Flowcharts & Programming

Microcontrollers

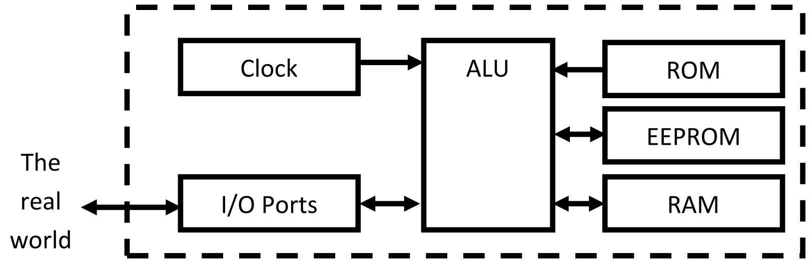
Advantages of microcontrollers

- Increased reliability compared to a hard wired circuit as there are fewer components to break.
- Reduced quantity of component stock needed because one microcontroller can replace several electronic components.
- Simplified product assembly and smaller end product.
- Greater product flexibility and adaptability, since the features are programmed into the microcontroller rather than being hard-wired into the electronic hardware.
- Rapid product changes can be made by rewriting the programme rather than rewiring the electronic circuit/hardware.

Disadvantages of microcontrollers

- They need software and access to a computer to be reprogrammed.
- Software may be expensive.
- Software may need to be updated frequently.
- They are often more expensive than other ICs.

Inside a Microcontroller



RAM

The RAM (Random Access Memory) is a temporary memory that is used for storing information while the programme is running. This memory is volatile which means that when the power to the microcontroller is disconnected the RAM memory is lost.

ROM

The ROM (Read Only Memory) contains the operating instructions for the microcontroller. The ROM is programmed before the microcontroller is installed and the memory is permanent even if the power supply is removed.

EEPROM

EEPROM (Electrically Erasable Programmable Read Only Memory) is a special type of memory that runs on a microcontroller. Like normal ROM, it keeps the programme when the power supply is removed but it can be reprogrammed when needed. This means the microcontroller will start to run the program currently in its memory whenever the power supply is connected.

ALU

The processing unit (Arithmetic and Logic Unit) is the control centre of the microcontroller. It operates by reading instructions from the ROM and then carrying out the mathematical operations for each instruction.

Clock

The clock circuit controls the speed at which these operations occur. It synchronises all the internal blocks (ALU, ROM, RAM etc.) so that the whole system works correctly.

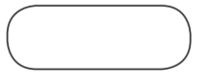
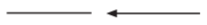
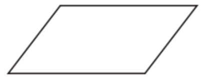
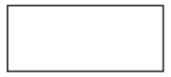
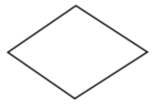
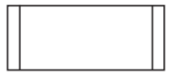
Buses

Information is carried between the various blocks of the microcontroller along groups of wires called buses.

I/O Ports

The I/O ports (Input/output ports) are the communication lines between the microcontroller and the "real world". This is where any input or output devices would be connected.

Flowcharts

Terminator symbol		Used for the start and end of a main program or sub-procedure.
Line symbol		Shows the direction of program flow. For flow down or to the right, an arrow is not needed. For flow upwards or to the left, arrows are added.
Input/Output		Used to control outputs or to show that data is being received.
Process symbol		Used for operations which take place within the microcontroller, for example a delay.
Decision symbol		Program flow is determined by a "yes" or "no" answer to the question in the box.
Sub procedure symbol		Used to call a sub-procedure.

Variables

```
symbol counter = b0          'rename variable b0 'counter'
symbol green = 5             'rename output 5 'green'

flash:
    for counter = 1 to 10    'start a for... next loop
        high  green          'switch green on
        pause 1000           'wait 1 second
        low  green           'switch green off
        pause 1000           'wait 1 second
    next counter             'add 1 to counter
    end                      'ends program
```

Symbols

We can rename the pins with a symbol to make it easier to understand. For example pin 5 above has been renamed as green.

Variables

When we need part of a programme to repeat multiple times we use a "for...next" loop. We select the part of the microcontroller that will store the number of times it has looped. They are labelled b0 to b9.

Motor Control

When we connect a motor to a microcontroller, we can control the direction of rotation depending on which output is switched on. The table shows how to change the direction of the motor.

Pin 4	Pin 5	Direction
Off	Off	Off
Off	On	Clockwise
On	Off	Anti-clockwise
On	On	Off

Inputs and Outputs in PBasic

If you are usinng a Basic Stamp board you will need to tell the microcontroller which pins are inputs and outputs before you write your programme. (We don't need to do this when using PICAXE boards.)

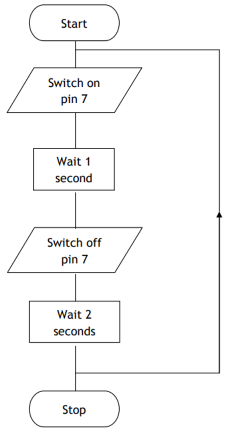
If we want pins 0-3 to be inputs and pins 4 to 7 to be outputs we would write:

Let dirs = %11110000

0 is an input and 1 is an output. The pins go from 0 on the right to 7 on the left.

Then you would add your programme below this.

Switching on outputs



```
symbol red = 7          'rename output 7 'red'

main:
    high red            'set pin 7 on
    pause 1000          'keep pin 7 on for 1 second
    low red             'set pin 7 off
    pause 2000          'keep pin 7 off for 2 seconds
    goto main           'jump back to main
```

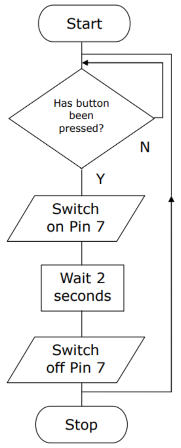
Labels

Labels are words that are used to group parts of the programme. For example "main" is used at the start of the programme. If we want to create a loop then we use the command "goto main" to jump back to where the word "main" is in the programme. Labels can have any name at all.

Comments

The green comments are there in the programme in plain English to help us understand what is happening.

Making decisions

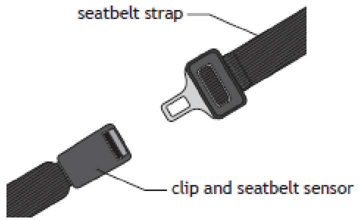


```
main:
    if pin0 = 1 then light
    goto main
light:
    high 7
    pause 2000
    low 7
    goto main
```

Decisions

Decisions are used to test inputs. In this programme the question is has the button been pressed? The programme asks "if pin0 = 1 the light", if the button is not on then the programme will jump to the next line which is "goto main" and then it will keep asking the same question until the button is pressed, then it jumps to the label "light".

8. A car safety system detects when the driver's seatbelt is unfastened.



The safety system is operated by a microcontroller.

Input and output connections to the microcontroller are shown in the table below.

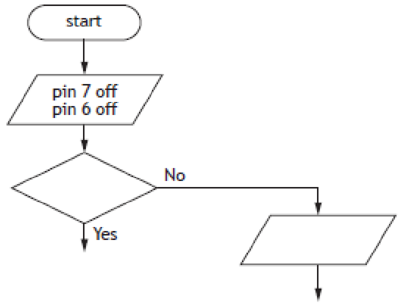
Input connections	Pin	Output connections
	7	warning lamp
	6	buzzer
seatbelt sensor	0	

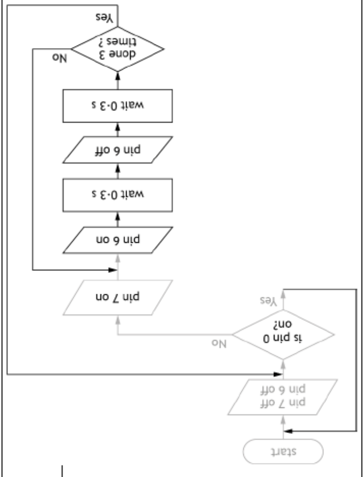
The safety system operates using the following sequence:

- A warning lamp and a buzzer turn off.
- When the seatbelt sensor is on, the sequence will return to the start.
- If the seatbelt sensor is off, the warning lamp will turn on.
- The buzzer will then turn on and off three times over a total period of 1.8 seconds.

(a) Complete the flowchart for the sequence shown opposite, with reference to the Data Booklet and input/output connections.

Include all pin numbers and delay units in your flowchart. 10



(a)			10	Pin 0 on ? decision - 1 mark. Yes loop to start (arrow provided) - 1 mark. Pin 7 on in correct position - 1 mark. Pin 6 on and off in correct position - 1 mark. Both delays in correct position - 1 mark. Total delay time with unit = 0.6s - 1 mark. x3 ? decision with Y/N - 1 mark. Correct repeat sequence loop with arrow back to before pin 6 on - 1 mark. Loop after sequence with arrow back to before pin 0 - 1 mark Accept (on, off, delay) sequence repeated three times - 2 marks. All symbols correct - 1 mark. Ignore any additional steps.

9. (continued)

A microcontroller based system is used to detect the bottles.

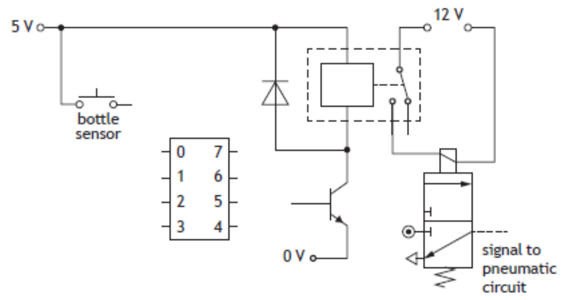
The program used to count six bottles is shown below.

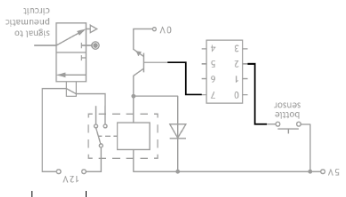
```
line   program
1      main:      let count = 0
2      check:         if input2 is off then check
3                     let count = count + 1
4                 if count = 6 then label_1
5                 goto check
6      label_1:   switch on 7
7                 pause 500
8                 switch off 7
9                 goto main
```

(c) State the line number that contains a time delay. 1

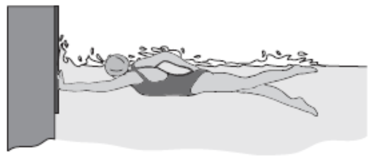
An incomplete diagram for the microcontroller based system is shown below.

(d) Complete, with reference to the program above, the wiring of the bottle sensor and the transistor to the microcontroller. 2



(c)		(Line) 7	1	
(d)			2	Bottle sensor correctly wired to pin 2 - 1 mark. Pin 7 correctly wired to base of transistor - 1 mark. Do not accept any additional pins wired to bottle sensor/transistor.

9. In a swimming competition, a system is used to automatically measure a competitor's time.



The system is operated by a microcontroller.

The input and output connections to the microcontroller are shown in the table below.

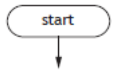
Input connections	Pin	Output connections
	7	buzzer
	6	timer
lane switch	1	
master switch	0	

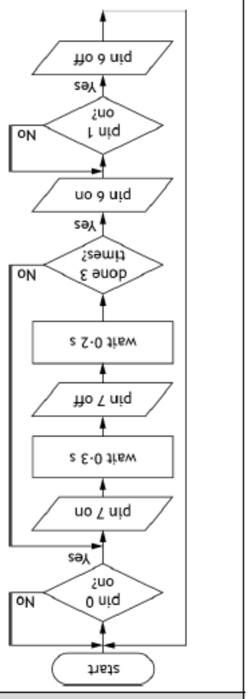
The system operates using the following sequence.

- A master switch is pressed
- A buzzer then sounds on and then off three times over 1.5 seconds
- The timer then starts
- When a lane switch is pressed the timer stops
- The system will then reset ready to be used again

(a) Complete the flowchart for the sequence, with reference to the data booklet and input/output connections.

Include all pin numbers and delay units in your flowchart. 10



	10	Pin 0 on ? with Y/N, loop and arrow in correct position - 1 mark. Pin 7 on and off in correct position - 1 mark. Both delays in correct position (ignore incorrect values and unit) - 1 mark. Total delay time(s) with unit = 0.5 s per cycle - 1 mark. X3 decision with Y/N in correct position - 1 mark. Loop and arrow (x3 decision) back to before pin 7 on - 1 mark. Pin 6 on and off in correct position - 1 mark. Pin 1 on? with Y/N, loop and arrow in correct position - 1 mark. Accept decision feedback loop to before pin 6 on. Continuous loop to start with arrow - 1 mark. All marked symbols correct - 1 mark. Ignore any additional steps. accept on/off repeated 3 times max 2 marks.

9. (continued)

A program used in a different control system is listed below.

```
line   program
1      main:      let count = 0
2      label_1:      switch on 4
3                    switch on 5
4                    pause 600
5                    switch off 4
6                    switch off 5
7                    pause 600
8                    let count = count + 1
9                 if count = 20 then label_2
10                goto main
11     label_2:   if Input0 is on then label_3
12                goto label_2
13     label_3:   switch on 7
14                pause 3000
15                switch off 7
16                goto main
```

(b) Describe the function of line 16 in the program. 1

Lines 2 to 9 should repeat twenty times before moving on to line 11.

During testing an electronic engineer found that this did not happen.

(c) Explain why lines 2 to 9 did not repeat twenty times before moving on to line 11. 2

(b)		To go back to line 1/main/restart the program To create a continuous loop.	1	Descriptive response. 1 mark for looping program back to start. Accept reset the program. Do not accept go to main on its own.
(c)		The program loops back to line 1/main/"let count = 0"/wrong line ...therefore it will reset the count/ the count will not pass 1/count will not reach 20.	2	1 mark for program looping back to the line 1 (cause). 1 mark for resetting the count (effect).