenhouse. A temperature sensor is used to read the temperature in the greenhouse. Let's suppose it was very cold. This information is fed into a computer. The computer processes the fact that it is very cold and sends signals to the motor to close the window. It would be a very poor system if, by the time the window was actually closed, there was a sudden heat wave and it was really hot in the greenhouse! Outputs need to be changed quickly enough to affect the environment in which the system operates.

57.2 Designing any real-time system - Input - Process - Output
When considering the design of any real-time system and the selection of sensors and actuators, it is important to try to picture the whole system and ask some questions:

1) What data from the outside world do I need to monitor?
2) How will I get the data from the outside world to the processor? (What sensors do I need?)
3) Do I need to change the signal type of the sensor so it matches what the computer can handle? In other words, do I need an interface for the sensors?
4) What will I actually do with the data, what decisions need to be made?
5) What do I want to control and when? (This is the program you need to write!)
6) How will I achieve control? (What actuators do I need?)
7) Do I need to change the signal from the computer's type to what the actuators can handle? In other words, do I need an interface for any of the actuators?
8) How can the system be started and stopped?

57.3 A plane booking system
Three different companies of travel agents sell plane tickets for a sightseeing flight around London. The plane is small, with only six seats available. The three companies use a central database to keep details of what seats are available.



Three companies use a central database to book seats.

If the first customer comes in to Exotica and buys two seats then that information must be sent to the central database (the input) and the database must be updated (the processing) before anyone else can be allowed to buy tickets. This is so that all the agents can see exactly how many seats are left (the output). If this didn't happen, then a family of five could easily go into Sunshine Hols and buy five tickets. This would result in the plane being over-booked. This is an example of a real-time system. The inputs into the system are the booking details, the processing is the taking of the booking information and updating the database and the output is the updated database and tickets. Whenever a booking is made, it influences the database quickly enough so that over-booking is not possible. Booking systems are examples of real-time systems.

57.4 A missile guidance system

When a missile is fired at a moving target, the sensors will be feeding data into the missile's processor (the input). The software program will be processing the data it gets from the input and making decisions about adjustments to the direction of the missile (the processing). The processor then sends signals to the fins and motors of the missile that causes the direction of the missile to change (the output). This is another example of a real-time system, one that is happening very quickly indeed compared to the real-time ticket booking system! Note also in this example the use of **feedback**. Feedback is when the sensors feed back the current situation, in this example, some co-ordinates. The actual value is compared to the desired value and any adjustments in the output are made. Feedback is very common in control systems where a 'target' or 'desired value' is needed.

57.5 Features of real-time systems

Most control systems are time-critical. They need things to happen quickly. They also need things to happen reliably. So when, for example, a pilot steers a plane to the left, the plane should move to the left, quickly and every time! Applications such as these are written in purpose-designed software languages. **ADA** is one such example. It is a language that has its roots in Pascal. It has features that make it suitable for real-time applications. For example, it can deal with **concurrent events** and can be used in **parallel processing** architectures. Parallel processing might be needed if there is a lot of data and it cannot be processed within the required time by one processor. By using parallel processors, you can split the program up and run different parts of it at the same time. You can also use parallel processing to build into the system some **backup systems** (sometimes called 'fail-safe systems'), in case the main one fails. It is possible to identify some features of real-time software as used in applications such as a plane's control systems.

- The output times are **predictable** within specified limits. This is important. When you use your computer at school and send something to print or start multi-tasking, then the speed that your computer will work at slows and slows unpredictably. In many real-time systems, it is important to be able to specify that XXX will happen within YYY milliseconds, regardless of whatever else is happening in the system, regardless of any interrupts that might occur.
- **Backup systems** will have been incorporated into both the software and hardware. If ever something fails in the system, there will be a backup system ready to take over immediately.
- The software is very fast! It will have been designed and written so that it runs at the **optimum** speed.

57.6 Transducers

According to the British Computing Society "A Glossary of Computing Terms", a transducer is "an electronic component which converts one form of energy to another". Some examples of sensors are described in the next section.

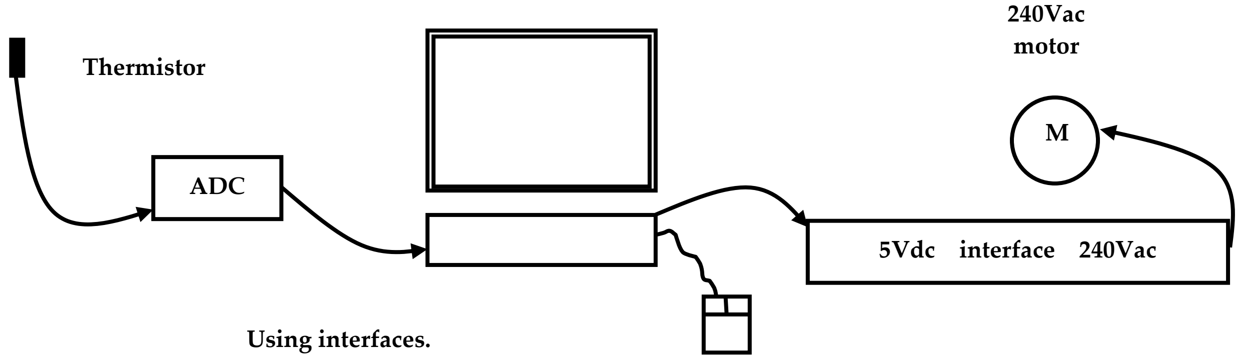
57.7Sensors

Sensors are devices that respond to a specific physical property in a system. They are transducers in that they take one form of energy and convert it into another. Here are some examples.

1) A **thermistor** is a temperature sensor (or temperature transducer). It converts a temperature into an analogue voltage. The voltage generated depends upon the temperature. There are an infinite number of temperatures and so there are an infinite number of voltages. Devices that generate an infinite range of values are known as 'analogue' devices. You could use a thermistor in a chemical plant to read the temperature of a reaction. The processor could convert the data from the transducer into a graph. Operators could then see the history of the temperature of a reaction.
2) A **photocell** is a light sensor (light transducer). It converts light energy into an analogue voltage. Light meters use photocells. A photographer can set up a camera according to the light available, as indicated on a light meter.
3) A **pressure sensor** (pressure transducer) in a chemical reaction vessel converts pressure into an analogue signal.
4) A **pressure mat** in a burglar alarm is a sensor. It senses whether someone is either on the mat or off it. It has two possible states: on and off so it is known as a digital device.
5) A **switch** or push-button is a digital sensor. It detects one of two positions: on and off. Switches can be used to send signals to a system. For example, an operator in a factory might start a system by pressing the 'start' button. This signal is fed into the computer, which processes the signal in its program.
6) A **proximity sensor** is a digital sensor. When something gets too close to it, it generates a voltage. When there is nothing next to it, it generates no voltage. Proximity sensors, for example, can be used to signal to a lift control system that a particular floor is coming up. The program can then order the motor to slow down.

57.8 Interfaces

Computers are digital devices. They can only use digital signals. A computer might use, for example the two signals 5 volts and 0 volts. Any signals going into the computer must be digital, use the same voltage the computer uses and be able to connect to the ports used by the computer. Suppose a proximity sensor, a digital device, uses 5 Volts but the wire carrying the signal is connected to a plug that can't be connected to the port at the back of the computer! Suppose a pressure mat uses 12 volts. How can this be connected up to a computer that can only take 5 Volts? How can you connect an analogue temperature sensor, with lots of possible voltages being generated, to a computer that can only accept two different voltages? You need an **interface**!

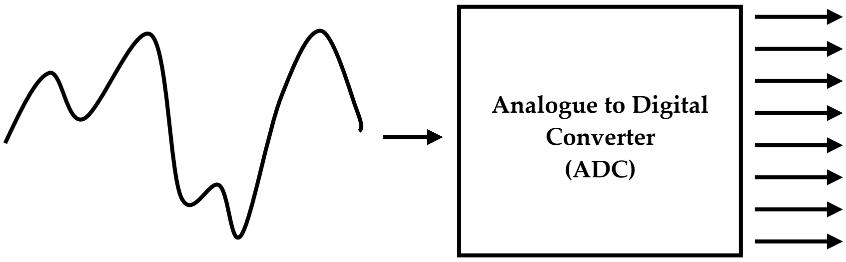


Using interfaces.

An interface converts analogue signals into digital (or vice versa). It converts signals from one voltage level to the computer's voltage level. It provides a means of ensuring that sensors and actuators can be physically connected to the computer.

57.9 Analogue to Digital Converters (ADC)

One common requirement is for analogue signals to be converted into digital signals. This is done using an Analogue to Digital Converter, or ADC. It takes the analogue signal and converts it into digital information, which can be read by the computer.



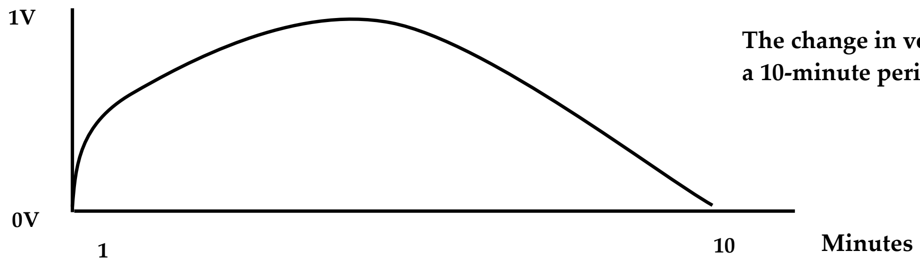
An ADC

57.9.1 An example of the need for an interface with an ADC

Suppose we have a chemical reaction experiment in class and we want to measure the temperature during the reaction. We want to display the temperature at minute intervals on the computer. We will use a thermistor to read the temperature. A thermistor is a type of transducer. The voltage signal that comes out of the thermistor constantly changes as the temperature changes. There are an **infinite number of temperatures** possible so there are an infinite number of voltages possible. For this reason, we say that the thermistor is an **analogue** device. We will assume that the voltage varies anything from 0 volts to 1 volts for a range of temperatures from 0 deg C to 150 deg C. We cannot just plug the analogue device into the computer. Firstly, the computer is a digital device and only understands digital signals. Secondly, the voltages sent out of the thermistor may not be of the same level as the voltages used by the computer. Thirdly, it may physically not be possible to connect the wires from the thermistor into an I/O port at the back of the computer. For these reasons we need an interface. One part of the interface will be the ADC or Analogue to Digital Converter.

57.9.2 How does an Analogue to Digital Converter (ADC) help?

We need to be able to convert any analogue voltage into a digital one. There will also be times when we need a Digital to Analogue Converter (DAC), to take a digital signal and generate an analogue voltage. If we constantly recorded the analogue voltage over a 10-minute period, we may end up with a graph that looks like this:

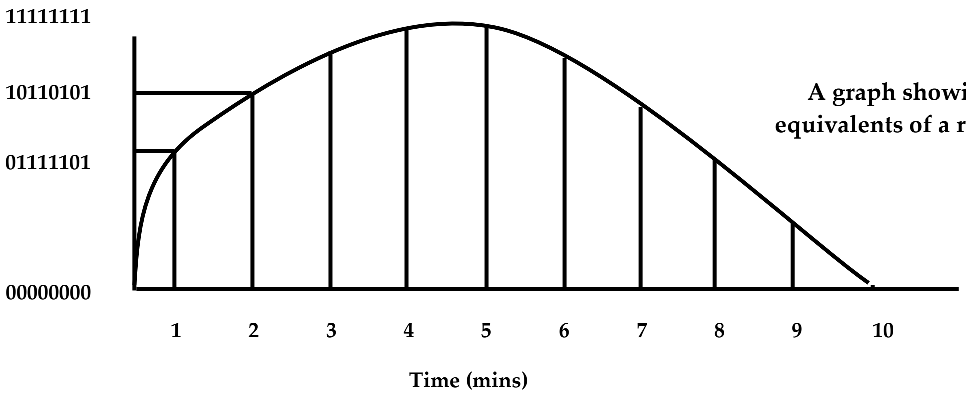


The change in voltage over a 10-minute period.

At the start of the experiment, the voltage from the thermistor is about 0.1 volts because the temperature is about 15 deg C. It rises steeply in the first minute and then steadily up to about 0.95 volts after 5 minutes, corresponding to about 140 deg C. It then declines steadily over the next 5 minutes. An ADC will read the voltage at set time intervals. To put it another way, it will **sample** the voltage regularly. How many samples we decide to take in any period of time is known as the **sample rate** and the only way to decide what is the ***best*** sample rate is to examine each problem on its own. In some applications, you may need to take a sample only once an hour, for example, measuring the temperature on top of a mountain. There may be other circumstances, such as in the above experiment where taking a sample once every hour would mean that important events are missed! A better sample time might be once a minute or once every 30 seconds. Of course, we could take 1000 readings per second if we wanted to but this would not give us any more information than taking a reading once a minute. All we would be doing is wasting computer storage. Choosing the right sample rate is a mixture of common sense and experience! Let us decide on a sample rate of one reading per minute. Each time a sample is taken, the voltage is converted into a digital value. The digital value will depend upon the type of ADC we have. For example, suppose we had an 8-bit ADC. That means that we have 256 different bit patterns (0 - 255).

- If the ADC reads 0 volts this will be represented as 00000000
- If the ADC reads 1/255 volt this will be represented as 00000001
- If the ADC reads 2/255 volts this will be represented as 00000010
- If the ADC reads 3/255 volts this will be represented as 00000011, and so on.

We can show this diagrammatically. The first two conversions have been shown on the graph.



A graph showing the digital equivalents of a range of voltages.

By using an 8-bit ADC, we can only divide up our 1 volt range into 1/256 divisions. We could get a much **greater accuracy** by using a 16-bit ADC. We could now divide up our 1 volt into about 65000 divisions and we could therefore measure even smaller changes to temperature. This is the same as saying we can measure the temperature more accurately! There is nothing stopping us using a 32-bit ADC for even better accuracy.

57.10 Actuators

According to the British Computing Society "A Glossary of Computing Terms", an actuator is "any device which can be operated by signals from a computer or control system causing physical movement". Sometimes, the processor in a system can change outputs directly (via an interface). E.g. lights could be switched on or off, an intruder alarm could be sounded or stopped, a graph could be plotted on the screen. However, when **physical movement** is required you need an **actuator**. You need a device that can cause movement when you send an electrical signal to it. E.g. if you needed to let water flow through a pipe, you might need to move a hydraulic arm that was connected to a valve. The hydraulic arm accepts electrical signals and moves. It is an example of an actuator. You might need to start and stop a lift by starting and stopping a motor. The motor is another example of a device that accepts electrical signals that causes physical movement. You might need to send a signal to a solenoid valve, so that it operates and opens a lock on a security door. The **solenoid valve** is an actuator. Just as you need interfaces to convert signals from various sensors into the signals used by the computer, you also need interfaces to convert signals from the computer into voltage types and levels that actuators (and other output devices) use. A motor might use 440 Volts ac. You would use an interface to allow 5 volts dc to switch on and off a device that uses a much bigger voltage.

Q1. Describe five sensors and two actuators.
Q2. What is the purpose of an interface?
Q3. How does an analogue signal differ from a digital signal?
Q4. Why is the sample rate important?
Q5. Use the Internet for research. What is a solenoid valve? Describe two applications for a solenoid valve.

Chapter 58 - Robots

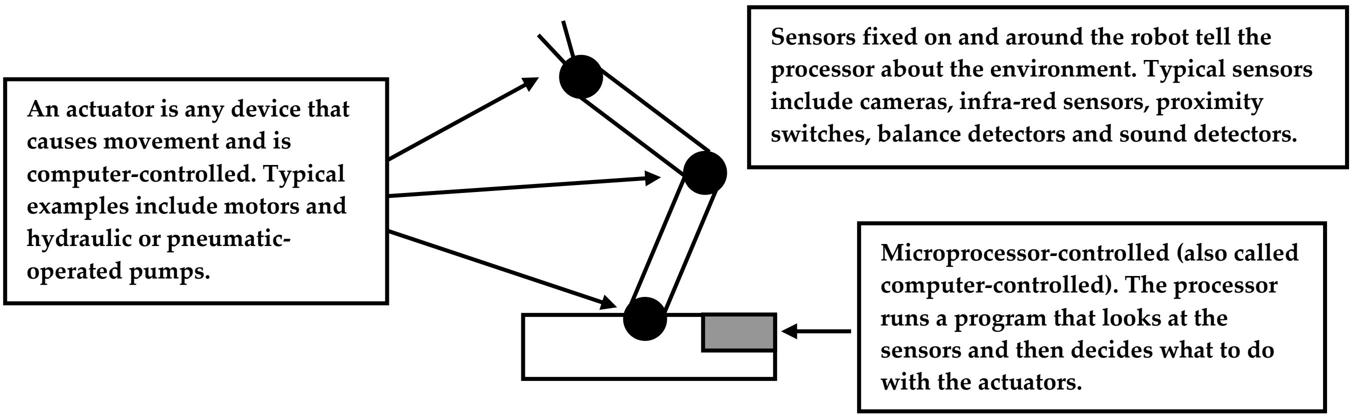
58.1 Introduction

According to the British Computing Society's "A Glossary of Computing Terms", a robot is "a computer-controlled mechanical device which is sufficiently flexible to be able to do a variety of tasks". Robots are used extensively, including for welding, painting cars, assembling parts of machines, shearing sheep, packing and unpacking boxes, in warehouses to collect items, space exploration, deep sea exploration, picking fruit, bomb disposal, surgery, maintenance in nuclear reactors, assembling complete houses, building circuit boards, exploring volcanoes and so on!

58.2 What is a robot?

The word 'robot' comes from the Czech word 'robotnik' meaning 'worker' and was coined for English use in the early 20th century. Robots are essentially computer controlled mechanical devices that are based on the human arm. The arms often have 'joints', a little bit like elbow joints, to allow greater flexibility. These are positioned using computer-controlled motors (known as 'actuators'), the computer calculating the correct position for the arm. Sometimes, the joints have a sensor. The sensor 'feeds back' to the computer what angle a joint is actually at and the computer compares that with what angle it wants the joint to get to. It then uses the motor to adjust the joint position and the process of 'feedback' is repeated. Robots usually have a specialised 'hand' or gripper on the end of the arms that can be designed for carrying out different jobs. The grippers can also be controlled by actuators but pneumatics and hydraulics can be used as well.

They are usually designed to perform a task over and over again with great accuracy and speed and they have the ability to react with their environment; when the environment in which the robot is operating changes, the robot's behaviour changes. Some robots are static. They are fixed in one place. Others have the ability to roam around!



A typical fixed-position robot.

A typical robot will be made up of sensors, actuators and a computer. The computer's job is to read the data from the sensors, make sense of it and then operate the actuators as appropriate.

58.2.1 Degrees of freedom

If we examine the human arm for a moment, we can see the movement it provides us with is quite complex. You can rotate your shoulder up and down as well as left and right. At the elbow, you can move your lower arm up and down. The wrist can be rotated, moved up and down and also left and right. Each finger is complex, being made up of more joints that provide more opportunities for movement. Each of the ways of movement at any particular joint is known as **a degree of freedom**. So the shoulder joint has 2 degrees of freedom. The elbow has one degree of freedom, the wrist three and so on. The more degrees of freedom in a system, the more complex the movement that can be achieved. The human arm has many degrees of freedom indeed and so we can manipulate objects easily. When you look at a robot, one of the key features to take note of is how many degrees of freedom it has. The more degrees of freedom it has, the more complex the robot.

58.3 Nanorobots and insect robots

Robots do not have to be very big. Nanorobots (one nanometer is one thousand millionth of a meter) are tiny robots designed to carry out very specific tasks to a very fine tolerance. They can come with their own on-board computer, just like the one we discussed in the previous section. There can also be grouped as a set of nanorobots under the control of one computer. These systems are known as **insect robots**. The use of insect robots, lots of nanorobots working together, has potential applications in medicine, where they can be used to fight disease by seeking out and destroying very specific targets such as bacteria in the body. Their use in building very small machines is also under investigation. Nanorobots are tiny and work faster than larger machines. They also need very little power to work and have the added bonus of lasting a long time! Nanotechnology is an exciting area to be working in and you may want to investigate this area as you consider options for university!

58.4 Benefits and drawbacks of robots

Expensive in the short-term but saves money in the long term

Save on low-skilled salaries

Can be used in hazardous places – saves lives

High quality work produced

Opportunities for higher skilled jobs

No sickness, tiredness or holidays needed

Human facilities not needed e.g. toilets, car parks

Change in employment patterns affects community

1) Robots are extremely expensive to buy. They also need highly skilled (and expensive) people to reprogram them to do different things.
2) Robots should produce financial benefits because you don't have to pay a salary (amongst the other reasons listed below) but these benefits are not usually felt for a number of years.
3) The quality of work produced by robots may be higher than with humans. For example, spot welding can be done to much finer tolerances. Perhaps more importantly, however, the work done by robots will be done to consistent standards. With humans, mistakes can vary with how tired the employee is, the day of the week. Cars that keep breaking down are sometimes referred to as 'Friday afternoon' or 'Monday morning' cars. Mistakes may also creep in if an employee is having personal problems, for example.
4) Employees get tired easily, go sick, need breaks and have holidays. Robots can work for much of the time. They do need to stop sometimes, however, for such things as routine maintenance and system upgrades.
5) Employees need facilities that don't need to be provided for robots. These include lighting and heating, car parks and toilets, a personnel department to look after them and an accounts department to make sure they're paid!
6) Robots can be used in hazardous environments such as for bomb disposal, maintenance in nuclear reactors, space or in very deep parts of the ocean.
7) Robots provide opportunities for a higher paid, more skilled workforce, for example, programmers of robots or maintenance people. However, the workforce may be smaller. This means there will be fewer jobs available in a car factory and less unskilled or semi-skilled jobs in a community. This will have an effect on the local economy.

Q1. Define 'robot'.
Q2. State three uses for robots. Use the Internet for research.
Q3. State the advantages to a company of using robots to assemble cars.
Q4. State the impact on employment that introducing robots onto the assembly line in a car factory might have.
Q5. What is an actuator?
Q6. A robot arm has to position a box accurately in a warehouse. What is meant by 'feedback'?
Q7. What does a proximity switch sense?
Q8. What is meant by a degree of freedom in robots?
Q9. Use the Internet for research. Describe a medical use for nanorobots.
Q10. Use the Internet for research. Why are robots used for bomb disposal?

Chapter 59 - Embedded systems

59.1 An overview

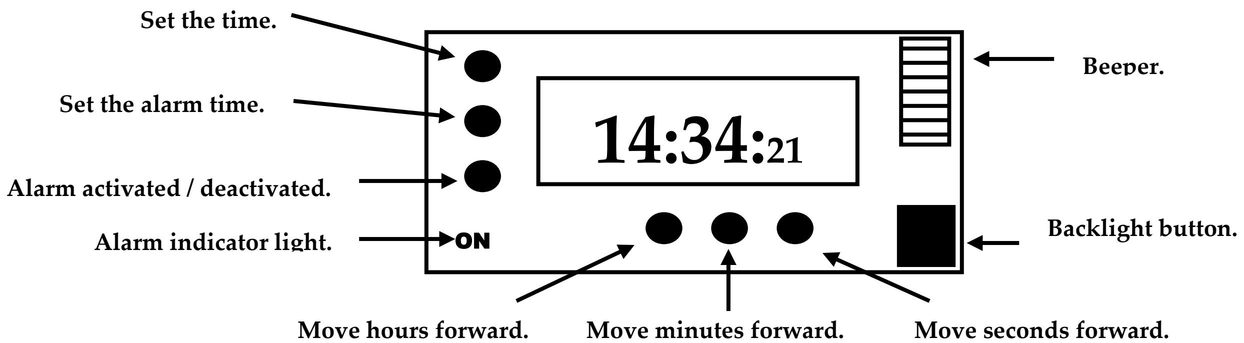
A typical home computer is a general-purpose system that can easily be adapted to many different applications. We all like to have the best home computer that we can afford so we can do as much as possible with our computers. We want the fastest processor, as much RAM and cache as we can get, the best storage device we can afford such as a DVD writer and the latest peripherals. Embedded systems, on the other hand, are computer systems designed for a very specific purpose. They are often mass-produced for goods such as washing machines, telephones and videos and it is often not at all important to have the fastest processor, the biggest memory and so on. Keeping costs low is a far more important concern and so slower processors and simplified designs are common. Also of concern is the amount of power consumed by the embedded system. It is important for a designer to keep the power consumption of a system low to keep running costs low and power needs simple.

We have already said that washing machines and videos use 'embedded systems'. They have sensors (the inputs) that read data and pass it to the processor. The processor examines the inputs and decides what outputs to set (the processing). Signals are sent to the actuators (for example, motors and pumps) to actually make things happen (the outputs). We have said that embedded systems often have simplified designs. They often do not have an operating system, a disk drive, a VDU or keyboard. All they have is a processor with a program that reads in input data, processes it and then outputs signals to make things happen. The program is in the machine code of the processor and stored in ROM. The program is relatively simple, requiring only very specific tasks to be done. It will often be written within a loop, so that, once set, the program simply repeats itself until it ends or is aborted.

Whilst home computers generally use either the Intel 80x86 family of processors (for IBM compatible computers) or the 680x0 family of processors (for the Apple Macintosh computers), embedded systems use a wide range of processors. A further development is that the processor is often designed into an integrated circuit along with all of the other circuits needed so that there is just one chip. This helps simplify the design, aids mass production and helps to keep production costs low.

59.2 A digital alarm clock - a case study

Consider a modern digital alarm clock. It is an example of where an embedded system is commonly used. The alarm clock is mass-produced. To keep costs low, an integrated circuit (I/C) is designed for it. The I/C will have the processor on it along with any other electronic circuitry needed by the processor, for example, some ROM. The I/C will be connected to the inputs such as the buttons to set the time, and the outputs, such as the time display, beeper and 'alarm on' indicator. The ROM chip will be used to hold the program.



A typical digital alarm clock.

- **Function 1**. When the 'Set the time' button is held down, the time can be set by pressing the 'Move' buttons under the display. If a Move button is pressed once then the corresponding part of the time display moves forward by one number. If a Move button is held down, then the corresponding part of the time display moves through the numbers quickly.
- **Function 2**. When the 'Set the alarm time' button is held down, the alarm time can be set by pressing the 'Move' buttons under the display. If a Move button is pressed once then the corresponding part of the time display moves forward by one number. If a Move button is held down, then the corresponding part of the time display moves through the numbers quickly. If the 'Alarm activated / deactivated' button is pressed, the alarm is toggled between set and reset. The alarm indicator is lit as appropriate.
- **Function 3**. The current time is displayed.

- **Function 4**. If the alarm is set then the program needs to monitor the current time and the alarm time. At the appropriate moment, the beeper is made to beep. It continues to beep until the alarm is deactivated or 10 minutes have passed.
- **Function 5**. The backlight is turned on whilst the backlight button is pressed.

59.2.1 How would we design the program?

There are different ways we could write a program to control the alarm clock. If we were to use a procedural approach, we would note that there are five different procedures. We could write a program as a set of five procedures. The main program would simply be a loop, calling each procedure in turn. We have seen this idea before. Another slightly different approach might be to design a program where some procedures are called only if an interrupt happens in the main program. For example, instead of calling the procedure to turn on the light behind the display because it is its turn to be called in the main looping program, it *only* gets called if the backlight button is pressed. When the button is pressed, it sends an interrupt signal to the processor that then reacts by calling the backlight procedure. This has the advantage of not wasting time calling procedures even if they are not needed! They only get called because there has been a signal to say that a function is needed.

We could program the alarm clock using an object oriented approach. We could start by identifying the classes needed. For example, one obvious class is the display. We would identify the data needed for the display and the methods required to set and display the time or alarm time. Another class might be for the beeper. We would need to identify what data would be needed for the beeper and what methods it would need. However the program was written, whatever language was used, it would be translated into object code and then burnt into the ROM chip.

59.3 Critical embedded systems

Some embedded systems are **critical**. By that, we mean that they are in places where they do a very important job and if they fail, it may have catastrophic implications. Embedded systems in planes, for example, are often critical. These types of embedded systems need a different design approach to the embedded systems found in washing machines, for example. They need to run predictably. When we looked at the alarm clock, one way to design the program was using interrupts. If the backlight button was pressed, then an interrupt occurred that resulted in a procedure being called. If that procedure was called within 3 milliseconds one time and within 20 milliseconds the next time it wouldn't really matter. With critical systems, it matters a lot. It is very important to be able to predict with accuracy the reaction times to interrupts and the time that some code will run. They need to run reliably. It is important that a mission-critical system will work whatever happens! To achieve this, critical embedded systems have many fail-safe systems built in. They may have a second or even third processor, for example, ready to step in and take over if the main processor fails. They may have methods that allow data to be read in twice using different sensors and checked against each other. They may have sub-systems ready to take action monitoring the main systems. If necessary, a sub-system can, for example, alert a pilot to a problem. They may have software that can attempt to recover from faults. Programming languages such as ADA are designed for this. Critical systems are expensive. They have a lot of extra hardware and software that may never be needed. Although any computerised system can never be 100% reliable, critical embedded systems attempt to reduce the risk to acceptable levels.

59.4 The use of embedded systems

Embedded systems are so widespread and taken for granted that you may not realise how widespread they actually are! Estimates differ on the number of embedded systems in a new car these days but estimates somewhere between 30 and 100 seem popular. The average number of embedded systems in a 'typical' home may be between 200 and 300. How many are there in your home? You have many devices that have a processor controlling them including alarm clocks, digital wristwatches, toasters, microwaves, washing machines, cookers, central heating systems, Hi-fi equipment, guitars and amplifiers, scanners, digital cameras, TVs, video players, tamagotchi, burglar alarms and the list goes on. There is a shortage of skilled embedded systems designers. It is a very satisfying career because you can be involved in lots of different aspects of the whole design. You may want to look up computer-based courses at university that include at least an element of embedded systems design!

Q1. What is meant by an embedded system?
Q2. What is RAM used for in a laptop?
Q3. What is ROM used for in a laptop?
Q4. How does an embedded system differ from a laptop?
Q5. State five items in your home that has an embedded system.
Q6. What is meant by a 'fail-safe' system? Describe an example.
Q7. How might fail-safe systems differ from an embedded system to control the TV at home?
Q8. Design the user interface for a 'typical' digital alarm clock.
Q9. Who was the programming language ADA named after?
Q10. When was she born? What does she have to do with computing? Who was her father?

Chapter 60 - Nomadic networks

60.1 Introduction

The term 'nomadic' simply means 'mobile'. A nomadic network is a network that makes use of mobile access to a network. It can be applied interchangeably to mobile phone networks or to a network whose users access the resources of a network using mobile technology. It can be used to describe mobile access in a building, country, continent or the world. People need to keep in touch while on the move for a number of reasons. Travelling salesmen may need access to their company's central database from a laptop and mobile phone or may need to place an order into an automated ordering system. Parents may want to know where their children are. People want to use a network for entertainment or to socialise so they would want access to email or SMS (see later), for example. Within a building, you may need to contact building maintenance personnel and so would want to use a two-way radio system (still a nomadic network) or a paging system.

60.2 Problems associated with mobile networks

There are problems associated with any mobile hardware, be it a mobile phone or a computer on a mobile link. These include: **Poor bandwidth** (limited information can be transferred to the mobile device). **Poor reception** (the signals received are subject to atmospheric conditions, the presence of mountains and valleys and the proximity of transmitters or base stations). **Poor security** (transmissions can be intercepted and captured with the right equipment). **Hardware limitations** (these include the power available from the battery, the limited display available, poor keyboard design and bandwidth limitations).

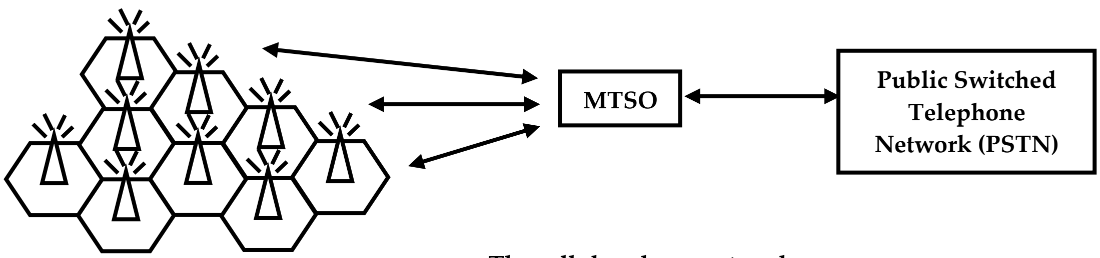
60.3 Recent trends in mobile phone use

Mobile phone companies have been selling ever-increasing numbers of phones. However, users are not so willing to use them! They are not cheap to use and while users have been buying the latest phones, their actual usage is limited. Mobile phone companies have been doing their best to encourage us all to use our phones more and more. They have, for example, been developing and marketing both voice and data communications on the latest phones, using WAP (Wireless Application Protocol) technology to promote Internet use and email. Developing SMS (Short Message Service) business is another area where cellular phone companies have been busy. This operates within the Global System for Mobile (GSM) communication network and allows users to send messages up to 160 characters long. Initially, mobile phones operated within one country. The GSM network was developed to enable communication to take place all over Europe and into countries beyond Europe. This means that users can easily communicate across national boundaries, but at a cost!

60.4 The demands placed on nomadic networks

Both business and home users want faultless, reliable connections so that their normal patterns and methods of communication and business remain the same. They need equipment that is not only portable but which can be connected up to a range of peripherals easily. **PCMCIA cards** enable laptops to be connected to mobile phones. They then can gain access to the Internet.

60.5 How does a mobile phone network work?



The cellular phone network.

A mobile phone is simply a radio that can both send and receive voice sounds in the form of radio waves. It can do this at the same time because it uses one radio frequency to send voice signals and a different one to receive them. This is known as 'duplex communication'. Consider a country such as the UK and a mobile phone company called Theteacher Telecom. Theteacher Telecom will split up the country into areas called 'cells'. In the middle of each cell will be a 'base station'. The base station is simply a radio transmitter/receiver. Base stations will typically be positioned 500 metres apart in cities but may be 10 km apart in the countryside.

When you switch on your mobile phone, your phone listens out for a special signal that is constantly being broadcast by the base stations. This signal is known as the **System Identification Code**, or SID. When your phone picks up a SID signal, it then knows that it is within range of a base station in a particular cell. If it cannot receive this signal then it displays a message on your phone that it can't find your network. This means you can't make a phone call! Assuming that your phone has received the SID, your phone then transmits a signal that identifies your phone and where you are (which cell you are in). This signal goes via a base station transmitter to the area **Mobile Telephone Switching Office**, or MTSO. The MTSO keeps track in a database of all the phones that are in that MTSO area at that time and which cell they are in.

When you want to make a phone call, the MTSO will allocate your phone a unique pair of frequencies to use from the hundreds it has available for the duration of the call. When you make a call, you will send your voice on one of these frequencies via the base station's transmitter in the cell you are in to the MTSO. The MTSO will then relay your call to the destination using the normal **Public Switched Telephone Network**, the PSTN. The PSTN is the normal land-based telephone system that we use. When you listen to someone talking, the process is reversed using the other frequency you were allocated.

Whilst you are making a call, the strength of your radio signal is being monitored. The transmitter in the cell you are in as well as the transmitters of the adjacent cells do this. If you move towards the edge of a cell, your radio signal will reduce. As you cross into a new cell, your phone call will come under the control of a new transmitter and leave the control of the transmitter of the cell you were just in. This happens reasonably seamlessly as this change is coordinated by the MTSO. You may have noticed, though, that when you are in a train, for example, the signal does diminish sometimes. This is the effect of you moving to the extremes of a cell's boundary and the limit of a particular transmitter.

60.6 Are mobile phone networks safe?

It has been known for a long time that radio waves can cause a heating effect on the body and can affect the efficiency of the nervous system. This had led to mounting speculation on the effects of mobile phones. Whilst people working with powerful radio waves have complained of problems in the past, mobile phones use very low power radio waves. To investigate the problems associated with mobile phones, the Government set up an independent enquiry called the Stuart Enquiry. The scientists reported in May 2000 and found: that there was little evidence that mobile phones caused health risks but that this may be due to gaps in their knowledge and the fact that the long-term effects of low power radio waves are not documented; they acknowledged that radio waves do affect cells in the body but this may not necessarily result in a health problem; children's bodies are developing and they may therefore be more susceptible to health risks than adults; mobile phone companies should publish the SAR value of each phone. This is a measure of how much radio energy a type of phone transmits to the body; there was no evidence as yet of any damage to people living close to a base station; drivers are at risk of accidents if they use their phones whilst driving and should be encouraged to use hands-free sets. Various reports have recently indicated that pupils in schools may be at risk from both WIFI signals and from the radio signals from their phones. Reports can be found e.g. on the **http://wifiinschools.org.uk/** website.

60.7 Nomadic networks and the walkie-talkie

Walkie-talkies are transmitters/receivers that can send and receive voice signals in the form of radio waves. A handset is typically a little bit bigger and heavier than a mobile phone and you can usually see a short aerial sticking out of it. Unlike cellular phones, you cannot talk and listen at the same time. You can only do one or the other and for this reason walkie-talkies are an example of half-duplex communication. You've probably seen people using walkie-talkies on TV - they keep saying "Over" when they have finished talking, to signal to the other person that it is their turn to speak.

60.7.1 Radio frequencies

There are a fixed number of radio frequencies that are available in any country. Some frequencies are reserved for TV and radio and some are reserved for mobile phone use. For a mobile phone company to operate, they must buy the right to use certain frequencies. They can then have exclusive access to them and no other company can use them. In this way, one company's phone calls will not interfere with another company's. There are some frequencies, however, that are public. In Europe, there are 8 public frequencies that can be used by anyone. You do not require a licence to use them. Within each of the eight frequencies, there are 38 'bands' that can be used, making a total of 8 x 34 or 304 different communication channels. Manufacturers of walkie-talkies make use of these public communications channels in their products.

60.7.2 Uses of walkie talkies

Walkie-talkies can form an ideal communications network in certain circumstances. They have a range of around two miles and are ideal for security personal to keep in touch in shopping centres, clubs, festivals or on a factory site, for example. They could be used to keep key people who could be working anywhere in a building in touch with reception. A Network Manager looking after a computer network, for example, or a maintenance person, could be anywhere in a building but may be needed at short notice. They would be ideal candidates for a walkie-talkie.

60.7.3 Pros and cons of walkie-talkies

Walkie-talkie handsets are cheap to buy and you don't need to then rent a phone line or airtime from any company. Handsets use rechargeable batteries so are cheap to run and they are small and highly portable. We have already said that they don't need to be licensed and a company can have as many handsets as they like using whatever public frequencies they like. Because the handsets use public frequencies, however, anyone can tune in to your conversations so they are not suitable for sending secure messages. Their limited range may be a problem for some applications.

60.8 Using satellites for phone calls

Satellite phones are used widely for communication in the media, navigation, video conferencing, data transfer and for military purposes, for example. A transmitter on the ground sends signals to a satellite using microwaves or some other form of transmission. Using a different frequency to avoid interference, the signals are then redirected back to the planet. If it doesn't reach the intended recipient, it is then bounced to other satellites and on to its destination.

60.9 Pros and cons of using satellites for phone calls

If you make a phone call using satellite technology, the cost of the call is not dependent on distance. A short distance phone call costs the same as a long distance one. Overall, the cost of a phone call is higher than non-satellite methods. Satellites can handle a very high number of simultaneous phone calls (they can handle a large bandwidth). Satellite phones can be used in the remotest parts of the world, from the arctic to mountainous areas to deserts although the actual equipment is very expensive. Because of the distance between the surface of the planet and the satellites, there is a short delay in any phone call communication. This results in 'staggered' conversations. Weather conditions can also affect the quality of the transmission.

Q1. What is a 'nomadic' network'?
Q2. What does SMS stand for in the context of using a mobile phone?
Q3. Why is 'bandwidth' important in mobile communications?
Q4. What is a protocol?
Q5. What does WAP stand for?
Q6. Give an example of simplex, half duplex and duplex communication.
Q7. In the context of mobile phone communications, what is a SID?
Q8. Compare using mobile phones to walkie talkies for communication.
Q9. Use the Internet for research. How do satellite phone systems work?
Q10. Use the Internet for research. Are the wireless signals sent out by mobile phone networks and Wi-Fi hotspots safe, especially for young people?