Thomas Robinson

Unit 24, Assignment 1

Controlling Systems using IT

Solihull College

January 2023

Task One P1

Control Systems

Command Systems

A command control system operates based on commands or instructions from a human. Examples of this type of system include everything from television remote controls to applications in military and aviation.

A television remote issues a command to a television which then acts based upon this input. When flying an aircraft, a pilot may flip a switch which the aeroplane's system then interprets.

Programmable Systems

A programmable control system is a type of control system that is designed to operate to a pre-determined set of instructions. They are used heavily in manufacturing, for example a manufacturing robot on an assembly line. Domestically, these types of systems can be found in appliances such as washing machines and microwaves.

Sensing Systems

Sensing systems operate based on data input from a sensor. Sensing systems are used for monitoring the environment as well as reacting based on changes in the environment. Examples of sensing systems include dryers that sense when a hand is placed underneath and cruise-control that takes a variety of sensory inputs.

Conditional Systems

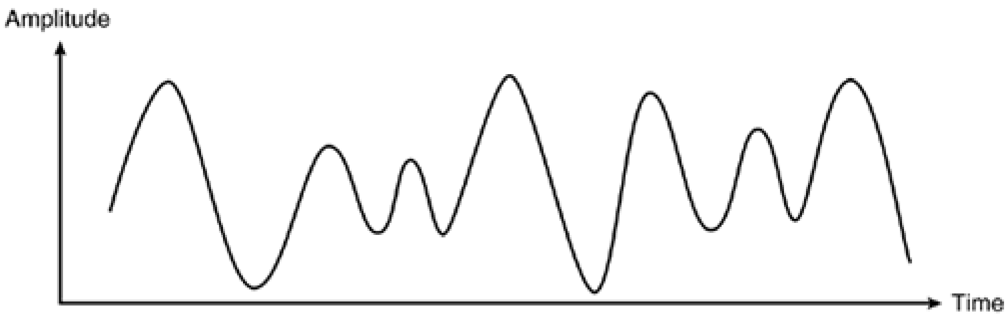
A conditional system is used in decision-making applications. The system evaluates an input or multiple input and then reacts based upon that. Oftentimes the reaction is instantaneous. Examples include a fridge that reacts based on temperature or streetlamp that reacts based on the light level.

Task Two P2, M1

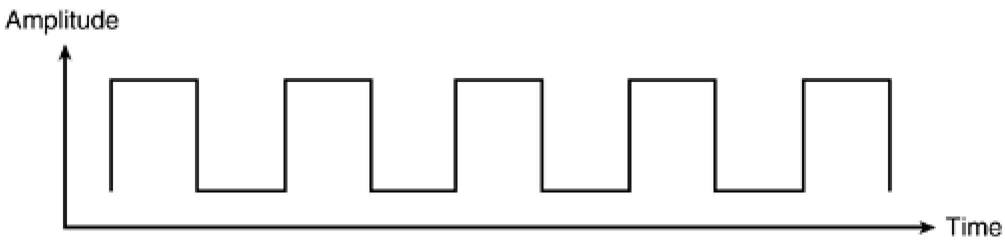
Analogue & Digital Signals

Analogue signals are continuous waves that have effectively infinite values in each range. This contrasts with digital signals, which are in discreet steps and can only take on a finite number of values in a given range.

Analogue signals are used to transmit information that changes continuously. An example would be the sound waves of sounds, music, and speech or a voltage measurement of a component. Digital signals are used for transmitting information that can be represented numerically, such as binary digital data. All information dealt with by a computer will be in a digital format.



Waveforms are how analogue signals are represented. These show the changes in the signal over time.
Image Source: computerscience.chemeketa.edu under a Creative Commons License.

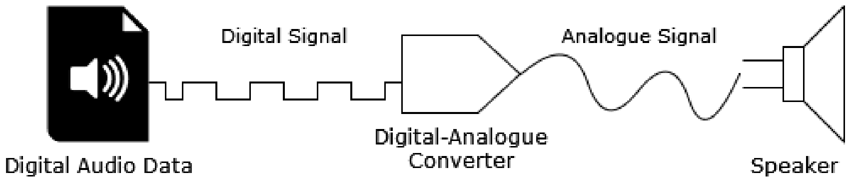


Digital signals are shown using a series of pulses, each of which represents a different value of the signal.
Image Source: computerscience.chemeketa.edu under a Creative Commons License.

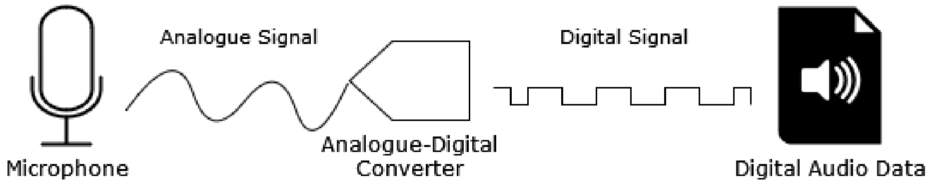
Task Three M2

Conversion

It is often necessary to convert signals from one form to another because different systems and devices are designed to work with specific types of signals. As an example, a speaker requires an analogue signal to produce audio, whereas a modern television expects a digital input to display a picture.



Using the example from above, to play audio from a speaker it must be converted into analogue. This is done using a digital-to-analogue converter (DAC). This takes a digital input signal and converts it to an analogue format that the speaker can then interpret.



In the opposite direction, a computer is unable to directly process an analogue signal. CPUs and other components within a computer deal with digital information stored, processed, and represented as binary. To use a microphone with a computer, the analogue audio signal must first be converted into a digital signal. This uses the opposite process to a speaker; an analogue-to-digital converter (ADC) quantises the analogue signals to digital values that can then be understood by the computer.

Other examples of digital-to-analogue conversion include digital potentiometers which can vary an output voltage used to control the speed of a motor or brightness of an LED. Another use for analogue-to-digital conversion is the light entering the sensor of a camera or digital multimers that measure voltage.

Image: A camera's sensor converts analogue values representing the light hitting it to digital information that can be processed and assembled into an image. *Source: @insungyoon on Unsplash*

Task Four P3

Input & Output

Input Devices

Proximity Sensor

A proximity sensor can detect the presence of an object and the distance to it. They work by sending out a signal that bounces off things in its path and measuring the time it takes to return. This signal can take the form of an optical mechanism (laser or infrared) or ultrasonic (sound).

Proximity sensors are used in smartphones to detect when they are against a users' ear, in cars as a parking aid, and in some aircraft as a way of measuring distance from the ground. Additionally, they can be used in industrial applications for measuring the speed of certain devices, such as rotational cams and cogs.

Accelerometer

An accelerometer measures acceleration, that is the change of velocity. They are used to sense orientation, vibration, shock and whether an object is falling.

In a mechanical accelerometer, a tiny mass is suspended on a cantilever. The movement of this mass generates a voltage which can then be measured. Other types of accelerometers include those that operated piezoelectrically.

An example of their use is within computer hard disks. They implement accelerometers to measure shock and freefall, allowing them to intelligently attempt to protect the internal components from damage when dropped or impacted.

Image: A packaged accelerometer on a circuit board *Raimond Spekking on Wikimedia Commons, under Creative Commons BY SA*

Gyrometers

Image: A gyroscope in operation Lucas Vieira on Wikimedia Commons, Public Domain

A gyrometer is a microelectromechanical gyroscope that is used in electronic devices. They are used to measure orientation, that is, the way in which an object is facing.

In consumer electronics, smartphone orientation sensors are gyrometers which ensure that the content on a device's display is rotated the correct way.

Measuring motion is often achieved using a combination of an accelerometer and a gyrometer. An example of this use is within a Wii remote.

Output Devices

Speaker

Speakers are output devices that receive an analogue input and convert it into sound. They are made up of three principal components: a magnet, a coil of wire, and a cone or diaphragm. The voltage applied to the coil causes the diaphragm to vibrate back and forth very quickly. This movement produces waves in the air that we perceive as sound.

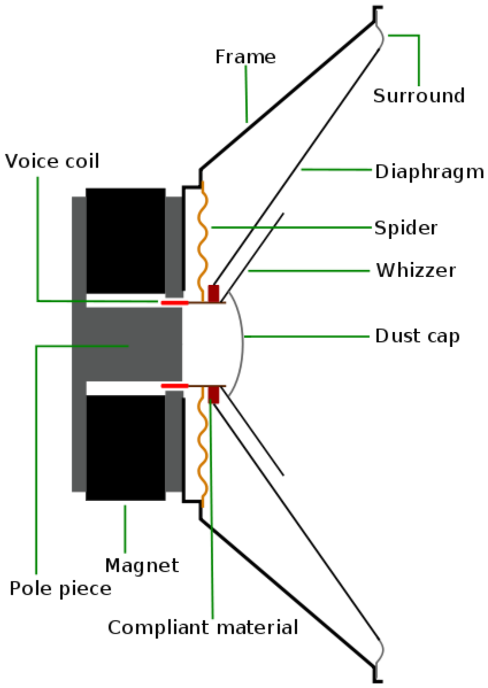
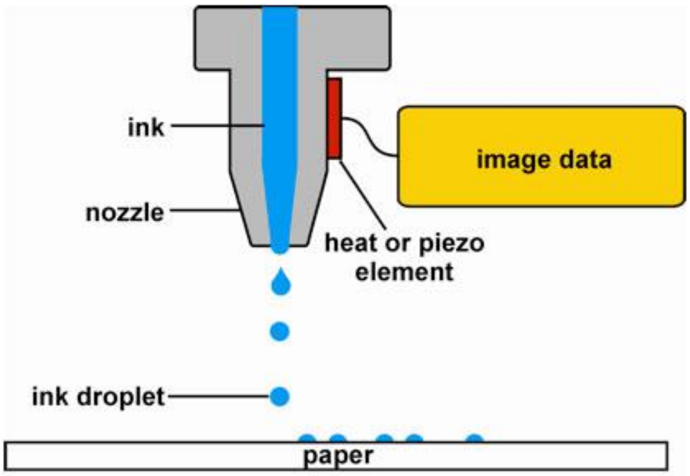


Image: A diagram of the components of a speaker *Rohitbd on Wikimedia Commons, under GNU Free Documentation Licence*

Printer

A printer is an output device that converts digital data into a physical form. Depending on the type of printer, they operate in different ways.

An inkjet printer contains a print head with small nozzles that can dispense small drops of ink onto the paper.

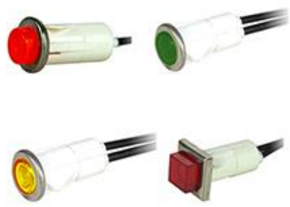


When the printer receives data, it processes it and converts it into a series of commands that control the movement of the print head and the dispensing of the ink. As paper moves past the print head, the ink is ejected at the appropriate time in order to build up an image. The dispensing of the ink is achieved either thermally or piezo-electrically.

Image: A diagram of a drop-on-demand inkjet printer nozzle *Image Permanence Institute, Rochester Institute of Technology*

Incandescent Lighting

Even the simplest devices can be an example of an output method. A simple incandescent bulb could be used to show the state of a device or flash a coded message. This could be achieved through dimming the bulb and through systems such as morse code, respectively.



An incandescent bulb works by sending electricity through a wire filament, which then glows. Most of the energy is lost to heat, making incandescence an inefficient lighting technology. To protect the filament from rusting due to oxidation, within the glass is a vacuum or inert gas, rather than air.

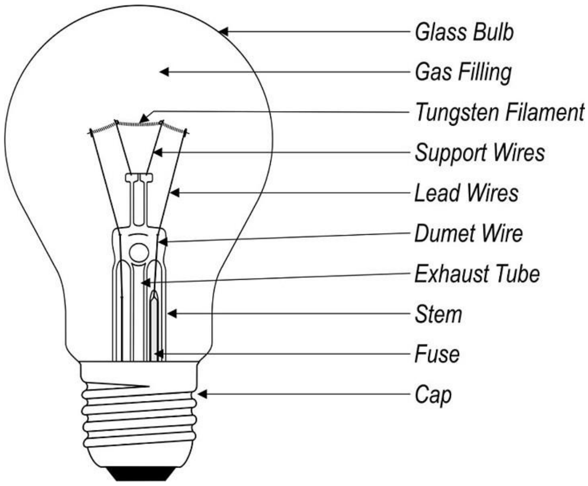


Image (top): Examples of incandescent indicator lights *Visual Communications Company, LLC*

Image (bottom): A diagram showing a more traditional bulb shape, and the components it is made up of *LampTech.co.uk*

Task Five P4

Representing Data

Binary

Data stored in binary is represented by a series of 0s and 1s. Binary uses a base-2 counting system meaning there are only two values. These 0s and 1s can also convey other meanings, such as true/false or on/off. Each binary digit is called a 'bit' and a group of eight is a 'byte.'

An example of an eight-bit binary number: *00101010*. To convert this to a decimal (denary or base-10) representation, we can start form the right side. The rightmost bit represents 2^0 and the one to its left represents 2^1. If there is a 1 in this column, we add on this value. This example binary string represents 42 since:

$$(0 * 20) + (1 * 21) + (0 * 22) + (1 * 23) + (0 * 24) + (1 * 25) + (0 * 26) + (0 * 27) = 42$$

Integer

In computing, an integer is a common data type in programming languages representing a whole number, with no fractional component. A computer cannot represent all possible integers, since the set of all integers is infinitely countable. Depending on the programming language's implementation, there may be further restrictions based on the number of bits allocated to the number. Some languages have data types allowing a number as large as is storable in memory.

Floating Point

To store fractional numbers in a binary format, it is necessary to convert them to a standardised format. One of the ways of doing this is floating point numbers. There are many different standards, the most common is IEEE 754. This standard stipulates that floating point numbers are stored as a fixed length, with an exponent component and a mantissa component.

An IEEE 754 binary32 floating point number looks like this:

Sign	Exponent	Mantissa
0	**100000111**	**111110100011111**00000000

Hexadecimal

Hexadecimal is a way of representing binary numbers using Base 16. It consists of the numbers 0-9 and the letters A-F. Each hexadecimal character represents 4 bits.

An example of a hexadecimal number is 7B. To convert this to a Base 10 (decimal) number, we use a similar method to binary. The B represents denary 10 and the 7 represents denary 7.

There are several ways of denoting hexadecimal numbers:

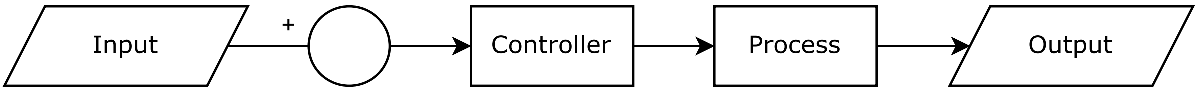
- $0xABCD$ – Used in many programming languages
- $\#ABCD$ – Used for hexadecimal colour codes
- $ABCD_{16}$– Using mathematical notation with a radix

Task Six P5

Control Loop Operations

A control loop is the control functions and components required to adjust a process to reach a desired output. Basic systems vary a process based on a desired output. More complex systems can take in feedback from sensors and react based on this. All systems consist of an input—also referred to as a reference, set point or desired output—and an output.

Open-Loop

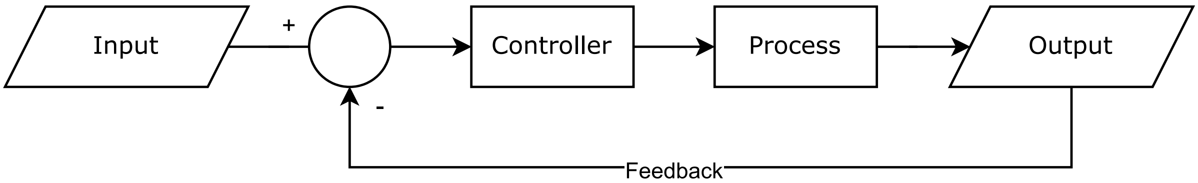


In an open-loop system, the output is not compared to the reference input. The output of the system does not have an impact on the control of the system. Any error within the system needs to be accounted for by the user.

An example of an open-loop system is a space-heater. The device does not know when to stop heating and the action taken by the control system is independent of the output.

Open-loop systems are simpler, making them more economical in some contexts where reliability or precision is not required, or when an output is difficult to measure.

Closed-Loop



A closed-loop system makes use of a feedback loop. The output of the system is continuously fed back into the controller so it can measure the error and compare the desired input to the actual output of the system.

Using the same heating example, an example of a closed-loop system would be a boiler with a thermostat. The boiler can monitor the desired temperature (the input) and compare it to the current temperature (the feedback). The controller can use this information to adjust the process to determine when to turn on or off the boiler.

Closed-loop systems are more complex than open-loop systems, however the incorporation of the feedback makes them more accurate. This is because they can monitor themselves to more accurately handle disturbances and error.

Stages

A control system goes through several stages during its operation. Which take place depends on the specific system; an open-loop system will not incorporate any comparison step since it does not take the current state into account.

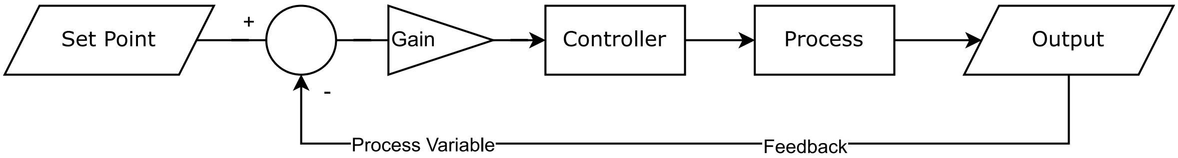
- **1. Measurement**
  During the measurement stage, the system measures the current state of the system using a sensor
- **2. Comparison**
  The system compares the measurement to the desired output
- **3. Calculation**
  The comparison is used to determine what action needs to be taken
- **4. Execution**
  The process is carried out or the action modified according to the calculation
- **5. Evaluation**
  The system returns to the measurement stage and evaluates the change again

Task Seven M3

Proportional Control

Proportional control is a type of closed-loop control system in where the output of the system is directly proportional to the input. It is one of the simplest ways of regulating a device.

Examples of uses of this type of system include controlling and maintaining the speed of motors and engines for adaptive cruise control and automating dangerous machinery.



A proportional control system makes use of three key variables:

- The **set point** or **reference** is the desired output. It is the input to the system from the user. Using the cruise control example, this could be the desired speed of the vehicle.
- The **process variable** is obtained by a sensor. It is the actual output of the system. This would be the actual speed of the car.
- **Error** is the difference between the set point and the process variable. This is what needs to be corrected for – the name proportional control comes from this; the control output is based on the error.

The error is obtained by subtracting the process variable from the set point. If this delta is insufficient in increasing or decreasing the process, **proportional gain** can be applied. This is the process of amplifying the error by a set amount.

Proportional control systems are not precise; there will always be an error and the set point will never be consistently met. Due to the nature of the way proportional control systems work—based on the error and correcting based on that—there is a tendency for the process variable (output of the system) to oscillate above and below the set point (desired output).