Control Circuits

Learners should be able to:

(a) define a microcontroller as a programmable integrated circuit into which software can be loaded to carry out a range of different tasks.

(b) interface sensing circuits and output devices with microcontrollers.

(c) design and analyse flowchart programs to enable microcontrollers to perform tasks.

(d) describe applications of microcontrollers and the reasons for their adoption as standard technology in the vehicle and domestic appliance industry.

**(e) use the following operations in flowcharts:
input/output, counting, branching, testing, time delay and simple arithmetic operations.**

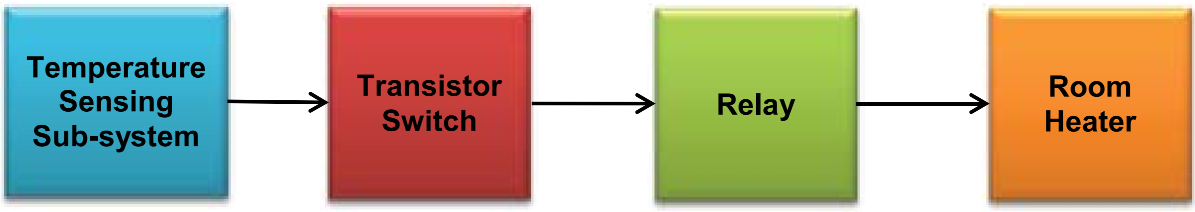
(f) describe the use of a servo motor for positional control in a microcontroller system.

Introduction

We have already covered two types of control system:

1. On - Off Control:

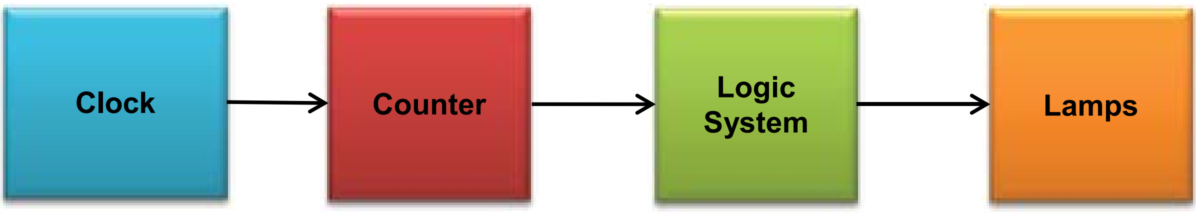
The output is switched either on or off depending on the signal produced by the input sensing sub system which monitors the environmental condition being controlled. A temperature controlled room heater is a typical example of such a control system.



If the temperature in a room falls below a predetermined value the heater will be switched on. The heater remains on until the temperature rises above the predetermined value and the heater will switch off, and so on. The temperature in the room is continually monitored and adjusted automatically.

2. Sequential control based on counters and logic circuits

A traffic light sequence is a typical example of such a system.



The system spends the same amount of time in each of four output states and the sequence continually repeats itself.

In these control systems, the function of the system depended on what components are used and how the components are interconnected. They are said to be ***hard wired***.
Changing the connections and/or the components used is the only way to change the function of the system.

In this chapter, we will consider software control systems using a microcontroller.

Microcontrollers

A microcontroller is a programmable integrated circuit into which software can be loaded to carry out a range of different tasks.

Microcontrollers are totally self-contained. A microcontroller contains its own program memory, data storage memory, bidirectional (input/output) ports and a clock oscillator all in a single IC.

Microcontrollers are widely used in everyday products. In the home, they are used in microwave ovens, refrigerators, dishwashers, washing machines, cameras, telephones, toys and TVs etc.

Modern cars contain several microcontrollers: The engine is controlled by a microcontroller, as are the anti-lock brakes, cruise control, intelligent lights and windscreen wipers.

Other applications include traffic light controllers, industrial robots, security systems, health monitors, automatic livestock feeding etc.

Advantages of using a microcontroller:

- The circuit is more compact as one microcontroller can replace several logic gates/counters/timers.
- More flexible than a hard-wired circuit since a microcontroller can be repeatedly reprogrammed to perform a different function.

Disadvantage of using a microcontroller:

- They are more expensive than other ICs.
- Access to a computer and programming software required.

Programming a microcontroller

Different microcontrollers have different amounts of memory to hold the program that controls them. They read the program in a form called machine code which looks like a long sequence of numbers and letters. Machine code is very difficult to understand, so programs are usually written in other programming languages and converted into machine code using a software program.

There are several control program languages available to program a microcontroller. Once the control program is written on a computer, it is converted into machine code and can then be transferred from the computer to the microcontroller via a cable.

The program area within a microcontroller can store a program even if the power is switched off. A program can be written using a flowcharting program and then downloaded to the microcontroller. It can then be disconnected from the programmer and the microcontroller runs the program independently.

Interfacing to a microcontroller

The number of inputs and outputs that can be connected to a microcontroller depends upon its size. For example, a microcontroller packaged in an 18 pin IC would typically have 15 or 16 pins available as inputs and/or outputs.

Some of the pins on a microcontroller IC may only be used with digital inputs whilst others are dual purpose and can be re-designated or configured either as analogue or digital inputs. In addition, some pins may be configured as either inputs or outputs.

Once you have a design specification for a problem you can choose the most appropriate microcontroller to match the number of inputs and outputs identified in the specification. Data sheets are available for each type of microcontroller giving the possible pin configuration combinations.

All the input sub-systems studied in Component 1 can be connected directly to a microcontroller input pin.

LEDs can be connected directly to a microcontroller output pin. Other outputs can be interfaced to a microcontroller with a transistor or MOSFET, so even high-powered output devices can be accommodated easily.

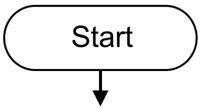
A program can be downloaded and tested using the actual input and output devices. It can then be modified until the system performs exactly as required.

Flowcharts

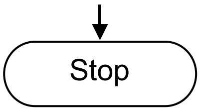
A flowchart is a set of statement boxes linked by arrows used to arrange the different steps in the sequence in a logical order. Flow charts can be used to sort any set of complex instructions, not only in the writing of computer programs.

A flowchart is particularly useful in determining the structure of a computer program. Program commands are written inside boxes of different shapes. The boxes are interconnected by arrowed lines called flow lines.

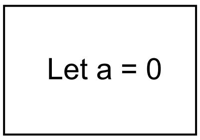
Some of the more common flowchart boxes are given below.



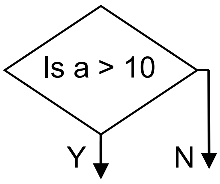
A start symbol is used at the beginning of each flowchart.



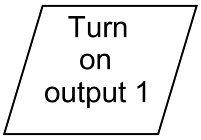
A flowchart may contain one, none or many stop boxes.



A process box is used if a calculation or a delay is required.



A decision box is used to ask a question which can be answered with either YES or NO. If the answer is YES, the sequence follows one route. If the answer is no, it follows a different one.



An output box is used to send data to a particular output.

Flowchart based programming

Converting a flowchart line by line into a control program can be both difficult and time consuming. For this reason, control programs that use a graphics interface have been developed. Flowchart symbols are chosen from a menu and ‘dragged’ onto the computer screen. An editing window allows the contents to be translated into a program avoiding typing or syntax errors. Then the flowchart can be tested and edited.

There are several flowchart control programs available. These include programs systems such as: ‘FLOWAL’, ‘GENIE’, ‘LOGICATOR’, ‘PICAXE’ and ‘FLOWCODE’. All five allow you to simulate a flowchart program on a computer and download it via a cable to a microcontroller situated on either a dedicated interface circuit board or a breadboard circuit.

Note: The flowchart programs in this chapter are written in a generic format as individual flowchart program have their own way of writing commands and setting up the inputs and outputs. Your teacher will show you the specific differences relevant to the flowchart program you will be using.

Example 1:

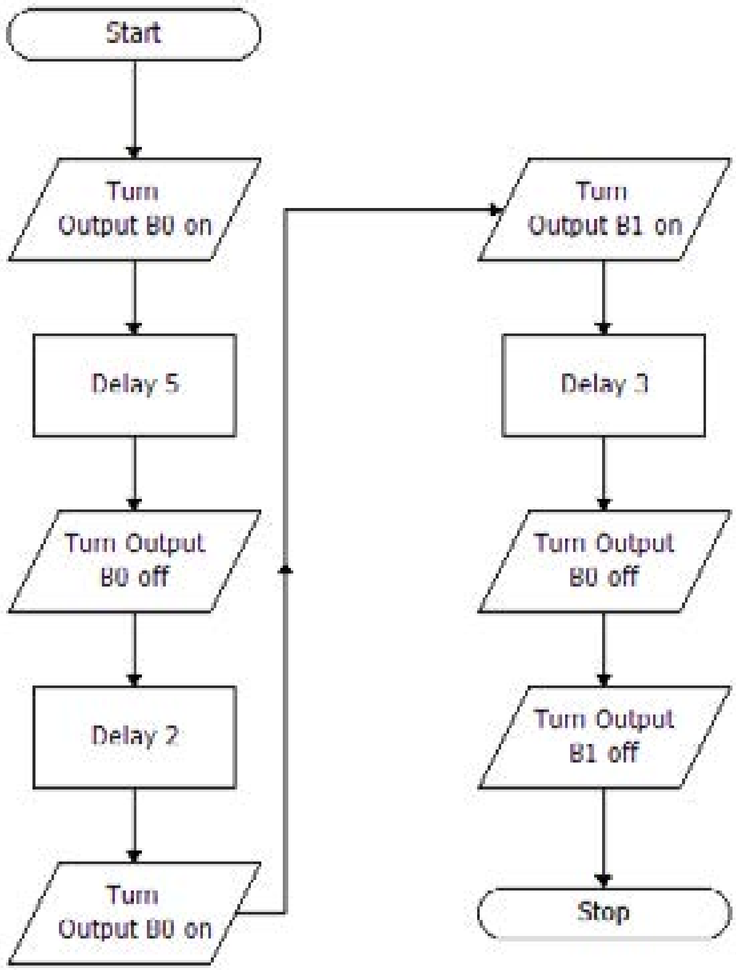
Design specification

A control program is required for a child's toy that:
- switches on a red LED for 5 seconds
- turns the red LED off for 2 seconds
- turns the red LED and a blue LED on for 3 seconds.

Solution 1

Red LED connected to output B0
Blue LED connected to output B1

The following flowchart shows the program.

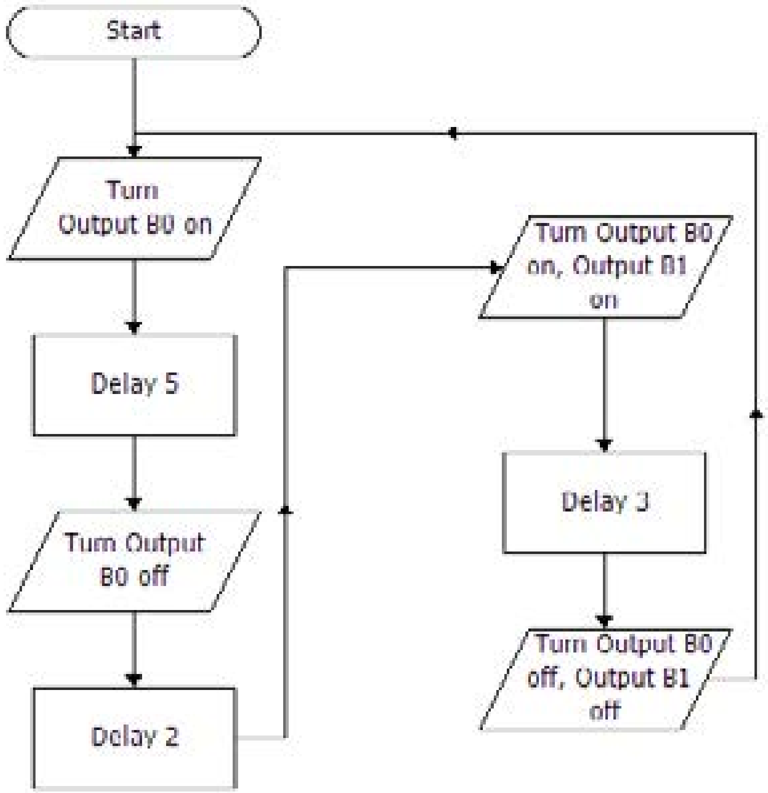


There are two issues with this solution:

- The program runs through the sequence once then stops.
- Outputs are switched on and off, one at a time, which would be tedious if several outputs needed to be activated simultaneously.

Solution 2

Study the modified solution below:



You should notice that when both LEDs have been turned off the program loops back to the first flowchart command so the program runs continuously. Also at the two points in the program where both LEDs are turned on or off this has been achieved with a single command in each case.

Note: Some flowchart programs:

- Use **Wait** rather than **Delay** to represent a time delay.
- Use **High** or **Low** to turn a single output on or off and **Outputs** to turn multiple outputs on and off, whilst others use **Outputs** for both situations.

In example 1 above the design specification stated that a red and a blue LED was required. The solution stated that the red LED was connected to output B0 and the blue LED to output B1of the microcontroller. The flowchart showed for how long B0 and B1 was switched on and off but did not mention what device was connected to them. Sometimes a flowchart will provide information about the actual output device being switched on or off and not mention the microcontroller output pins.

Investigation 4.1

1 (a) Construct the flowchart program shown in Example 1, solution 2 using your flowchart program and test it.

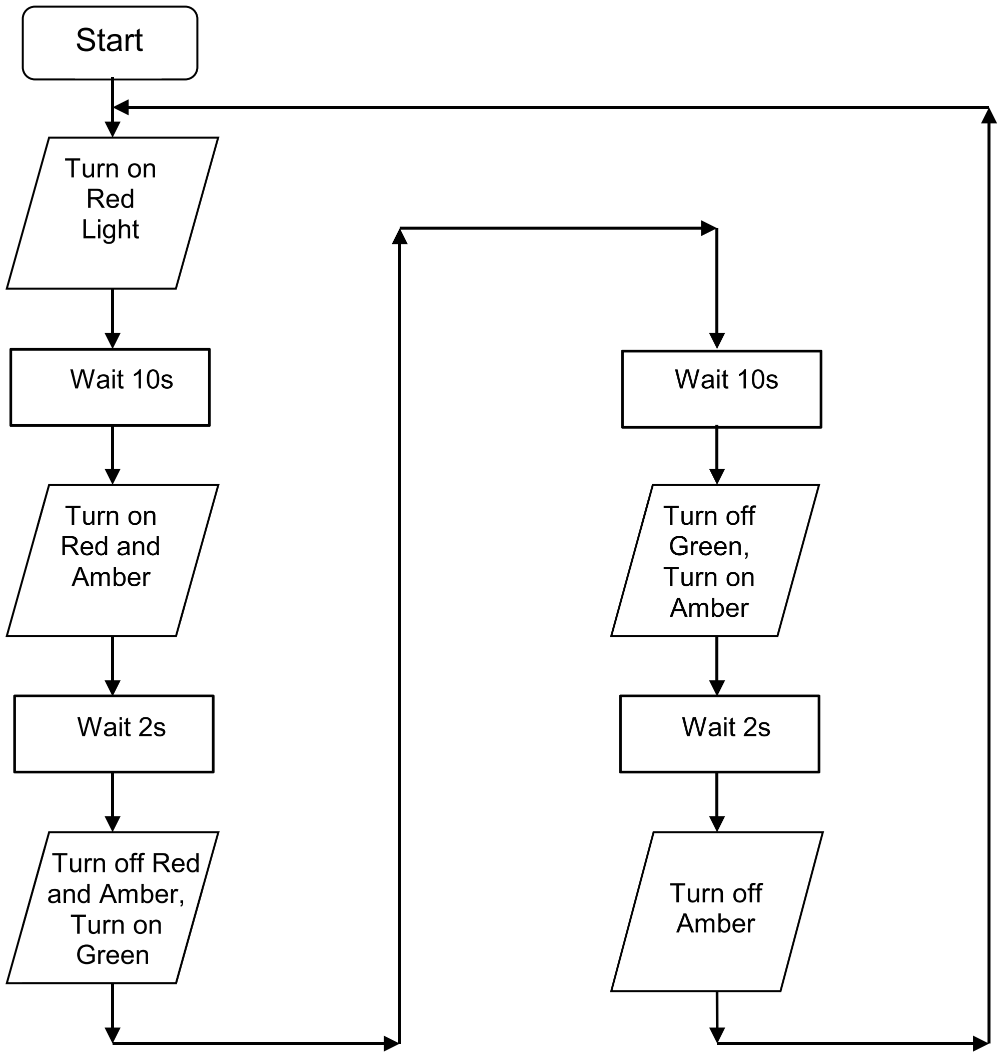
(b) Show the output pins used for each output of your chosen microcontroller:

Microcontroller ....
Red LED connected to output ....
Blue LED connected to output ....

(c) Does the program perform the correct sequence?

....

2. The flowchart shows a traffic light sequence for a single set of traffic lights.



(a) Show the output pins used for each output of your chosen microcontroller:

Microcontroller ....
Red LED connected to output ....
Amber LED connected to output ....
Green LED connected to output ....

(b) Construct the flowchart program and test it.

(c) Does the program perform the correct sequence?

....

(d) Either print your solution or copy it into the space below.

Exercise 4.1

1. Design a control program for the traffic light sequence at a busy crossroad junction. There will be two sets of lights, one in a North-South and one in an East-West. The timing sequence should be as follows.

Step	North-South Lights	East-West Lights
1	Red On (20sec)	Green On (20sec)
2	Red & Amber (3 sec)	Amber (3 sec)
3	Green On (20 sec)	Red On (20 sec)
4	Amber On (3 sec)	Red & Amber On (3 sec)
Repeat Sequence from Step 1		

(a) Show the output pins used for each output of your chosen microcontroller:

Microcontroller ....

North-South Lights

Red LED connected to output ....

Amber LED connected to output ....

Green LED connected to output ....

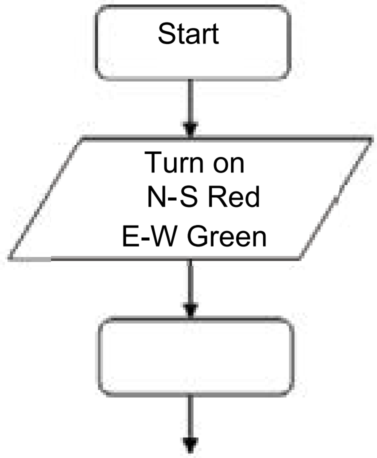
East-West Lights

Red LED connected to output ....

Amber LED connected to output ....

Green LED connected to output ....

(b) Complete the flowchart in the space below.

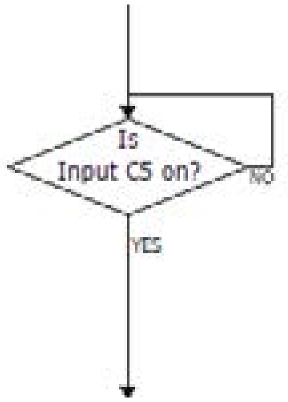


Using decision boxes to monitor inputs

All the programs so far have involved only outputs. Now we look at a program where decisions are made based on the state of the inputs.

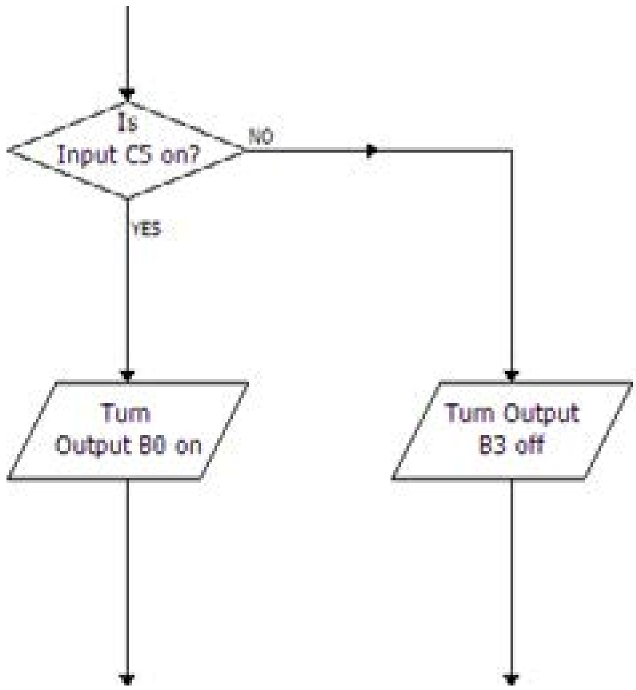
Decision boxes ask questions which have only two possible outcomes, referred to as **Yes** and **No** routes. In programming language decision boxes cause ***branches*** in a program. The program flow is directed one way or another depending on the result of the question.

Digital Inputs
A decision box can be used to cause a program to 'wait' and repeatedly test until an input comes on or goes off. This is shown opposite:



Here the 'NO' path loops back on itself repeating the question "Is Input C5 on?" until Input C5 is actually on. Then the program leaves this decision box and continues through the rest of the program.

A second use of a decision box is to branch to a different part of the program depending on the state of an input, as shown below:



The question is the same - "Is Input C5 on?" but the outcome depends on the answer. If input C5 is **on,** the program will turn output B0 on. If input C5 is **off,** the program will turn output B3 off. The program will enter the decision box only once. The route it takes depends on the state of Input C5.

Example 2:

Design specification

A control program is required to provide a flashing lamp that can be used on a rubbish skip at night to warn drivers to avoid the rubbish skip.

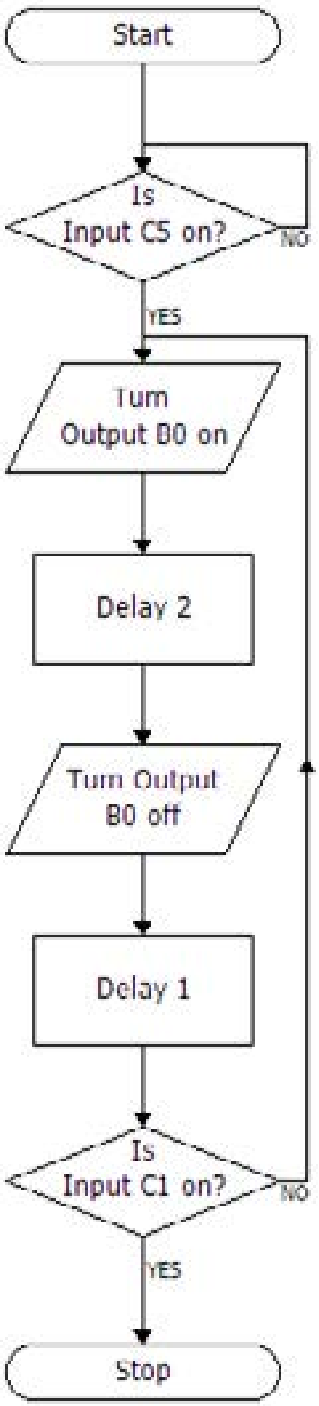
- The light should be switched on when a switch is pressed.
- A lamp should be switched on for 2 seconds then off for 1 second.
- This should repeat until a second switch turns the system off.

Solution

Lamp connected to output B0
Start switch connected to input C5
Stop switch connected to input C1

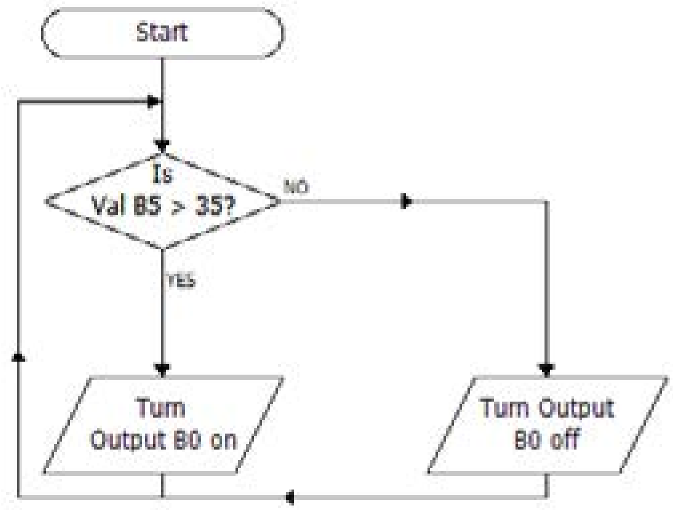
In this example, we have used a decision box to check if the start switch has been pressed. If the switch is not pressed then it is continuously tested until it is. Only when the start switch is pressed will the program continue past the first decision box.

The second decision box changes the flow of the program. If the stop switch is not pressed then the program repeats the lighting sequence before checking the switch again. Once the stop switch is pressed the program changes direction and stops.



Analogue Inputs

The last two examples used a digital, on/off decision box. It is also possible to have an analogue decision box where a value can be tested. An example might be to test an analogue input for the temperature in a bakery.



In the example above, if the Value of the analogue input that's connected to input B5 goes above 35 °C then some cooling fans connected to Output B0 are switched on. When the value of input B5 falls below 35 °C then the cooling fans are switched off.

Example 3:

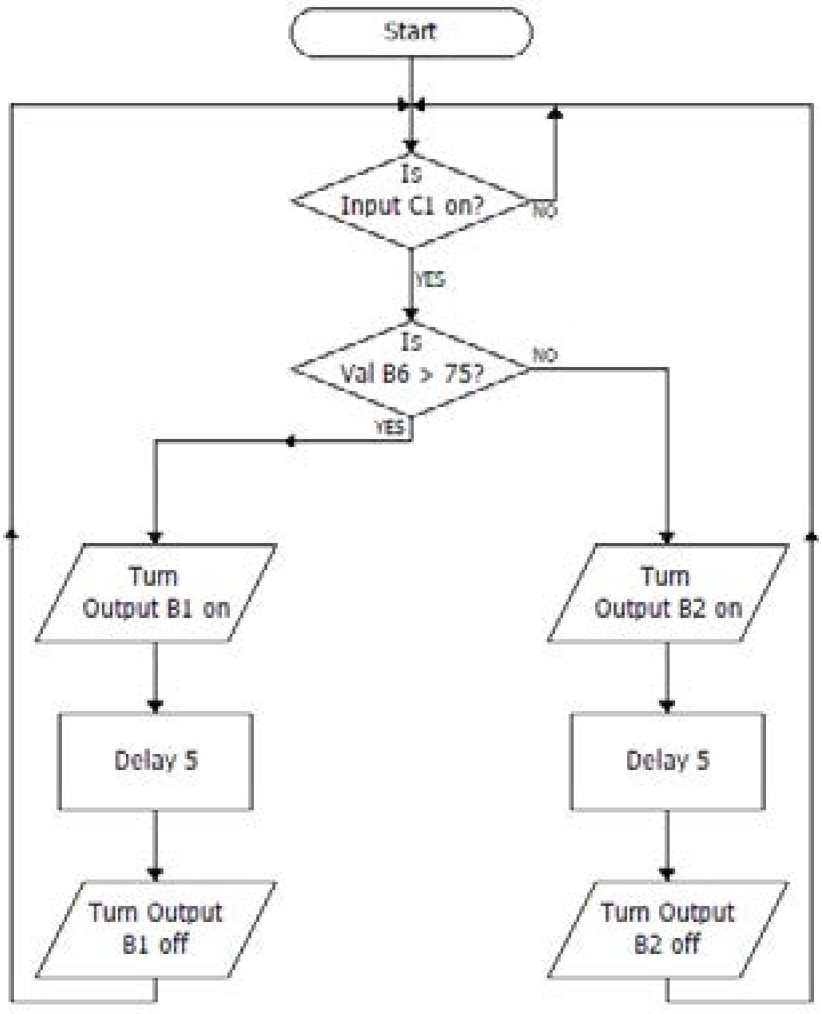
Design specification

An intelligent door bell is required for a family with a young baby.

- If it is light then when a switch is pressed a buzzer should come on for 5 seconds.
- If it is dark, then an LED should come on for 5 seconds.

Solution

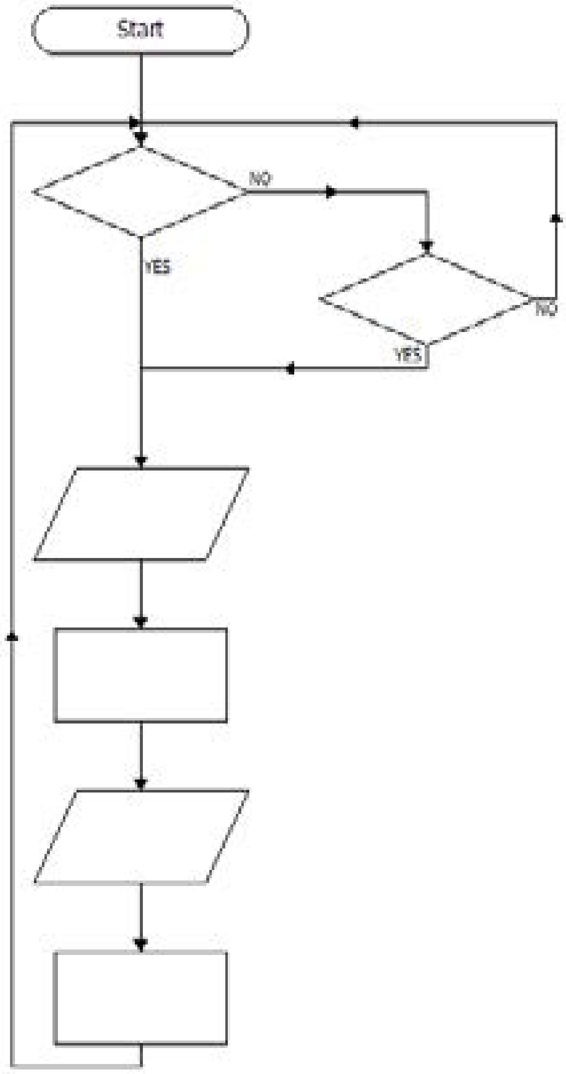
LED connected to output B1
Buzzer connected to output B2
LDR connected to analogue input B6 (Value >75 when Dark)
Stop switch connected to input C1



In this example, we have used a decision box to check if the doorbell switch has been pressed. If the switch is not pressed then it is continuously tested until it is. Only when the doorbell switch is pressed will the program continue past the first decision box. Then, if it is dark (B6>75) then LED is switched on, but if light (B6<75) then the buzzer is switched on.

Investigation 4.2

The Highways Agency requires a safety light system to light a chain of lights along the side of roadworks on a motorway. A flowchart program is required to satisfy the following design specification:



- The system should switch on a warning light that can be placed by the roadworks to warn passing cars and to prevent an accident.
- The lamp should flash when it starts to get dark.
- The lamp should stay on for 2 seconds and then switch off for 1 second.
- This pattern should repeat continuously until daylight occurs.
- The system should include a switch which enables workmen to check the flashing system before they go home.

(a) Complete the flow chart for the program required by writing the correct instruction from the following list, into the six empty boxes:

Switch on warning lamp
Switch off warning lamp
Is the light sensor level < 90?
Has test switch been pressed?
Wait 1 second
Wait 2 seconds

(b) Construct the flowchart program and test it.

(c) Show the input/output pins used for each input/output of your chosen microcontroller:

Microcontroller ....

Lamp connected to output ....

Light sensor connected to input ....

Test switch connected to input ....

(d) Does the program perform the correct sequence?

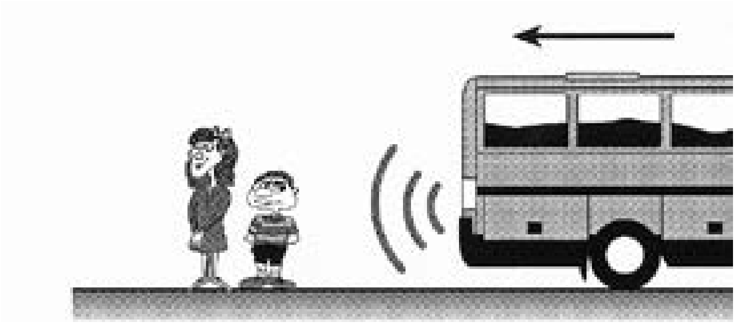
....

(e) Print out your solution.

Exercise 4.2

1. Some school buses have a reversing system controlled by microcontroller.

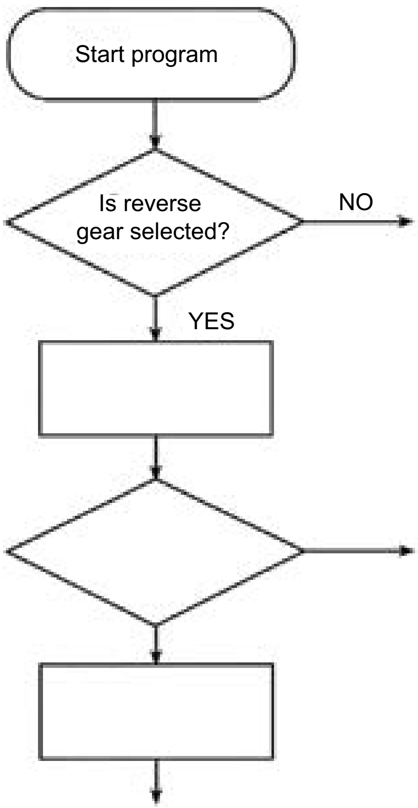
- The system pulses a bleeper when reverse gear is selected.
- The system also switches on a warning lamp inside the bus if it senses something near the back of the bus.



The flowchart shows **part** of the operation of this reversing system.
Some parts of the flowchart have been left out.

(a) Name a suitable device to sense when reverse gear is selected.

....



(b) Complete the flow chart for the program required by:

- Writing the correct instruction from this list in the empty boxes

Is something near? *Switch on reverse bleeper* *Switch on warning lamp*

- Adding correct branches to the decision boxes.
- Writing yes/no on the second decision box.

2. An automatic watering system is needed for an industrial greenhouse.

The specification for the watering system is as follows:

- Watering can only take place when it is dark.
- Watering should only start if the soil is dry.
- The water should be switched on for 15 seconds.
- There should be a delay of 1 minute before re-testing the moisture of the soil.

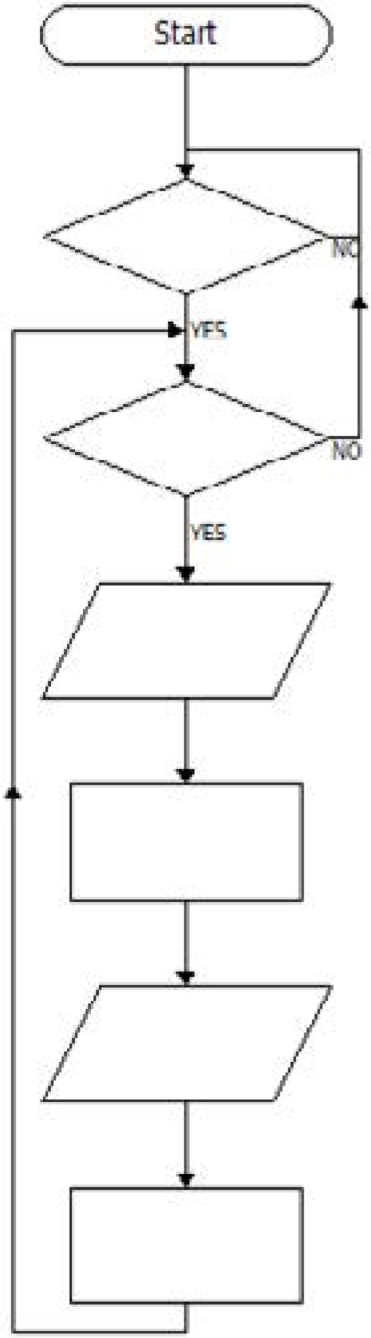
The flowchart shows the program needed for this system. Some parts of the flowchart have been left out.

Complete the flow chart for the program required by:

- Writing the correct instructions from this list in the two decision boxes

  Is it light? *Is the soil dry?*
  Is it dark? *Is the soil wet?*

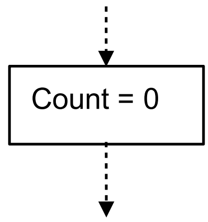
- Completing the Output and Process boxes with suitable instructions.



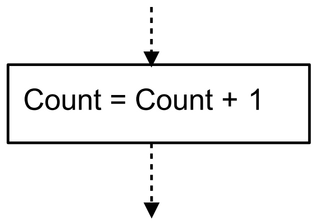
Using Decision Boxes to compare values

Sometimes, we need to repeat sections of the program a set number of times. e.g. to pack 6 cans into a box on a production line. To do this, we setup a count within the program.

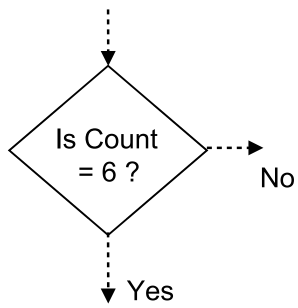
The first thing to do is to set a counter to zero. This is achieved by creating a variable, called ‘Count’ to keep a record of how many times an event has happened, and by setting this to a value of zero, as shown below:



When we need to increase the count by one we use another process box with a different instruction:

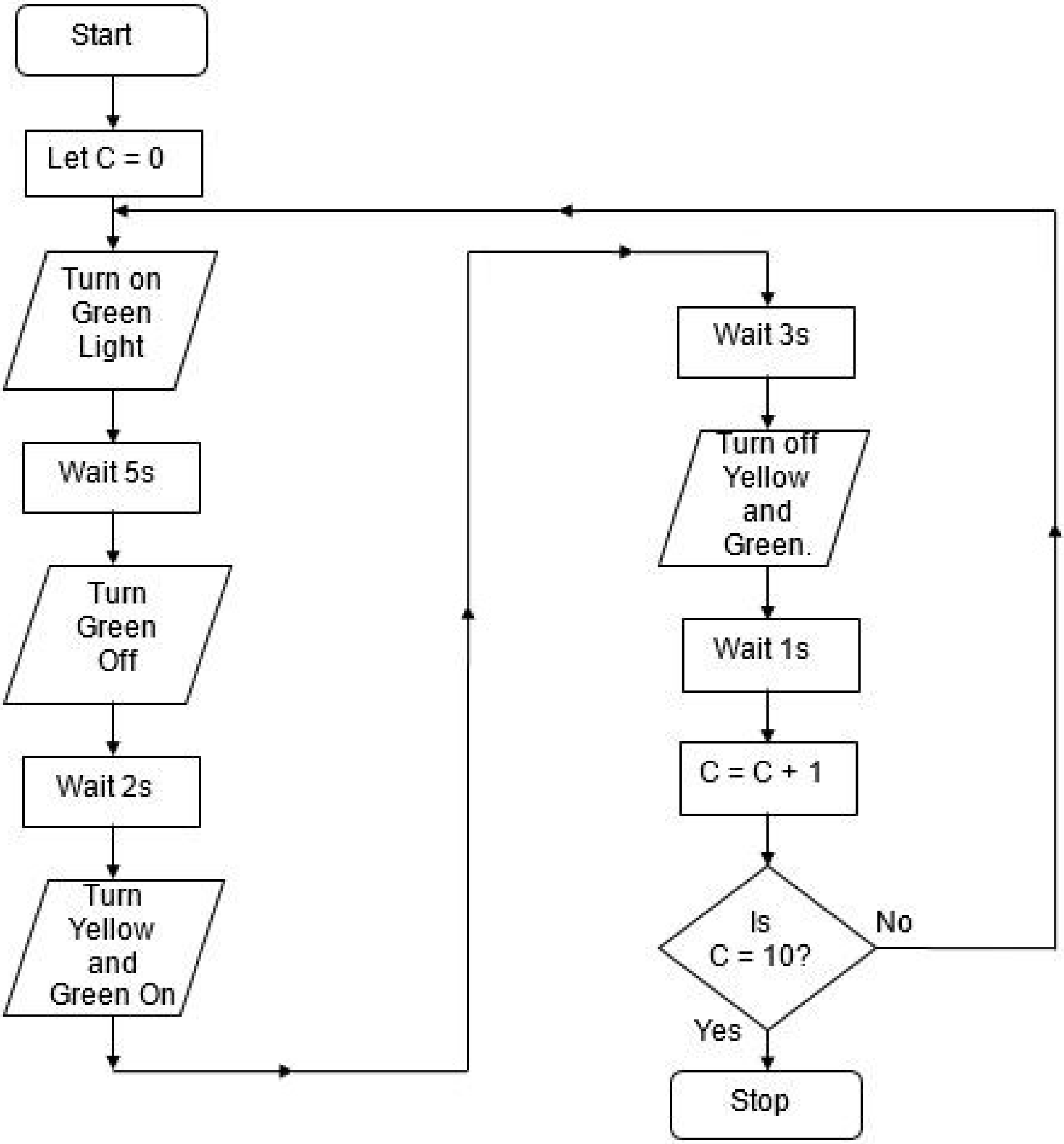


Finally, we use a decisions box to check when we have reached the required count of 6, as follows:



Here the variable ‘Count’ is tested to see if it has reached 6. If it has then the program will follow the ‘Yes’ branch. If it has not reached 6 then it will follow the ‘No’ branch.

Study the flowchart below.



Try and work out what it does

Notice that:

- The **Let C = 0** statement box is used to set a counter to zero at the start of the program.
- The **C = C + 1** box increases the count by 1 each time the sequence repeats itself.
- The **Is C = 10** decisions box monitors the count. When the count is less than 10, the sequence continues. When the count equals 10, the program stops.

Investigation 4.3

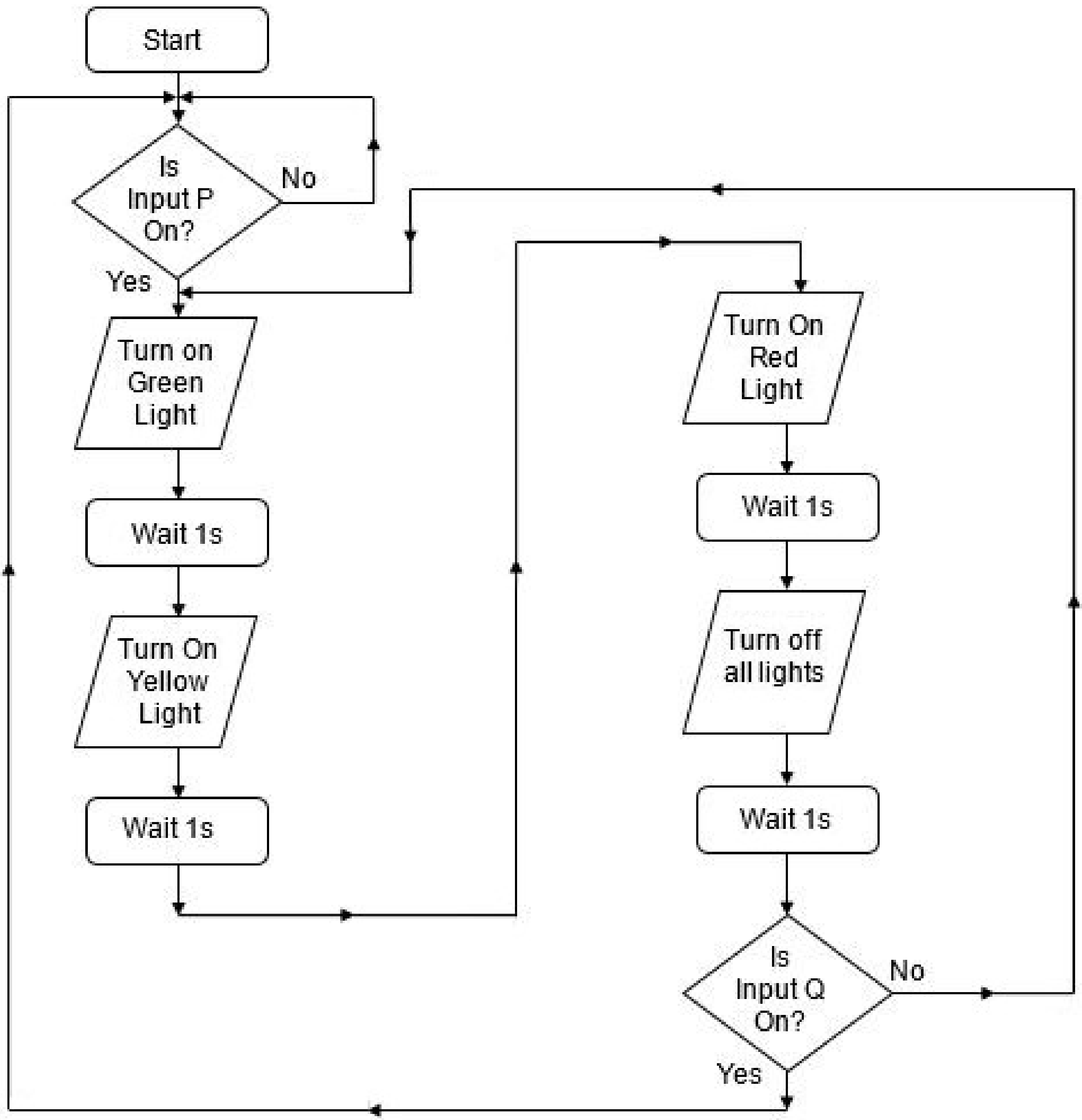
1. (a) Construct and test the flowchart below.

(b) Show the pins used for each input/output of your chosen microcontroller:

Microcontroller.................................... Green light connected to output ....

Yellow light connected to output .... Red light connected to output ....

Input P connected to .... Input Q connected to ....



(c) Describe the programs operation.

....

....

....

....

2. The UK cycling team is busy preparing for the Olympic Games. At the velodrome the team coach wants to have a lap counter fitted to the track to light a lamp when 10 laps of the track have been completed.

A pressure switch is fitted to the track so that when the wheel of the bike passes over it, the switch is closed.

(a) Can you spot a potential problem with this method?

....

(b) Set up the flowchart program shown on page 120.

D6 is checking that the bike is correctly positioned on the start line.

D7 is the pressure switch checking for completion of a lap.

(c) Show the pins used for the output of your chosen microcontroller:

Microcontroller...................... Yellow light connected to output ....

(d) Explain what each step of the program does.

....

....

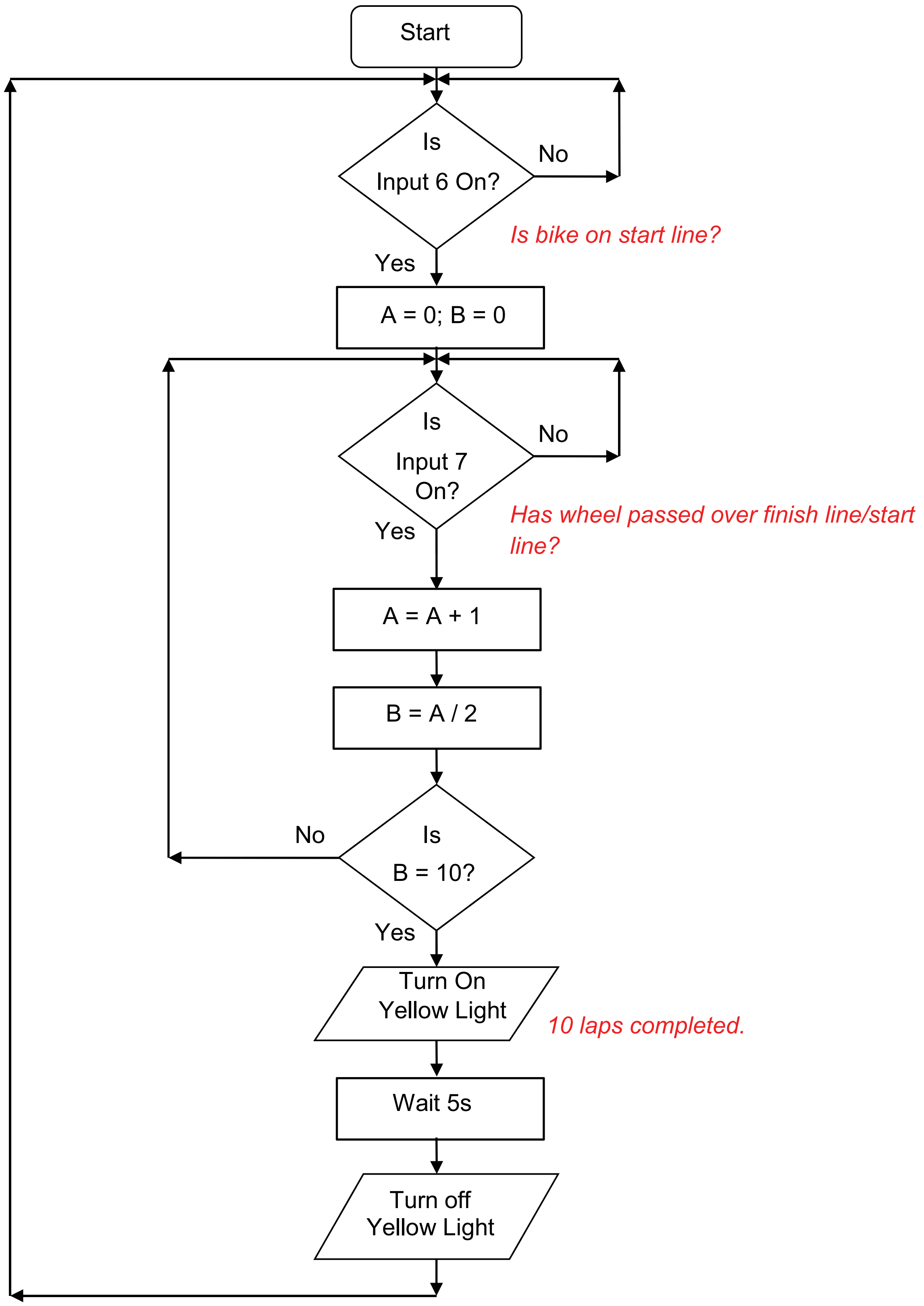
....

....

....

....

The flowchart below provides a possible solution to the problem.



Servo motors

Small servo motors can be very useful in electronic projects. For example:

- In a model car to move levers back and forth to control steering.
- In model railways to operate signals, points and level crossings.
- In toys such as model cranes, tipper trucks, robotic animals and fairground rides.

The picture below shows a typical mini servo suitable for use in toys or models:

The servo motor is an assembly of four parts: a DC motor, a gear reduction unit, a potentiometer and a control circuit.

The function of the servo is to receive a control signal from a microcontroller that represents a desired output position of the servo shaft, and apply power to its DC motor until its shaft turns to that position. It uses a potentiometer to determine the rotational position of the shaft, so it knows which way the motor must turn to move the shaft to the required position. The shaft does not spin freely like a DC motor, it can only turn up to maximum of 180°.

The servo motor requires a pulse every 20 milliseconds and the length of the pulse will determine how far the motor turns.

For example, a 1.5ms pulse will make the motor turn to the 90° position. Shorter than 1.5ms moves it in the anti-clockwise direction toward the 0° position, and any longer than 1.5ms will turn the servo in a clockwise direction toward the 180° position.

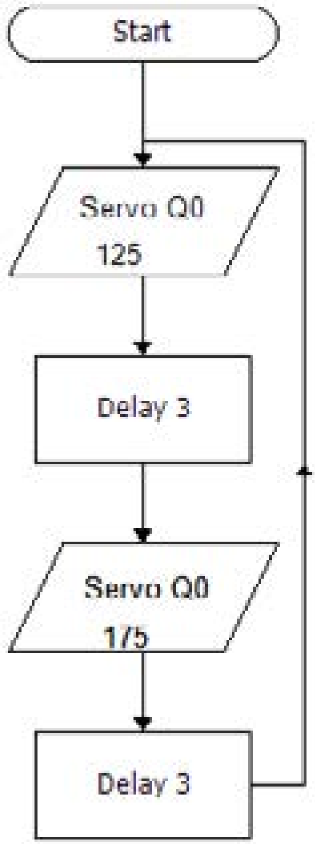
In a flowchart program, a position in range 75 to 225 can be entered to turn the motor to the desired position:

- **75** produces a 0.75ms pulse and turns the motor to 0°.
- **150** produces a 1.5ms pulse turns the motor to 90°.
- **225** produces a 0.225ms pulse turns the motor to 180°etc.

Example 4:

In this example, the shaft of a servo connected to output Q0 moves to position 125 (60°) waits there for 3 seconds then moves to position 175 (120°) for 3 seconds.

The servo shaft continuously moves between these two positions.



To convert a required angle into a servo position

In project work you may need to work out what position to enter into a program for a servo to turn to a certain angle.

This can be done by entering the required angle into the equation.

$$\textit{Servo position} = \frac{\text{angle}}{1.2} + 75$$

Examples: Answers added to equations to show how to round

1. To turn to 82°

$$\textit{Servo position} = \frac{82}{1.2} + 75 = 143.3$$

so position 143 would be entered in the servo command.

2. To turn to 158°

$$\textit{Servo position} = \frac{158}{1.2} + 75 = 206.7$$

so position 207 would be entered in the servo command.

Note: These equations are provided to be used in practical work. You will not be tested on their use.

Exercise 4.3

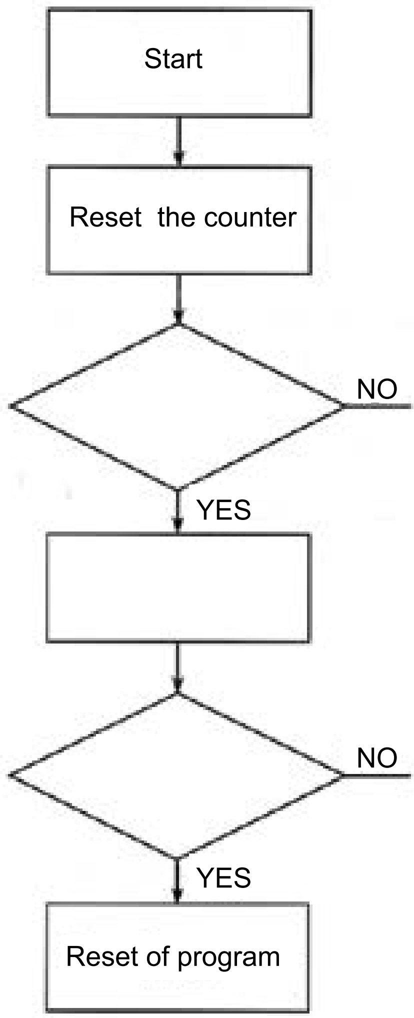
1. A microcontroller is used to pack baked beans tins into a box in a canning factory.

- The microcontroller receives a signal when a tin moves past a sensor on a conveyor belt.
- The microcontroller uses a counter to keep track of how many tins have passed.
- At the end of the conveyor belt, the tins are placed in the box.
- When the box contains 10 tins, it is closed, and replaced with an empty box.

(a) Part of the flowchart for this control system is shown below.

Add these instructions to the correct boxes in the flowchart:

Add 1 to 'count' *Has a can passed?* *Is 'count' equal to ten?*



(b) Add the links to the two decision boxes to show how the flowchart branches when the answer to the decision box question is 'No'.

2. A lift can carry 9 persons safely. A flowchart program is used to count the number of persons and sounds an alarm if 10 people have entered the lift.

(a) Name a suitable sensor to be used at the entrance.

……..

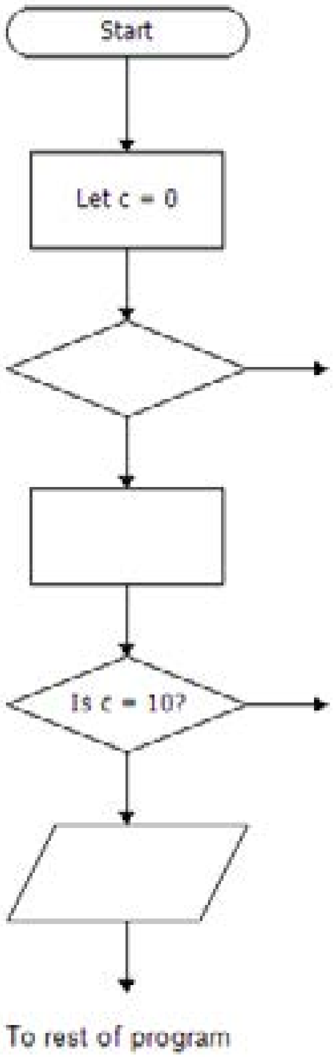
(b) The program makes use of a counter, c. How does the signal from the sensor affect the counter?

……..

……..

(c) Complete flow chart for the program required by:

- writing the instructions in the empty boxes
- adding correct branches to the decision boxes
- writing yes / no on the first decision box.



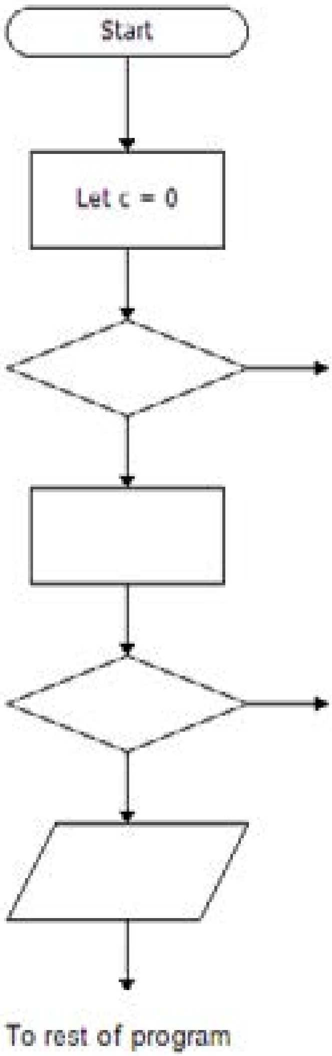
3. A maximum of 100 cars can enter a car park. A microcontroller program is used to count the number of cars and close a barrier if 100 cars have entered.

(a) Name a suitable sensor to be used at the entrance.

....

(b) Complete the following flow chart for the program required by:

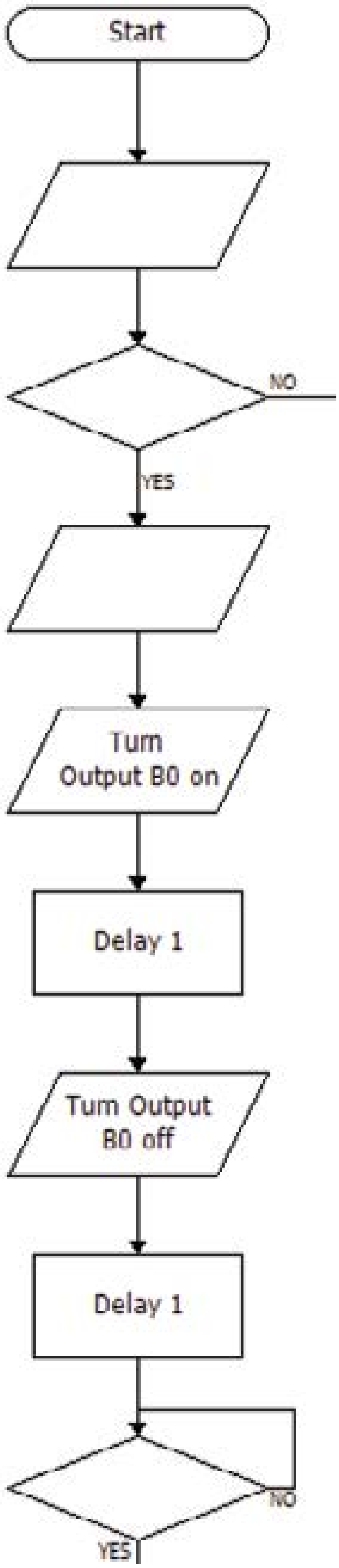
- writing the instructions in the empty boxes
- adding correct loops to the decision boxes
- writing Yes/No on the decision boxes.



4. A microcontroller is used to control a level crossing on a model railway.

Reed switches are placed on the track before and after the level crossing. A magnet on the underside of a train operates the first reed switch when the train passes. This causes the barrier to be lowered through 90° and a lamp to flash on and off.

When the train has passed the level crossing the second reed switch operates, the barrier returns to the upright position and the lamp switches off.



(a) Part of the flowchart for this control system is shown on the right.

The lamp is connected to output B0

Add these instructions to the correct boxes in the flowchart:

Servo C0 150 *Servo C0 75*

Is reed switch 1 on? *Is reed switch 2 on?*

(b) Complete the links from the two decision boxes to show how the flowchart branches.

(c) When the system was tested it was found that the lamp did not flash on and off for as long as expected. State the reason for this and suggest how the problem can be rectified.

....

....

....

....

....