SMART

BOX

User Guide

ECONOMATICS

Smart Box User Guide

The manual contains four sections:

SMART BOX	A guide to using the Smart Box.
SMART MOVE PC	Tutorial and reference guide to the PC version of the software.
SMART MOVE MACINTOSH	Tutorial and reference guide to the Macintosh version of the software.
SMART SENSORS	Information sheets on individual sensors.

ECONOMATICS (EDUCATION) LTD
Epic House, Darnall Road, Attercliffe, Sheffield S9 5AA, UK
Telephone: (0742) 561122 Fax: (0742) 439306
International: +44 742 561122 Fax: +44 742 439306

L60/026/01

SMART BOX

Smart Box

ECONOMATICS

Copyright

Trademarks

Limited Warranty

Smart Box, Smart Move and Smart Sense sensors are manufactured for Economatics (Education) Limited.
Economatics (Education) Limited warrants that these products and accompanying documentation comply with the specification applied to them and will be free from defect in materials and workmanship for a period of 12 months from their date of purchase. This warranty shall not apply if the products are used otherwise than in accordance with the manufacturer's instructions or if any alterations or repairs are made without the Manufacturer's approval. The Manufacture's liability under this Warranty shall be limited (at the Manufacturer's option) to the repair or replacement of the defective product or the return of the purchase price. Save as mentioned above, in no event will the Manufacturer be liable for any direct indirect special or consequential loss by reason of any representation implied warranty or other term in connection with the supply of the products or their use, failure to perform or otherwise.

THIS LIMITED WARRANTY IS IN ADDITION TO AND DOES NOT AFFECT ANY STATUTORY RIGHTS YOU MAY HAVE AS A CONSUMER.

Contents

Introduction

What the Smart Box package includes

Check that you have all the items listed below, If you don't have all of these items, or if any of them are damaged, notify your supplier immediately.

The package contains

Smart Box pack:
A Smart Box
A mains power cord

Software/ Connector pack:
Smart Move software
A serial communication cable
This manual
A PC Keystrip
A registration card

Register your Smart Box

Please complete and return the registration sheet. This will enable Economatics or your distributor to keep you informed of new sensors and updates to your software.

About Smart Box

Smart Box has been designed to allow you to monitor the environment and control external devices using your computer. Smart Box is itself a small computer which has sockets to allow the connection of many types of sensors and other devices. Because Smart Box does not have its own keyboard or monitor it receives all its instructions via the cable which connects Smart Box to the 'host' computer. The 'Serial' port at the back of Smart Box is connected to the serial port on your computer.

The 'ports'

Smart Box has a number of different sockets which are used for the external connections. These are divided into two types, the sensor ports and the output ports. The sensor ports are used to measure external signals. Smart Box can handle both analogue and digital sensors. Analogue sensors produce a signal which can vary continuously and sometimes very rapidly, for example the signal from a sound sensor can change several thousand

times every second whereas a temperature sensor may change relatively slowly. Smart Box can also be used with digital sensors. These will only switch on or off. A pressure mat or a thermostat which switches at a preset temperature are examples of digital sensors.
The output ports are used to switch on and off external devices such as lamps and motors. The 8 'Digital Outputs' can each be used to switch on or off a low power device such as a buzzer or lamp. The motor outputs have a greater power capability and can also be used to drive a dc motor forward and reverse.

Rear panel

On the rear panel of Smart Box you will find more connectors. The power inlet socket is used to connect the mains cord. This socket includes a fuse holder. In the unlikely event that this fuse should need replacing ensure that only the correct type is used. The rear panel also houses the serial port and the 'reset' button, which is used to reset Smart Box if the host computer is powered up after Smart Box (see p.8).

Software

Once Smart Box is connected to the host computer, you will need to load appropriate software to be able to program Smart Box in the way you require. The Smart Box package includes the Smart Move control software. This package is intended to provide a powerful enviroment for writing programs to control models and other devices which you have attached to Smart Box. It also allows some capture of the data from sensors and quite sophisticated data logging can be programmed. See the Smart Move section for further details.

Setting up

Connecting Smart Box to your computer

Smart Box is connected to your computer using a serial port on each machine. Smart Box is supplied with a serial cable to suit your computer. One end of the cable is fitted with a 5 pin DIN plug which plugs into the serial socket on the back panel of Smart Box. The other end of the lead plugs into your computer's serial port.

PC compatible computers

Plug the lead into the socket marked ‘Serial port’. The lead is supplied with an adaptor to fit 25 pin serial ports where appropriate.

Apple Macintosh

Printer Modem

The Macintosh has two serial ports at the rear known as the modem port and the printer port, which are identified by the two symbols shown here. Smart Box can be connected to either but the modem port is usually used for this type of device. Note, on the Macintosh PowerBook 100 the printer port is the only external serial port.

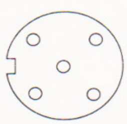
Acorn Archimedes

The Archimedes has a single 9 pin serial port on its rear panel. If your Archimedes has a label with the wording ‘Serial port not fitted’ it will require the serial upgrade to be installed. Contact your local Acorn dealer.

RM Nimbus PC 186

The RM Nimbus computer has a serial port labeled 'Aux or Piconet' on its rear panel. Use this port for Smart Box. Note, when disconnecting the lead push the latch down before pulling out the plug.

BBC Micro / Master computer



The BBC Micro serial port is on the rear of the case. The lead can be connected in two ways, ensure that the bump is to the left when viewed from the rear as shown in this diagram.

Setting up procedure

1 Make sure that your computer is switched off.

2 Connect the serial cable between your computer and Smart Box

3 Plug the power cord supplied with Smart Box into the power inlet socket on the rear panel

Warning *This equipment is intended to be electrically grounded. The power cord supplied is fitted with a three pin plug and should only be used with the correct three pin - grounding type - outlet.*

4 Switch on your computer in the normal way.

5 Switch on Smart box. The switch is on the moulded power inlet socket. You will see the red 'on' LED light on the top of the box. All the Green LEDs will also flash. Smart Box is now ready for use.

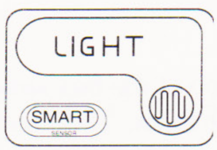
Note, if Smart Box is powered up first it may receive spurious signals from the serial port as the host micro starts up, in this case press the reset button on the rear of Smart Box to make sure that it is ready to receive commands from your software.

Sensors

What are sensors?

Sensors are devices which measure some physical quantity or movement and send an electrical signal back to Smart Box. This in turn converts the signal into a form that the software can understand. The sensors provide *feedback*. An example of a system involving feedback might be a plant heating system. The computer can switch on and off a heater, but it needs to be able to measure the temperature of the environment being controlled in order to determine when to switch the heater on and off. Sensors can either be Digital or Analogue. Analogue sensors produce a voltage which can vary continuously, but like most computers, Smart Box can only handle digital signals internally. An analogue to digital convertor converts the analogue signals into a form that Smart Box can understand.

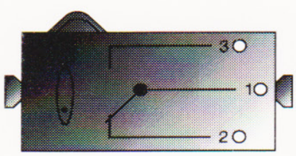
Analogue sensors



Light sensor

Smart Box has a number of sockets which enable you to connect sensors that can be used to monitor and measure the environment. A temperature sensor is provided in the Smart Box pack. This plugs into any one of the four analogue sensor sockets on the side of the case. The Smart Move software can identify which analogue sensors are connected and calibrate the software accordingly. A range of sensors can be connected to the analogue sockets including light, sound, position and humidity.

Digital sensors



Microswitch

Digital sensors are connected to the red and yellow sockets on the top of the case. The software can detect if the switch connected between the sockets is open or closed. Digital sensors are useful for detecting a 'condition' such as when a door is open or when a buggy has hit a wall. The digital sensor inputs can also be used for counting. For example the rotation of a wheel can be accurately controlled if a microswitch is positioned so that it is closed as the wheel rotates.

Making your own sensors

You can make your own analogue and digital sensors for use with Smart Box. Analogue sensors tend to be slightly more complex as they produce a voltage which varies depending on the condition of the sensor. Digital sensors on the other hand simply complete the circuit between the red and yellow sockets.

Analogue sensors

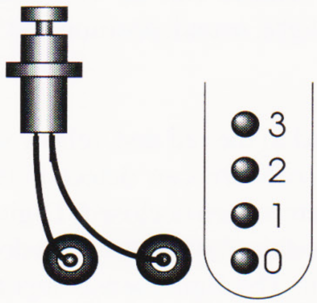
You can connect your own analogue sensors to Smart Box via the sensor adaptor. This brings the connections from the analogue sensor sockets to 4mm connectors on the lid of the adaptor. The sockets are marked Input, Ground and Vref.

The Smart Sense range of analogue sensors are identified by Smart Box when they are plugged in. The sensor adaptor allows other sensors to be recognised.

The voltage between the input and ground sockets should vary between 0 volts and a maximum of 2.55 volts. The Vref socket provides a 2.55 voltage source. The inputs to Smart Box are protected against over voltage.

Digital sensors

The eight input sockets on the top of the box simply require a connection to be made between them. This can be a switch such as a push switch or microswitch, or any device which will complete the circuit. Connect your 'switch' to Smart Box using a length of 2 core cable terminated with 4 mm plugs.



Outputs

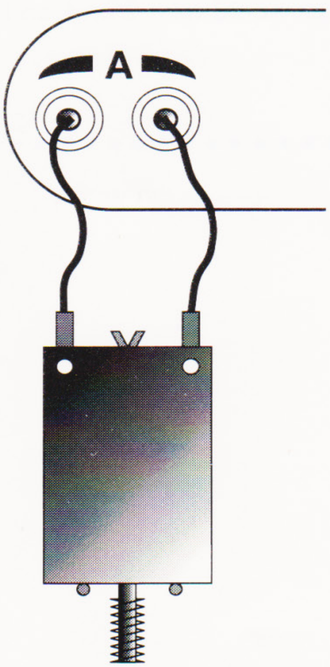
What are Outputs?

Outputs are the physical connection between Smart Box and an external device that it is controlling such as a lamp or motor. Smart Box has two types of output sockets, the blue and red 'Digital Outputs' on the top of the case and the 'motor' outputs on the side.

Digital Outputs

The eight digital outputs are designed to switch on and off lamps and buzzers or similar devices. Each output can be switched on and off individually and, using the Smart Move software, the power from each socket can be varied so that, for example, a lamp can be dimmed . The device is connected between the red and blue sockets. When the output is 'on' 4.5 volts will be available between each pair of sockets and the indicator beside each socket will light.

Motor Outputs

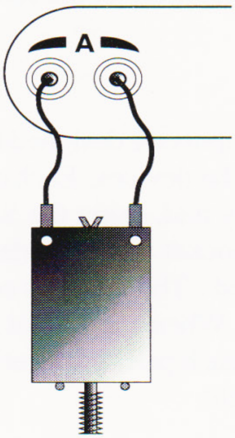


The motor outputs are pairs of sockets labelled A,B,C,D on the side of the case. These pairs of sockets give 5 volts. They can be switched on and off and the direction of current flow can be changed to allow a motor to be reversed. As with the digital outputs, the power output can be varied through the software, so that motor speed can be controlled.

Warning *The outputs on Smart Box are protected against a 'short circuit' by overload protection, but as with any electrical power outlet you should take care to avoid shorting the outputs as large currents can flow. Tip - When connecting devices that have sockets on them such as a motor, plug the leads into the motor before you plug them into Smart Box so that the connectors cannot touch together.*

Connecting your own output devices

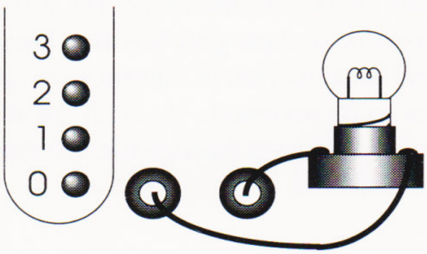
The connection between your output device and Smart Box is made using a length of 2 core cable terminated with 4mm plugs. These plugs are available from most electronic suppliers and come with either solder or screw fixing. Suitable cable is sold as 'figure 8' and is often used for loudspeaker connections.



Motors

Connect your motor to the pairs of motor sockets using 4mm connectors. The motor should be designed to operate at around 5 volts. Each output can supply up to 1.5 Amps but the total load must not exceed 2 Amps. Most small motors require a fraction of this current. If the outputs are overloaded the output voltage will automatically reduce until the load is removed.

Digital outputs



The digital outputs are designed to drive lower power devices than the motor outputs, any one load should not exceed 400 mA with a total loading of 150mA per output if all 8 outputs are on at once. Again for most small devices this is more than adequate. A typical buzzer would take 25mA and a bulb between 40 and 60mA.

SMART MOVE PC

Smart Move - PC

E C O N O M A T I C S

Copyright

Trademarks

Contents

ABORT	ASK
AVERAGE	BACKWARD
CLOSE	CLS
COUNT	END
FALSE	FILE
FOR	FORWARD
GET	GET$
HALT	IF
INKEY	INKEY$
INPORT	LABEL
LET	NEXT
OUTPUT	PARAMETER
PAUSE	POWER
PRINT	PULSE
REM	REPEAT
RESET	RND
RTC	SENSOR
SPEED	STORE
SWITCH	TRACE
TRUE	UNTIL
WAIT	

Mathematical operators

Introduction

About Smart Move

Smart Move is the control program designed for use with Smart Box. With Smart Move you can create your own programs to control models and other devices connected to Smart Box. Smart Move includes special commands to control the motors and other outputs. It also includes commands to detect the signals received from any sensors you have connected to Smart Box. The sensors are the *inputs* to the system. The motors, lamps or other similar devices are the *outputs*. The outputs can be switched on and off depending on the state of the inputs. For example, a lamp can be switched on when a light sensor detects a fall in light level, or a buzzer can be switched on when a temperature sensor detects a predetermined temperature.

Data capture

Smart Move also includes commands which allow you to save readings from the sensors to disk. These sensor readings can then be analysed using other programs such as a spreadsheet. The readings can also be stored when certain conditions are met. For example the temperature during the hours of darkness could be taken by recording the values from the temperature sensor only when the light sensor reading is below a certain level. This feature allows sophisticated data capture experiments to be designed.

Procedures

Each Smart Move program you create will consist of at least one *procedure* . A procedure is a sequence of instructions which are grouped together and given a name. You then simply refer to the procedure name to make your program execute the complete sequence of instructions. A typical Smart Move program will be broken into a number of small procedures, each procedure performing a precise part of the overall program. Breaking the program down into procedures allows each one to be tested independently. Each procedure can be used any number of times within any other procedure. You will usually have one main procedure which uses a number of sub procedures.

Presumed Knowledge

This guide presumes no prior knowledge of Smart Box, Smart Move or control technology. However, the guide does assume that you know how to perform certain basic tasks with your computer. You should know how to:

- Use the keyboard
- Copy disks, programs and documents
- Perform simple editing tasks, including deleting text

If you need to learn more about these basic tasks, consult the handbooks and disks that came with your computer.

System requirements

Smart Move PC requires:
An IBM PC or 100 % compatible computer.
512 K available user memory.
MS-DOS® 3.0 or higher.
A minimum of CGA graphics capability.

Installation

Smart Move is supplied on both 720K 3 1/2" and 360K 5 1/4" disk formats. Before using Smart Move make a backup copy of the master disk.

To use Smart Move PC from a floppy disk:

- Copy all the files from the master disk to a floppy disk formatted with a system.

To use from a network or hard disk based system:

- Copy all the files from the master disk into the appropriate path.

Starting Smart Move

Smart Move is run by typing Smart <return>. Smart Move defaults to using serial port Com 1 for communication with Smart Box. To use Smart Box on Com 2 type Smart 2 <return>. (Note the space between Smart and 2)
Smart Move will not work correctly if your computer loads a driver, for example a mouse driver, onto the same serial port that Smart Move is using. If you find that Smart Move will not load, or stops working when you start to type at the keyboard, check the sytem of your computer.

Tutorial

The following sections of this guide take you through creating your first program, and provide a number of example procedures. The language commands are detailed in full in the reference section.

Starting Smart Move

Make sure that Smart Box and your computer are correctly connected and switched on (see Smart Box guide page 8).

Load Smart Move by typing Smart < Enter > for Com 1 or Smart 2 < Enter > for Com 2. (Note the space between Smart and 2) Once loaded you will see the main Smart Move screen as shown in fig. 1. The top of the screen is the monitor window. This area of the screen displays the current state of the various inputs and outputs.

The bottom section of the screen is the text window. You use this window to type in commands. Any values you ask Smart Move to print will also be displayed here.

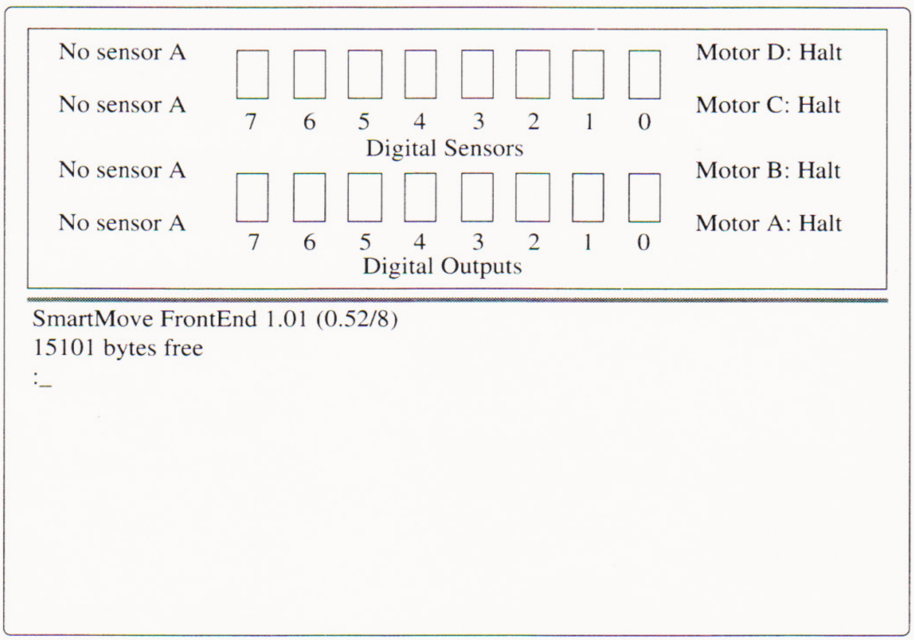


fig. 1

Procedures

You have typed some commands and Smart Move has acted on them immediately. The next stage is to build a sequence of commands to perform a task which Smart Move will carry out. A procedure is a sequence of instructions. Each procedure is given a name, which can be up to 15 characters in length, when you create it.

Follow these instructions to create a procedure called flash which will switch on one of the outputs for one second. Begin by typing BUILD flash < Enter >

The screen will change to display the Smart Move procedure editor as shown in fig. 2.

Type the lines show in fig. 2 into the editor.

You can edit any line by moving the cursor with the cursor keys and inserting characters, or deleting them using the delete key.

When you have entered all the lines press the < Esc > key. This will close the editor and display the main screen. Now type flash and press < Enter >.

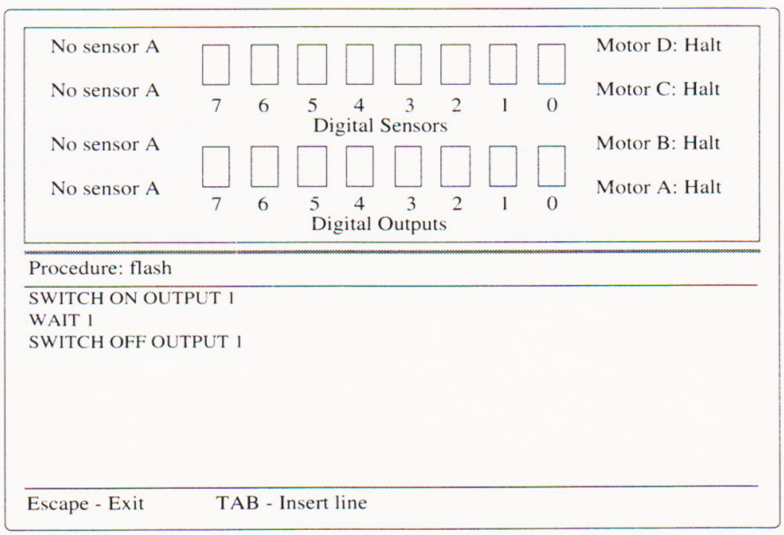


fig. 2

Repeating actions

You have created your first procedure and tested it. Hopefully it switched on output 1 for one second. Procedures can be developed, tested and then developed further. The flash procedure can now be extended to flash 5 times.

Return to the editor by typing EDIT flash and press < Enter >. Now edit the procedure to add the lines shown in fig. 3. Press < tab > to insert a line above the cursor.

When you have finished press < Esc > to close the procedure editor. Now type flash < Enter > to see the result.

```
Procedure: flash
FOR 5 TIMES
SWITCH ON OUTPUT 1
WAIT 1
SWITCH OFF OUTPUT 1
WAIT 1
NEXT

Escape - Exit          TAB - Insert line
```

fig. 3

How it works. The FOR NEXT lines form a loop. Any lines between the FOR and NEXT are repeated, in this case 5 times.

Try making the light flash for 10 times.

Variables

The FOR NEXT loop is a way of making a certain action occur a set number of times. Sometimes it is necessary to change something each time the action is performed.

One way of doing this is to use a *variable.* A variable is a word which can be given a numerical value. Try this example. From the Smart Move main screen type LET cars = 23 < Enter > Then type PRINT cars < Enter > The current value of cars (i.e. 23) will be printed.

The example shown in fig. 4 uses a variable to switch on and off each output in turn. Type BUILD FlashThemAll < Enter > and type in the lines shown in fig. 4. Then press the < Esc > key to display the main screen. Type FlashThemAll < Enter > to run it.

```
Procedure: FlashThemAll
-----------------------------------------
LET number = 0
FOR 8 TIMES
PRINT number
SWITCH ON OUTPUT number
WAIT 1
SWITCH OFF OUTPUT number
WAIT 1
LET number = number + 1
NEXT
-----------------------------------------
Escape - Exit          TAB - Insert line
```

fig. 4

How it works. As you can see variables are an important feature of Smart Move. Your variables can be up to 15 characters in length. The command VALUES will list all the variables you have created and their current value.

Procedures within procedures

When Smart Move sees a procedure name it performs the sequence of instructions that you have created, Smart Move treats your procedure as one of its commands. Type the example procedure lights as shown in fig. 5 into the editor.

```
Procedure: Lights
-----------------
flash
FlashThemAll




-----------------
Escape - Exit        TAB - Insert line
```

fig. 5

Return to the main screen by pressing < Esc >. When you type lights < Enter > the procedure will first flash output 1 five times and then switch each output on in turn.

How it works. When Smart Move performs the lights procedure it first meets the flash command. You have already defined a sequence of instructions known as flash so it performs these. At the next line it meets the FlashThemAll command and so performs that sequence of instructions. As you can see, you are able to use one procedure within another. In this way a problem can be broken down into a number of small procedures which perform specific tasks. They can be tested, edited and checked individually and then put together to form a complete program.

Whenever you create a Smart Move program try to break it down into small procedures in this way.

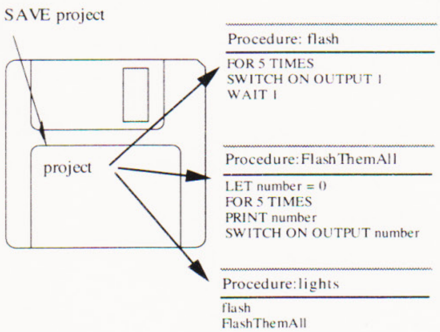
Saving procedures

As you create your procedures they are stored in the memory in Smart Box, but they are lost when the power is switched off. You can save your procedures to the disk on your PC so that they can be recalled at a later date. With Smart Move you can save individual procedures or save all the procedures you create as a group.

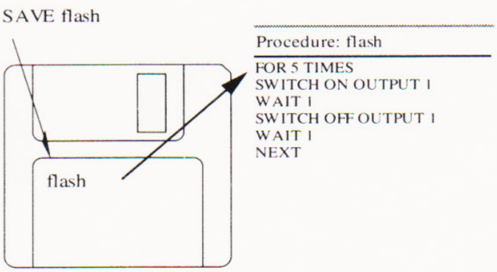
To save all procedures together type SAVE followed by a filename. The filename must be different from any of the procedure names you have used when you are saving a number of procedures together.

To save an individual procedure type SAVE followed by a filename followed by the name of the procedure you wish to save. The file name and procedure name can be the same when you are saving an individual procedure.

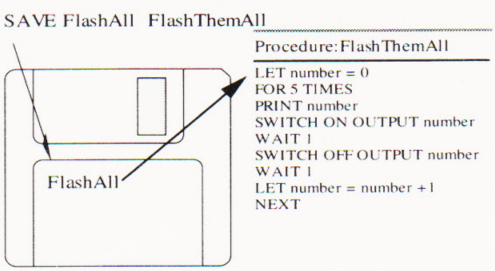
The following example presumes that you have created three procedures: flash, FlashThemAll and lights. (To see a list of the procedures you have created type LIST < Enter >)



SAVE project saves all the procedures in memory under the filename project.



SAVE flash flash saves the procedure flash as flash on the disk.



SAVE FlashAll FlashThemAll saves the procedure FlashThemAll as FlashAll on the disk. The filename is limited to 8 characters so you will have to abbreviate long procedure names in this way when you save them.

Loading procedures

To load a procedure or group of procedures type LOAD followed by the filename. If you are not sure of the filename type LOAD < Enter > this will display a catalogue of the available procedures.

Procedure:FlashThemAll

```
LET number = 0
FOR 5 TIMES
PRINT number
SWITCH ON OUTPUT number
WAIT 1
SWITCH OFF OUTPUT number
WAIT 1
LET number = number +1
NEXT
```

LOAD FlashAll

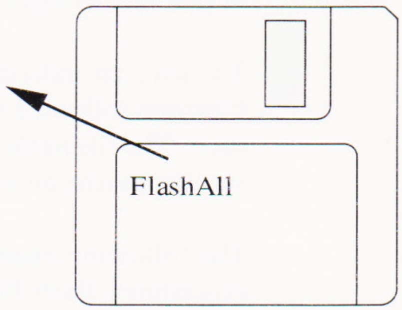


fig. 9

Using Sensors

You can attach various sensors to Smart Box and then write a Smart Move program which uses the reading from a sensor to determine whether or not to carry out a command. The sensors will return a value when you ask Smart Move to check them. The digital sensors will return either a 1 or a 0, the analogue sensors will return a number dependent on the reading from the sensor.

Connect the temperature sensor into sensor socket A. You will notice that the screen display changes to show the sensor name.

Type PRINT Temp < Enter >

A reading from sensor A will be printed in the output window. You can use the sensors in a number of ways. The program can wait for a sensor to change before doing something, or you can choose between two actions depending on the sensor state.

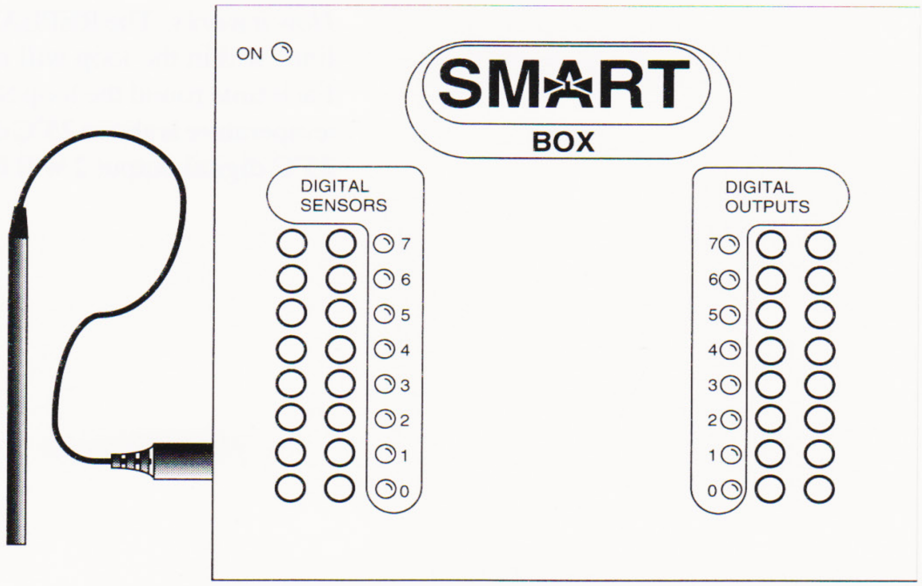


fig. 10

Making decisions

In many cases you will want to make your program choose between one action or another depending on the sensor reading. One example of this is given in fig. 11. This procedure will constantly check the temperature and switch on either one of two lamps depending on the reading.

Procedure: monitor

```
REPEAT
IF TEMP >25 THEN SWITCH ON 1 ELSE SWITCH OFF 1
IF TEMP <25 THEN SWITCH ON 2ELSE SWITCH OFF 2
UNTIL TEMP<10
```

Escape - Exit   TAB - Insert line

fig. 11

How it works. The REPEAT and UNTIL lines form a loop. The lines within the loop will repeat until sensor A falls below 10. Each time round the loop Smart Move checks the sensor. If the temperature is above 25°C digital output 1 will light, if it is below 25°C digital output 2 will light.

Data capture

Smart Move includes the facility to save readings from the sensors to disk. This allows data to be gathered under certain conditions and at various times. The information is saved in a form which can be read by most spreadsheets and word processors. An example procedure for capturing data is shown in fig. 12.

Procedure: collect

```
RESET CLOCK
FILE 1,"Data"
START CLOCK
REPEAT
STORE 1,SENSOR A
WAIT 0.5
UNTIL TIME >:10:00
CLOSE 1
```

Escape - Exit TAB - Insert line

fig. 12

How it works. The first line resets the internal clock to 0. Line two opens a file to save the readings to, in this case the file channel 1 is opened (you can open up to 10 files at once) and the file is given the name 'data'. The repeat until loop then stores the reading from sensor A once every half a second, for ten seconds. The final line closes the disk file. You will be able to open this file with a word processor or spreadsheet to produce reports and graphs. The data is saved on disk as a text file with the file extension .dat Consult the manual of your spreadsheet or word processor to see how to load a text file into your particular program.

Average

The AVERAGE command is useful if you want to look at the changes on a sensor over a longer period of time. For example taking a reading of sound levels during the night could be done by storing the sound level every five minutes, but this will only store the volume at the precise moment the reading is taken. By averaging the readings for every five minute period a better impression of the sound levels during the night will be built up. An example procedure using the AVERAGE command is shown in fig. 13.

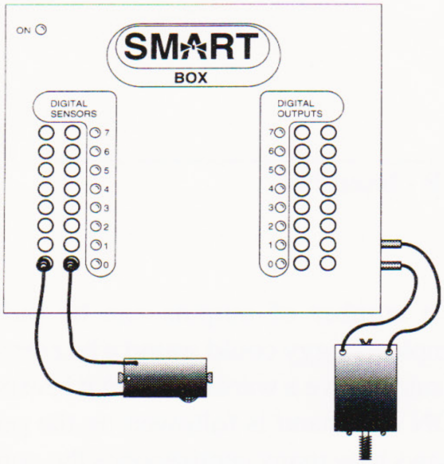
```
Procedure: sample
FILE 1,"data"
FOR 100 TIMES
AVERAGE CLEAR
FOR 10 TIMES
AVERAGE SENSOR a
WAIT 1
NEXT
STORE 1,AVERAGE
NEXT
CLOSE 1
Escape - Exit          TAB - Insert line
```

fig. 13

How it works. The first line of this procedure opens file 1 for storage and gives it the filename "data". The next line sets up a loop which will be completed 100 times. Line three clears the value of average to begin a new set of samples. Line four starts a loop which averages the reading from sensor A over 10 readings, taking a reading once every second. The average value is then stored and the program returns to line two to complete the next sample. Once 100 readings have been stored the disk file is closed.

Controlling a motor

Smart Move inclues special commands for controlling a motor. The example shown in fig. 14 will make the motor run forwards until the microswitch is pressed. The motor will then reverse slowly for 5 seconds before stopping. The motor speed can be varied between 1 and 9 but you may find that some motors do not operate at the lowest speed levels.



Procedure: drive

```
FORWARD a
WAIT UNTIL SENSOR 0 IS ON
BACKWARD a SPEED 6
WAIT 5
HALT a
```

Escape - Exit   TAB - Insert line

fig. 14

Varying Power and Pulsing Outputs

You can also vary the power on the digital outputs and set up pulse sequences as well. A pulsed output could be used to switch a buzzer on and off continuously to act as an alarm. The example in fig. 15 uses the POWER command to slowly increase the brightness of a bulb connected to digital output 0.

```
Procedure: bright
LET a=1
FOR 8 TIMES
SWITCH ON 0 POWER a
WAIT 1
LET a = a+1
NEXT

Escape - Exit          TAB - Insert line
```

fig.15

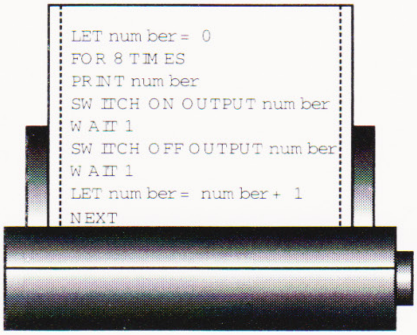
Figure 16 shows how a number of outputs can be pulsed independently. For example a buggy could sound a buzzer and flash lights at a different rate to give a warning when it reverses. The normal SWITCH ON command is followed by the pulse period, the first figure shows how many centiseconds the output will be on for, the second number is the off period. The maximum value in either case is 255 milliseconds.

```
Procedure: pulsing
SWITCH ON 0 PULSE 50,50
SWITCH ON 1 PULSE 50,255
SWITCH ON 2 PULSE 255,50
SWITCH ON 3 PULSE 255,255
SWITCH ON 4 PULSE 10,10
SWITCH ON 5 PULSE 125,125
PRINT "Press the space bar to stop"
WAIT UNTIL GET = 32
SWITCH OFF 0,1,2,3,4,5
Escape - Exit          TAB - Insert line
```

fig.16

Printing procedures

To print your procedures first check that your printer is on line and ready. Type LIST followed by the procedure name. The procedure will be printed to your printer.



This section has introduced the basic principles involved in creating a Smart Move program. The reference section gives full details of each command. Whenever you write a Smart Move program try to create a number of procedures, each doing a specific task, rather than one large program doing everything. In this way you will be able to test the individual parts of your program and identify the errors more easily.

Reference

The words which Smart Move understands are the 'Keywords' Some are 'Commands', ie. they perform some action. Others are 'Functions' ie. they will return a value.
Some of the keywords can be typed in only from the main screen. They cannot be used within procedures. These are known as the direct commands.
The direct commands are listed first, with the remaining keywords described in full in alphabetical order, beginning with ABORT on page 3-4.

BUILD procname

Used to create a new procedure. The procedure name can be up to 15 characters in length and include any letter of the alphabet, numbers and the underline character. Procedure names cannot begin with a number.

CMDS

Displays a list of the commands which can be used within procedures.

COMMANDS

Displays a list of the commands which can only be issued as a direct command, ie not used within a procedure.

DELETE proc

This is used to delete a procedure in memory. Type DELETE followed by the procedure name.

EDIT proc

Edit a procedure in memory. Typing EDIT without a procedure name will open the editor with the most recently modified procedure.

FNS

Display a list of available functions.

LBLS,LBLS ON, LBLS OFF

You can rename the sensors and some keywords using the LABEL command. Typing LBLS displays a list of the current labels. It is sometimes useful to print the original name when using the trace facility or when in the editor. This can be achieved by typing LBLS OFF. The labels can be restored with LBLS ON.

LIST

Display a list of the procedures in memory.

LIST *proc*

Sends a listing of the procedure *proc* to the current printer.

LOAD

Load procedure(s) from disk. Type LOAD without a procedure name to show a list of procedures on the disk. Type LOAD followed by a filename name to load a specific file. A list showing which procedures have been loaded will be displayed. See SAVE for details of how to save a procedure or group of procedures.

NEW

Clear all procedures and variables from memory.

QUIT

Exit the Smart Move software.

RENAME *oldprocname newprocname*

Give a procedure a new name. Type the current procedure name, then a space, then the new name for that procedure.

SAVE *filename*

Save all procedures or an individual procedure to disk. Typing SAVE with a filename will save all procedures currently in memory to the file. To save an individual procedure type SAVE *filename procedurename*. The filename is limited to 8 characters under MS DOS.

VALUES

The VALUES command will display the current values of any variables you have used.

ABORT

Syntax
ABORT

Description
This command interrupts a program which is running and prints the name of the procedure which was executing at the time. This can be used to finish a program when a certain condition occurs. This differs from the END command in that it ends all currently running procedures and returns to command mode. END will return to the calling procedure.

Examples

```
PRINT "Press S to stop!"

IF INKEY$ ="S" THEN ABORT
```

Associated keyword
END

ASK

Syntax

ASK S/N/T,"prompt string",variable

Description

This command is used to prompt user for a keyboard input. The user can type a string or number. The ASK command is followed by either S to indicate a string ,N to indicate a numeric value or T for a time value. The prompt string can be up to 70 characters in length. The response from the user is placed into the variable specified at the end of the command.

Example

```
ASK N,"How long should the lamp be on for ?",reply
SWITCH ON OUTPUT 1
WAIT reply
```

Associated keywords

INKEY, INKEY$, GET, GET$

AVERAGE

Syntax
AVERAGE

Description
The average function returns the average of the numbers passed to it since the last AVERAGE CLEAR command.

Example
Clear the average accumulator

```
AVERAGE CLEAR
```

Take the average of SENSOR A over 10 readings

```
FOR 10 TIMES
AVERAGE SENSOR A
NEXT
PRINT AVERAGE
```

Associated keywords
STORE

BACKWARD

Syntax
BACKWARD A/B/C/D

Description
Turn a motor backwards. If the motor turns in the wrong direction when you use the BACKWARD command reverse the plugs at the motor outputs. The command also resets the value of COUNT for the appropriate digital input to zero.

Example
Procedure to move a buggy backwards for 3 seconds

```
BACKWARD A,B
WAIT 3
HALT A,B
```

Associated keywords
FORWARD, HALT, SPEED, POWER, PULSE, COUNT

CLOSE

Syntax
CLOSE channel

Description
Close a disk file which has been opened for data capture. To close all open channels type CLOSE without a channel number.

Example
To close file 1

```
CLOSE 1
```

To close all open files

```
CLOSE
```

Associated keywords
FILE, STORE

CLS

Syntax
CLS

Description
This command clears all text in the text window and moves the cursor to the start of the first line.

Example

```
CLS
```

Associated keywords
PRINT

COUNT

Syntax
COUNT input

Description
The count function returns the number of pulses counted on a digital sensor. The value of COUNT can be reset by the command COUNT CLEAR. A special feature of the motor commands also clears the value of COUNT. Any command issued to Motor A will reset COUNT 0, Motor B will reset COUNT 1 etc. The count function will normally return an integer showing the number of times a sensor, for example a switch, has closed. Every time the switch changes state and goes back again will register as 1 count, ie. from off to on and off again. The count function can also be used to record the number of changes of *state*, ie. the number of times the sensor has changed from either on to off or off to on. Every transition registers as 0.5

Examples
Reset counter on INPUT 0 to zero

```
COUNT CLEAR 0
```

Reset all counters to zero

```
COUNT CLEAR
```

Switch on a motor until INPUT 0 registers 10 counts.

```
COUNT CLEAR
FORWARD A
WAIT UNTIL COUNT 0 >=10
```

Associated keywords
SENSOR, INPORT

END

Syntax
END

Description
This is used to make Smart Move end the current procedure. END is optional but may be used as many times as required in a procedure. See also ABORT.

Example

```
IF SENSOR A > 128 THEN END
```

Associated keywords
ABORT

FALSE

Syntax
FALSE

Description
FALSE is represented by the value 0 in Smart Move.

Example

```
PRINT FALSE
```

Associated keywords
TRUE

FILE

Syntax
FILE channelnumber,"filename"

Description
This command opens a disk file to store sensor readings. Before you can save data from sensors you must open a file The channel number can be in the range 1 to 10. A number of channels can be opened simultaneously.

Example

```
FILE 1,"Data"
```

Associated keywords
CLOSE, STORE

FOR

Syntax

FOR number TIMES

Description

The command FOR is used to begin a FOR...NEXT loop. This makes the computer execute a series of lines a specified number of times. The word TIMES is optional. FOR ... NEXT loops can be nested up to 40 deep. The maximum loop value is 255.

Example

To print the value of sensor a 5 times at 1 second intervals.

```
FOR 5 TIMES
PRINT SENSOR A
WAIT 1
NEXT
```

Associated keywords

NEXT

FORWARD

Syntax
FORWARD A/B/C/D POWER/SPEED/PULSE

Description
Turn a motor forward. If the motor turns in the wrong direction when you use the FORWARD command reverse the plugs at the motor outputs. The FORWARD command also resets the appropriate COUNT value.

Examples
Switch on motor 'A'

    FORWARD A

Switch on motors 'A','B',' and 'C'

    FORWARD A,B,C

Switch on motor 'A' at speed 6 (See the SPEED command for more details of controlling speed.)

    FORWARD A SPEED 6

Continuously pulse motor 'A' on for 2.5 seconds and off for 2.5 seconds (see the PULSE command for details of pulsing outputs)

    FORWARD A PULSE 250,250

Associated keywords
SPEED, POWER, PULSE, COUNT

GET, GET$

Syntax
GET($)

Description
These functions wait for a key to be pressed on the keyboard and then return the key as either ASCII (GET) or the character (GET$) The ASCII value is the numeric value associated with each key by the computer.

Example
Switch on motor 'A' if the 'F' key is pressed

```
IF GET$ = "F" THEN FORWARD A
```

Associated keywords
INKEY, INKEY$

HALT

Syntax
HALT MOTOR A/B/C/D

Description
Used to switch off any motor output. Also resets the value of COUNT for the approriate motor.

Example
Stop motor 'A'

    HALT A

Stop motors 'A','B' and 'C'

    HALT A,B,C

Associated keywords
FORWARD, BACKWARD, COUNT

IF

Syntax

IF condition THEN action ELSE action

Description

The IF command is used to test for a certain condition and then perform an action. The command can be extended using the word ELSE to perform one action if the condition is true or another if the condition is false.

Examples

IF SENSOR 1 IS ON THEN FORWARD A

IF SENSOR A IS GREATER THAN 128 THEN SWITCH ON 1 ELSE SWITCH OFF 1

Associated keywords

THEN, ELSE

INKEY, INKEY$

Syntax

INKEY($)

Description

These functions return the value of any key which is pressed down when the INKEY function is used, or has been pressed since the previous INKEY/INKEY$. The key is returned as either its ASCII value (INKEY) or its character (INKEY$). INKEY returns -1 if no key is pressed. The ASCII value is the numeric value associated with each key by the computer. INKEY$ returns a blank string if no key has been pressed.

Examples

Print SENSOR A until the space-bar is pressed

```
REPEAT
PRINT SENSOR A
UNTIL INKEY = 32
```

Move a buggy forward when the 'F' key is pressed

```
REPEAT
IF INKEY$ = "F" THEN FORWARD A ELSE
HALT A
FOREVER
```

Associated keywords

GET, GET$

INPORT

Syntax
INPORT

Description
The function INPORT returns the current 'value' of all digital sensors as a single number.
Each digital sensor has a unique 'value' as shown below:

SENSOR 0 = 1 when it is on.
SENSOR 1 = 2
SENSOR 2 = 4
SENSOR 3 = 8
SENSOR 4 = 16
SENSOR 5 = 32
SENSOR 6 = 64
SENSOR 7 = 128

So if SENSORS 0,3 and 5 are 'ON' INPORT will return 41 (1+8+32)

Example

```
IF INPORT = 255 THEN PRINT "All sensors are on"
```

Associated keywords
SENSOR

LABEL

Syntax LABEL oldname newname

Description

This command allows you to rename the sensors and some of the command words. Labels are saved when a complete project is saved to disk. The new label can be up to 9 characters in length. Sensors in the Smart Sense range will automatically label the appropriate sensor input when they are plugged in.

The following words can be changed.

BACKWARD	FORWARD	HALT
OUTPUT	SWITCHON	SWITCHOFF
SWITCH	INPORT	SENSOR0
SENSOR1	SENSOR2	SENSOR3
SENSOR4	SENSOR5	SENSOR6
SENSOR7	SENSORA	SENSORB
SENSORC	SENSORD	SENSOR
POWER	PULSE	SPEED
OFF	ON	MOTORA
MOTORB	MOTORC	MOTORD
OUTPUT0	OUTPUT1	OUTPUT2
OUTPUT3	OUTPUT4	OUTPUT5
OUTPUT6	OUTPUT7	

Examples

LABEL FORWARD UP

LABEL OUTPUT0 BUZZER

Associated keywords

LBLS, LBLS ON, LBLS OFF

LET

Syntax

LET variable = value

Description

The command LET is used to assign a value to a variable. Variables are used as a temporary place to store numbers, strings or time values. Variables can be up to 15 characters in length.

Example

Give the variable 'x' the value 10

```
LET x = 10
```

Add 1 to the current value of the variable 'x' and put the result into the variable 'turns'

```
LET turns = x+1
```

Put the time value 10 seconds into the variable 'MyTime'

```
LET MyTime = :10:0
```

Put a string of text into the variable 'MyText'. This can be useful where a long string of text is needed at a number of points in a program. Rather than typing the string in every time the variable can be printed.

```
LET MyText = "This is a message"
```

Associated keywords

PARAMETER

NEXT

Syntax
NEXT

Description
The NEXT keyword is used with FOR to create a loop which will be executed a number of times.

Example

```
FOR 10 TIMES
SWITCH ON 1
WAIT 1
SWITCH OFF 1
WAIT 1
NEXT
```

Associated keyword
FOR

OUTPUT

Syntax
OUTPUT value

Description
This command can be used to switch on and off the digital outputs simultaneously. Each digital output has a unique number associated with it. By adding up the values for any outputs you want to switch on you can set some of the outputs on while switching off any other outputs which may have been on.

Example procedure to switch on the outputs in a sequence.

```
LET A = 1
REPEAT
PRINT A
OUTPUT A
LET A = A+A
UNTIL A = 256
```

Associated keywords
SWITCH ON, SWITCH OFF

PARAMETER

Syntax
PARAMETER variable,variable,variable...

Description
The PARAMETER command assigns values to variables in a procedure. Any number of values can be 'passed' to the procedure in this way. It is like performing a number of LET commands at once.

Example
First type in this procedure.

```
PARAMETER number1,number2,number3
PRINT "Add together ";
PRINT number1;"+";number2;"+";number3
PRINT "= ";
PRINT number1+number2+number3
```

Now from the main screen call the procedure by typing

sum 1,2,3 <Enter>

The screen will display

```
Add together 1+2+3
= 6
```

Now try calling the procedure again using different numbers.

Associated keywords
LET

PAUSE

Syntax
PAUSE seconds.centiseconds

Description
The PAUSE command causes all processing to cease for a certain time. It is the same as using the WAIT command with a time period.

Example

```
PAUSE 10
```

Associated keywords
WAIT

POWER

Syntax
POWER powerlevel

Description
The POWER command varies the power from the outputs by rapidly switching them on and off. The powerlevel can be between 1 and 9, A powerlevel of 1 will switch the output on for 100 centiseconds and off for 900. A powerlevel of 9 will switch the output on for 900 centiseconds and off for 100.

Example

```
SWITCH ON 1 POWER 5
```

Associated keywords
PULSE, SPEED

PULSE

Syntax
PULSE onperiod,offperiod

Description
The PULSE command allows an output to be switched on and off continuously with varying pulse rates. The on and off periods can be individualy set to a maximum value of 255 which is the equivalent of 2.55 seconds.

Example

```
SWITCH ON 1 PULSE 50,50
SWITCH ON 1 PULSE 10,250
```

Associated keywords
POWER, SPEED

PRINT

Syntax
PRINT printlist

Description
The PRINT command is used to print a text string or variable to the output window. The command is followed by a list of items to print. Anything within inverted commas will be printed as it is. Anything not within inverted commas is assumed to be a variable and the current contents of the variable will be printed. A semi-colon after an item in the list will print the next item on the same line.An apostrophe between items in the list forces the following item to be printed on a new line.

Example
PRINT "This is a text string"

produces

This is a text string

PRINT "The value of INPUT 0 is ";INPUT 0

produces

The value of INPUT 0 is 0

PRINT SENSOR A'SENSOR B'SENSOR C'SENSOR D

produces

28
28
28
28

Associated keywords
CLS

REM

Syntax
REM anything

Description
The REM keyword allows you to add comments to your program as a reminder of the purpose of each part of the program. Any text after the REM will be ignored by Smart Move.

Example

```
REM This procedure switches on the alarm
```

Associated keywords
None

REPEAT

Syntax
REPEAT

Description
The REPEAT ... UNTIL keywords form a loop Any instructions within the loop will be repeated until a certain condition occurs. The condition can include multiple conditions with the AND/OR keywords. The loop can be made to continue forever, or until the Escape key is pressed, by using REPEAT ... FOREVER.
REPEAT loops can be 'nested' up to 60 deep.

Example

```
REPEAT
PRINT SENSOR A
UNTIL SENSOR A IS > 100 OR SENSOR B < 50
```

Associated keywords
UNTIL

RESET

Syntax
RESET CLOCK

Description
Smart Box has an internal clock which ticks 100 times per second. The RESET command is used to set the clock to zero.

Example

```
RESET CLOCK
```

Associated keywords
START, STOP

RND

Syntax
RND

Description
This function returns a random number between 0.00 and 65535.99

Example

```
PRINT RND
```

Associated keywords
None

RTC

Syntax
RTC

Description
This function returns the time from the host micro's clock in Smart Move time format.

Example
To print the value of the host micro's clock

```
PRINT RTC
```

To set the clock in Smart Move to the same time as the host micro's.

```
LET TIME = RTC
```

Associated keywords
TIME

SENSOR

Syntax
SENSOR sensorname

Description
This function returns the current value of the specified sensor.

Examples

```
PRINT SENSOR A

PRINT LIGHT
```

Associated keywords
INPORT

SPEED

Syntax
SPEED speedvalue

Description
The SPEED command varies the speed of devices connected to the outputs by rapidly switching them on and off. The speedvalue can be between 1 and 9, A speedvalue of 1 will switch the output on for 100 centiseconds and off for 900. A speedvalue of 9 will switch the output on for 900 centiseconds and off for 100.

Example

```
FORWARD A SPEED 5
```

Associated keywords
PULSE, POWER

STORE

Syntax
STORE channel,variable,variable,variable

Description
The STORE command is used to store variables to a disk file. This is typically used to store readings from a sensor, or sensors for data capture purposes. The command can store numeric variables with a value between 0 and 255, variables containing text strings and time values. each variable is separated in the file with a TAB character. The time is automatically stored at the beginning of each STORE statement.

Example

```
STORE 1,SENSOR A,SENSOR B
```

Associated keyword
FILE

SWITCH

Syntax

SWITCH ON/OFF

Description

Used to switch on and off the digital outputs. Smart Move will also accept SWITCHON and SWITCHOFF as valid keywords.

Examples

SWITCH ON 1
SWITCH ON OUTPUT 1
SWITCH ON 0,2,4,6
SWITCH ON 3 POWER 5
SWITCH ON 5 PULSE 50,50
SWITCH ON 1 SPEED 5

Associated keywords

POWER, SPEED, PULSE

TIME

Syntax
TIME Hours:Minutes:Seconds:Centiseconds

Description
The TIME keyword can be used either to set the internal clock to a certain time or to read the current time. A Time value begins with a colon. When setting the time include all values from the unit you are setting down to the centiseconds for example. To set the clock to a value of 1 minute type :1:00:00 (1 minute, zero seconds, zero centiseconds)

Examples
Print the current time

    PRINT TIME

Set the time to 10 seconds

    LET TIME = :10:0

Associated keywords
START, RESET, RTC

TRACE

Syntax
TRACE ON/OFF

Description
With TRACE ON all lines are printed to the output window as they are executed. This also has the effect of slowing the program down.

Examples

```
TRACE ON

TRACE OFF
```

Associated keywords
None

TRUE

Syntax TRUE

Description
TRUE is represented by the value -1 in Smart Move.

Examples

```
PRINT TRUE
```

Associated keywords
FALSE

UNTIL

Syntax UNTIL condition

Description
UNTIL can be used in two ways. The first, as part of the REPEAT ... UNTIL loop, causes a sequence of lines to be repeated until a condition becomes true. The second, as part of the WAIT .. UNTIL statement, halts all processing until a condition becomes true.

Examples

```
UNTIL INPUT 0 IS ON AND INPUT 1 IS ON
```

Associated keywords
REPEAT, WAIT

WAIT

Syntax
WAIT time/condition

Description

The wait command suspends processing until a certain event occurs or for a specified time period.

Examples

```
WAIT UNTIL SENSOR A >=  50

WAIT 5
```

Associated keyword
PAUSE

Mathematical operators

Description

Smart Move allows the use of mathematical operators using both standard notation and the plain English equivalent. The following operators are allowed:

+	PLUS
-	MINUS
/	DIVIDED BY
*	MULTIPLIED BY
=	EQUALS
<	IS LESS THAN
>	IS GREATER THAN
=	IS EQUAL TO
<>	IS NOT EQUAL TO
= 1	IS ON
= 0	IS OFF
	AND
	OR

Examples

```
PRINT 4+4
PRINT 4 PLUS 4

IF SENSOR 0 IS ON THEN SWITCH ON 1
```

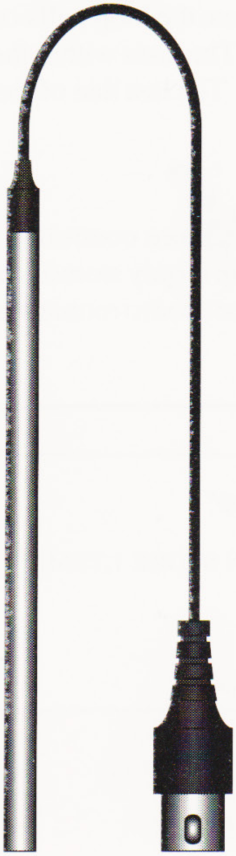
SMART SENSORS

Temperature sensor

The Smart Sense temperature sensor measures temperatures between 2°C and 100°C. It is mounted in a stainless steel tube so that it can be immersed in most types of liquid.

The sensor connects to one of the analogue sensor inputs Once the sensor is connected the Smart Move software will display the name of the sensor and label the appropriate input 'Temp'. The screen will also display the current temperature. Try holding the end of the sensor and watch the reading change.

Using the sensor to control an output



There may be times when you want to turn on or off one of the outputs depending on the reading from the temperature sensor. One example may be to switch on a warning buzzer if the temperature of a liquid falls below a certain level. The procedure shown in fig. 1 will turn on output 1 when the temperature falls below 15°C.

```
Procedure: Warning

REPEAT
IF TEMP IS LESS THAN 15 THEN SWITCH ON 1 ELSE SWITCH OFF
FOREVER
```

Fig. 1

How it works

The second line of this procedure takes a reading from the sensor and either switches on or off output 1 depending on the current value. In order to constantly check the sensor the REPEAT ... FOREVER loop repeats this statement.

Data capture

The readings from the temperature sensor can be stored on disk. The procedure in Fig. 2 will store the temperature once per second for 3 minutes.

Procedure: Cooling

```
RESET CLOCK
START CLOCK
FILE 1,"Cool"
REPEAT
STORE 1, TEMP
WAIT 1
UNTIL TIME IS GREATER THAN 3:0:0
CLOSE 1
```

Fig. 2

The first two lines reset and start the clock. The next line opens the file to save the reading to. Line four is the beginning of a REPEAT … UNTIL loop. This time the loop will continue until the clock has reached 3 minutes. The lines within the loop store the temperature once per second. The last line of the procedure closes the disk file.

Controlling data capture

Fig. 3 gives an example of data capture controlled by another sensor. In this case the temperature is only stored when the light level drops below 50. This could be used to monitor temperature and light during the night.

Procedure: Night

```
FILE 1,"Samples"
PRINT "Logging - Press space to stop"
REPEAT
IF LIGHT IS LESS THAN 50 STORE 1, TEMP
WAIT 60
UNTIL INKEY = 32
CLOSE 1
```

Fig. 3

Light sensor

The Smart Sense light sensor will measure light levels over a wide range. The sensor connects to any one of the analogue sensor inputs. Once connected the Smart Move software will display the name of the sensor and label the appropriate input 'LIGHT'. The screen will display the current light level as a value on a scale from 0, very dark to 255, very bright.

Responding to the sensor

The sensor can be used as an input device to a system which responds to changes in light level. An example might be an automatic lighting system which switches on when the light level falls below a certain value. The procedure shown in fig.1 will switch on a bulb connected to output 0 when the light level falls below 50.

Procedure: Auto

```
REPEAT
IF LIGHT IS LESS THAN 50 THEN SWITCH ON 0 ELSE SWITCH OF
FOREVER
```

Fig.1

How it works

The REPEAT ... FOREVER loop repeats the second line of the procedure. This takes a reading from the sensor and either switches on or off output 0 depending on the current light level.

Counting

The second example (Fig 2.) uses the light sensor as a counter to detect whenever a light beam is broken. As supplied, the sensor will detect light falling on it from a wide angle. For this type of use the sensor will perform more accurately if the sensor is mounted at the end of a tube with a light source, for example a torch, pointing at it from some distance away.

```
Procedure: Counter

number = 0
PRINT " Press the space bar to stop counting"
REPEAT
WAIT UNTIL LIGHT IS LESS THAN 50
LET  number = number +1
WAIT UNTIL LIGHT IS GREATER THAN 75
UNTIL INKEY=32
PRINT number
```

Fig. 2

Recording light levels

The readings from the light sensor can be stored to a disk file. This data can then be analysed using a spreadsheet. Figure 3. records the average light level every 5 minutes during the night. The recording starts when the light falls below a certain level at dusk, 150 in this example, and stops as the sun rises and the sensor reading rises above 200.

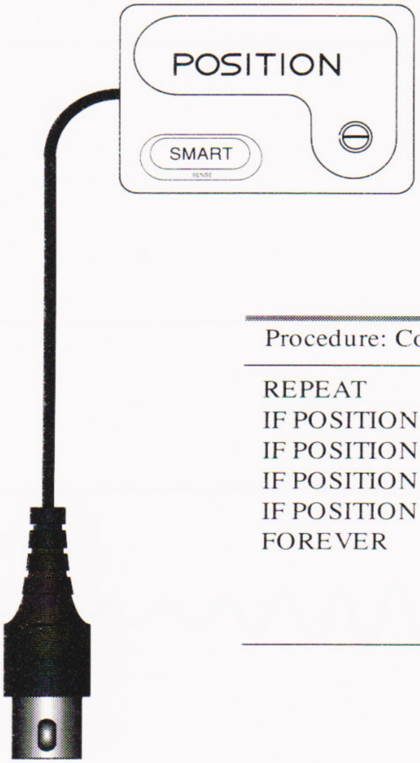
```
Procedure: Night

FILE 1,"Data"
WAIT UNTIL LIGHT IS LESS THAN 150
START CLOCK
REPEAT
RESET CLOCK
AVERAGE CLEAR
REPEAT
AVERAGE LIGHT
UNTIL TIME IS GREATER THAN :5:0:00
STORE 1,AVERAGE
UNTIL LIGHT IS GREATER THAN  200
CLOSE 1
```

Fig. 3

Position sensor

The Smart Sense position sensor will indicate rotational movement over a 360°range. The sensor connects to any one of the analogue sensor inputs. Once connected the Smart move software will display the name of the sensor and label the appropriate input 'Position'. The screen will display the current reading from the sensor, a complete rotation of the spindle will cover the range from 0 to 255. The reading can be converted to degrees by multiplying by 1.41. i.e.. PRINT POSITION * 1.41



The sensor can be used as an input device to a system which responds to changes in position. The example in fig 1 uses the sensor to produce an electronic 'compass'. One of the outputs will switch on as the sensor moves through each quarter of a turn.

Procedure: Compass

```
REPEAT
IF POSITION >= 0 AND POSITION <=63 THEN SWITCH ON 0 ELSE SWITCH OFF 0
IF POSITION >= 64 AND POSITION <=127 THEN SWITCH ON 1 ELSE SWITCH OFF 1
IF POSITION >= 128 AND POSITION <=191 THEN SWITCH ON 2 ELSE SWITCH OFF 2
IF POSITION >= 192 AND POSITION <=255 THEN SWITCH ON 3 ELSE SWITCH OFF 3
FOREVER
```

Fig.1

How it works

The first and last lines of the procedure form a loop which continues until the user escapes from the procedure. Each line within the loop checks if the sensor reading is within a certain range and switches on or off the appropriate output.

Measuring rotation

The second example uses the sensor to gather data as a pendulum swings. The procedure in fig 2 will wait until the user presses the space bar. At this point the pendulum should be released and the readings from the sensor are stored to a disk file. Once the pendulum has come to rest the space bar is pressed and the readings are saved to disk. This file can then be used with a spreadsheet to produce results as in fig 3. This example compares results from two pendulums of different length.

```
Procedure: Pendulum

FILE 1,"data"
WAIT UNTIL GET = 32
PRINT "sampling"
REPEAT
STORE 1,POSITION
UNTIL INKEY = 32
CLOSE 1
```

Fig. 2

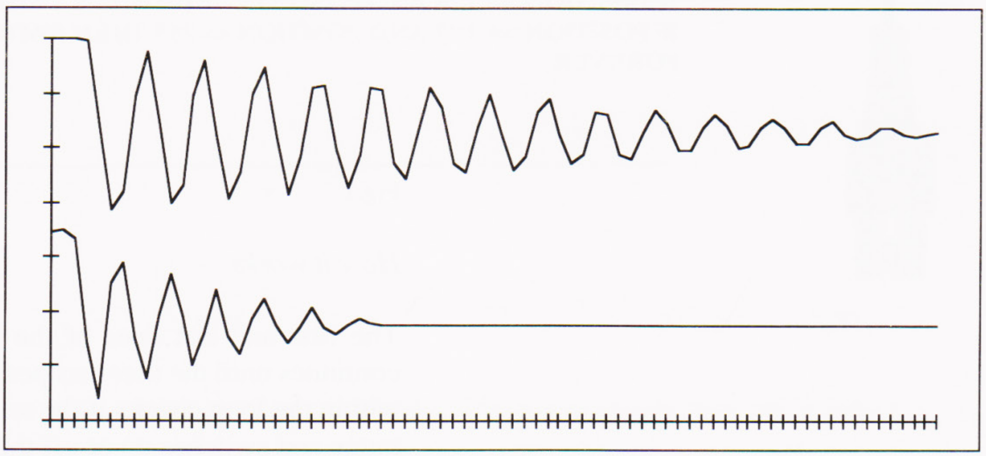


Fig. 3

Sound sensor

The Smart Sense Sound sensor will measure sound levels on one of four ranges. The sensor connects to any one of the analogue sensor inputs. Once connected the Smart Move software will display the name of the sensor and label the appropriate input 'Sound'. The sensitivity of the sensor is reduced as the knob is rotated clockwise

Responding to the sensor

The sensor can be used as an input device to a system which responds to changes in sound level. The example in fig 1 uses the sensor to produce a sound 'meter'. The eight digital outputs are used to indicate sound level. As the volume increases more of the outputs are switched on.


```
Procedure: Soundmeter

REPEAT
IF SOUND < 255 THEN SWITCH OFF 7 ELSE SWITCH ON 7
IF SOUND < 224 THEN SWITCH OFF 6 ELSE SWITCH ON 6
IF SOUND < 192 THEN SWITCH OFF 5 ELSE SWITCH ON 5
IF SOUND < 160 THEN SWITCH OFF 4 ELSE SWITCH ON 4
IF SOUND < 128 THEN SWITCH OFF 3 ELSE SWITCH ON 3
IF SOUND < 64 THEN SWITCH OFF 2 ELSE SWITCH ON 2
IF SOUND < 32 THEN SWITCH OFF 1 ELSE SWITCH ON 1
IF SOUND < 1 THEN SWITCH OFF 0 ELSE SWITCH ON 0
FOREVER
```

Fig.1

How it works

The first and last lines of the procedure form a loop which continues until the user escapes from the procedure. Each line within the loop checks if the sensor reading is below a certain sound level in which case the output is switched off, otherwise it is switched on.

Recording sound levels

The readings from the sound sensor can be stored to disk for analysis with a spreadsheet or graphing program. Fig. 2 shows an example where recording begins as soon as the sound level rises above 20. The recording will stop when it falls below 10. Fig. 3 shows a typical graph produced using this procedure.

Procedure: Recorder

```
FILE 1,"sounds"
WAIT UNTIL SOUND IS GREATER THAN 20
REPEAT
STORE 1,SOUND
UNTIL SOUND IS LESS THAN 10
CLOSE 1
```

Fig. 2

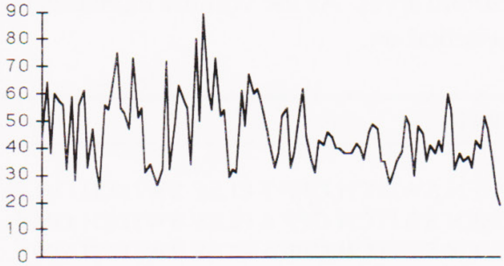


Fig. 3

When sampling in this way about seven samples per second will be stored. If logging is to be performed over a long period, for example recording the level of sound during the night, very large files will be produced. Samples could be taken less frequently, for example once every minute, but as each reading is taken in a fraction of a second any change in sound level between samples will be missed.

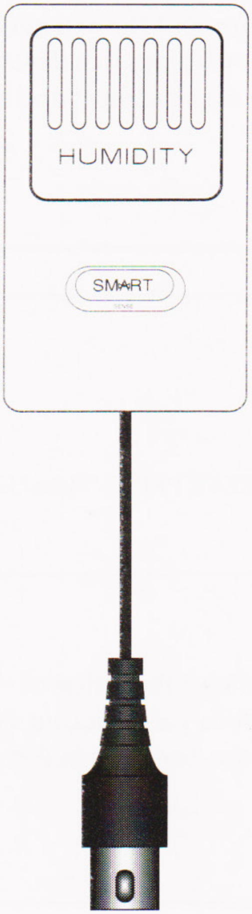
To avoid this problem you can take readings at more frequent intervals and average the result. Fig 4. shows an example using the average command. Readings are taken over a ten second period and the average value is then stored. This routine is repeated 60 times to give an overall logging time of ten minutes.

Procedure: LongTime

```
FILE 1,"data"
START CLOCK
FOR 60 TIMES
RESET CLOCK
AVERAGE CLEAR
REPEAT
AVERAGE SOUND
UNTIL TIME IS GREATER THAN :10:00
STORE 1, AVERAGE
NEXT
CLOSE 1
```

Fig. 4

Humidity sensor



The Smart Sense Humidity sensor is designed to indicate humidity between 10% to 90% relative humidity (R.H.). The sensor connects to any one of the analogue sensor inputs. Once connected the Smart Move software will display the name of the sensor and label the appropriate input 'HUMIDITY'. The screen will display the current reading from the sensor.

The graph (Fig.1) shows the sensor response for a given R.H. The sensor will return readings between 0 and 255 for a change in humidity between 0% and 100% R.H but will not produce accurate readings outside the range 10% to 90% R.H. The shaded band indicates that the sensor will follow the actual humidity to within 5% R.H. The reading from the sensor can be converted to R.H. by dividing by 2.55. The Smart Move INT function can be used to indicate the R.H. to the nearest whole number i.e. PRINT INT (HUMIDITY / 2.55) The sensor will typically respond to changes in humidity within 3 minutes. The sensor is designed to be used between 5°C to 40°C
As with most humidity sensors variations in temperature will produce errors so changes in humidity should be measured at the same temperature whenever possible. Errors of up to 5% R.H. may be caused by variations in temperature between 5- 40°C.

Warning

Do not expose to temperatures outside the sensor's working range.
Do not expose the sensor to organic solvents or vapours.
Do not immerse in any liquid
Do not breathe directly onto the sensor or allow condensation to form on the sensor.

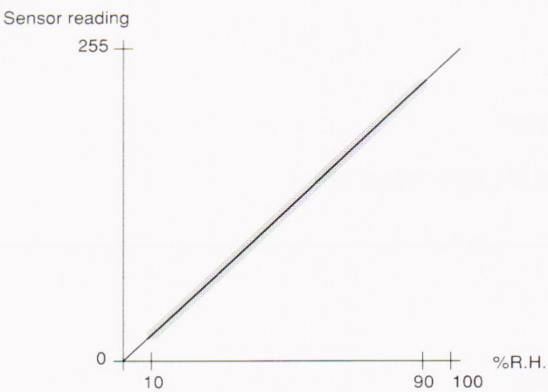


Fig. 1 Working range 10-90% R.H.

Example procedures

As a room fills with people the temperature and humidity can change dramatically. The procedure in figure 2 monitors the room once every minute and prints the current temperature and humidity values. This procedure could be extended by using the STORE command to store the readings to a disk file. The data file could then be loaded into a spreadsheet and a graph of the changes produced.

Procedure: Humid

```
REPEAT
LET a = HUMIDITY
PRINT  "Humidity is";a/2.55; "R. H."
PRINT  "temperature is"; TEMP; "deg C"
WAIT 60
LET b =  HUMIDITY
IF b>a THEN PRINT "Humidity is rising" ELSE PRINT "Humidity is falling"
FOREVER
```

Fig. 2

The second example (Fig. 3) allows the action of silica gel to be observed. Place a sachet of silica gel desiccant and the sensor into a polythene bag. Seal the bag just before running the procedure.

Procedure: Drying

```
RESET CLOCK
START CLOCK
FILE 1, "Drying"
REPEAT
STORE 1, HUMIDITY
WAIT 10
UNTIL HUMIDITYIS LESS THAN 25
CLOSE 1
```

Fig. 3

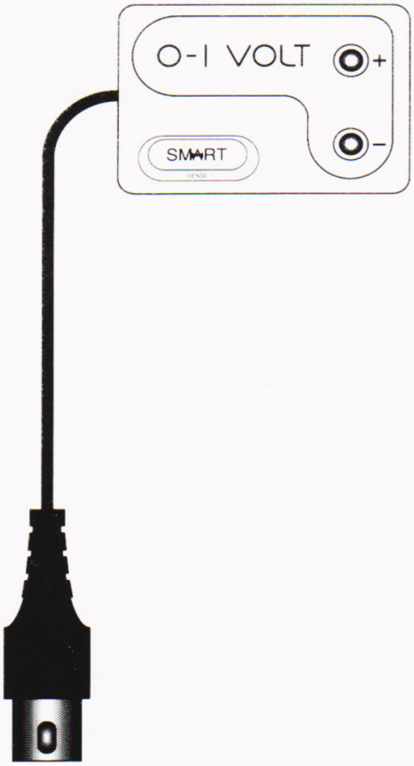
0 - 1 Volt adaptor

The 0-1 Volt adaptor allows sensors which produce an output ranging between 0 and 1 Volt to be used with Smart Box. The adaptor amplifies the signal to give the normal Smart Box input range of between 0 and 2.55 volts.

Connection to the adaptor should be made using standard 4mm plugs. When the adaptor is connected to Smart Box the appropriate input will be labelled 'Adaptor'.

Typical use

The following example shows how the adaptor can be used to connect a Philip Harris Blue Box Light Sensor to Smart Box.



- Plug the adaptor into one of the analogue sensor sockets on Smart Box. The Smart Move software will rename the appropriate input 'Adaptor'. This default label can be changed as detailed later.
- Connect the Black socket on the sensor to the Black socket on the adaptor using a lead with 4mm plugs.
- Connect the Red socket on the sensor to the Red socket on the adaptor.
- Switch the sensor on.
- The Smart Move screen display will now show the current reading from the sensor. The sensor can be used in the same way as any other Smart Box sensor. The light level can be displayed using the PRINT command eg. PRINT ADAPTOR.

Renaming the sensor

The default name 'Adaptor' can be changed using the LABEL command as follows:

- Type as a direct command
  LABEL ADAPTOR LIGHT

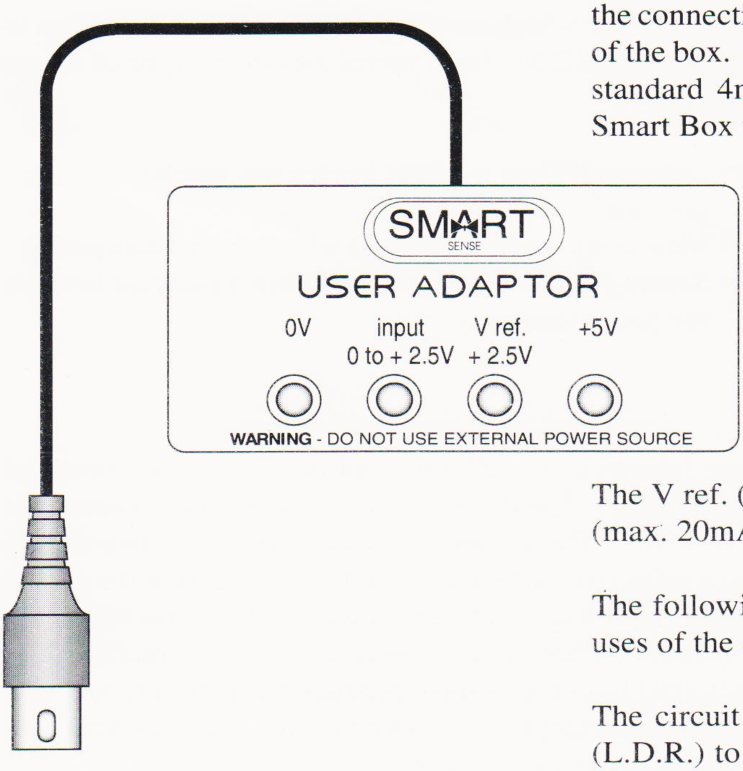
The sensor can now be referred to by the name LIGHT. For example PRINT LIGHT will now display the current value from the sensor in the output window.

Calibration

The calibration information supplied with the sensors can be used with Smart Move. Smart Move will display a full scale reading (255) when the sensor produces 1 volt of output. The table below shows the data for the Blue Box Light Sensor with the value that Smart Move will display alongside. This value is simply obtained by multiplying the Output Voltage column by 255.

Output Volts	Light Level Lux	Smart Move reading
0	1	0
0.2	10	51
0.4	100	102
0.6	1000	153
0.8	10000	204
1.0	100000	255

User Adaptor



The Smart Sense User Adaptor allows easy connections to the Analogue Sensor inputs to be made. The adaptor brings the connections from the Smart Box to the sockets on the top of the box. Connection to the adaptor should be made using standard 4mm plugs. When the adaptor is connected to Smart Box the appropriate input will be labelled 'Adaptor'.

The Yellow ‘input’ socket on the adaptor accepts a voltage between 0 and 2.5 volts. As the voltage on this socket rises the reading from the adaptor will rise linearly from 0 to a maximum reading of 255. The adaptor also includes a 0 Volt (black) and + 5 Volt (Red) socket to provide power to an external circuit. The external circuit can draw up to 50 mA from the adaptor. The V ref. (White) socket provides a stable 2.5 Volt source (max. 20mA) which can be used as a reference voltage.

The following example circuits show some of the possible uses of the adaptor.

Simple light sensor

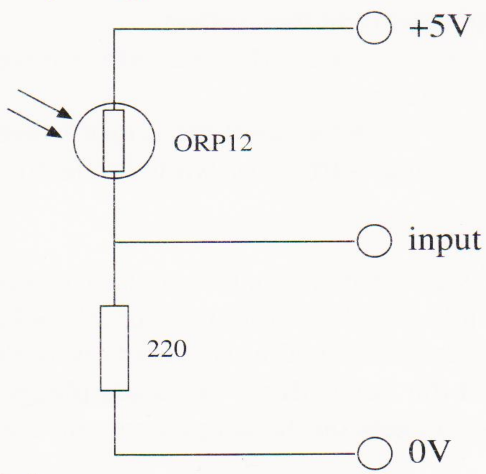


Fig.1

The circuit shown in fig.1 uses a light dependent resistor (L.D.R.) to form a light sensor. The L.D.R. and the 220Ω resistor form a potential divider. The resistance of the L.D.R. increases as the light falling on it decreases. In very dark conditions the resistance will be at least 1MΩ so only a small voltage will be present at the input socket. In bright conditions the L.D.R. will have a resistance similar to the 220Ω fixed resistor and so the input socket will see a voltage of around 2.5 volts which will give a maximum sensor reading of 255. Typical uses of this type of sensor include:

- Measuring daylight during the day to determine if it has been ‘sunny’ or ‘overcast’
- Detecting dusk and switching on a security lighting system.
- Measuring the amount of light which passes through a variety of materials.

Detecting rotation

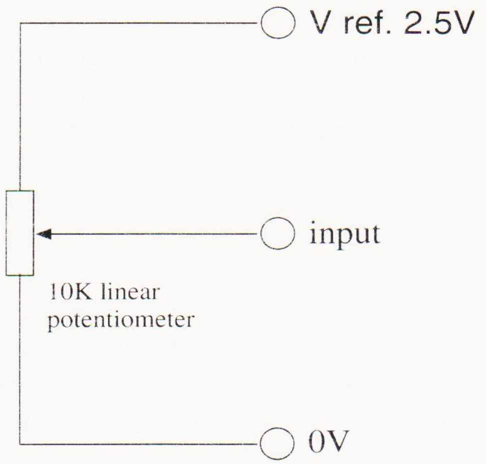


Fig.2

The circuit shown in fig. 2 uses a linear potentiometer to detect rotation. The track of the potentiometer is connected between the Vref. and 0 Volt sockets. The input socket is connected to the wiper and as the spindle is rotated the potential divider formed by the two sections of track produces a voltage which swings between around 0 volts up to the maximum of 2.5 volts. Typical uses of this type of sensor include:

- Detecting when a lifting bridge has reached the open position.
- Measuring the angle through which a pendulum swings.
- Sensing a water level by attaching a buoyant lever to the potentiometer.

Reflective opto switch

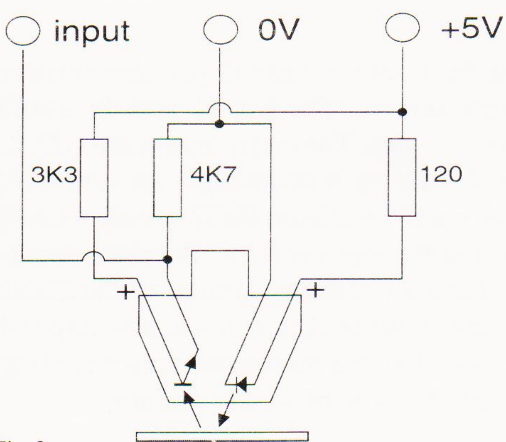


Fig.3

Figure 3 shows how a reflective opto switch can be connected to the adaptor. The device contains an infrared transmitter and receiver. The infrared beam is directed outwards and when a reflective surface is placed near the end of the device a change in voltage will occur on the line connected to the input socket. When the detector receives only a small amount of reflected light the voltage produced will be low and will rise as the detector picks up more infrared light. Typical uses of this type of sensor include:

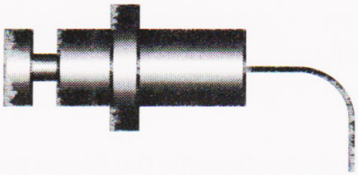
- Sensing the 'colour' of various objects by measuring the amount of infrared light they reflect.
- Measuring short distances as a reflective surface moves within the sensor's range.
- Reading 'bar codes' by detecting if the sensor is over a black (non-reflecting) surface or white (reflecting) surface.

The adaptor also allows the reflective opto switch to be used as a digital sensor. Unplug the lead from the 'input' socket on the adaptor, and plug it into the yellow socket of one of the digital sensor inputs on the Smart Box. The leads plugged into the '0V' and '5V' sockets on the adaptor remain connected.

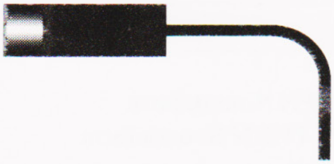
Digital Sensors

The range of digital sensors allow simple on / off conditions to be detected. The sensors are connected between the pairs of red and yellow digital sensor sockets on the top of the box. When the sensor is in its 'On' condition the indicator on the box will light. The Smart Move software can detect the condition of the switch as 'On' or 'Off' or alternatively '1' or '0'. Both these Smart Move program lines are valid:

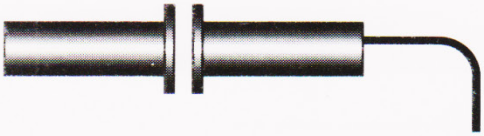
```
IF SENSOR 1 IS ON THEN FORWARD A
IF SENSOR 1 = 1 THEN FORWARD A
```



A **Push Switch** can be used as a manual input device. The switch is normally 'Off' and will be detected as 'On' when pressed.



A **Tilt switch** can be used to detect when an object has moved beyond a particular angle.



A **Magnetic Switch** will be 'On' when a magnet is close to the end.

Making your own sensors

Any device which completes a circuit between a red and yellow socket on Smart Box can be used as a digital sensor. This can be as simple as two contacts closing together. Switches can be manufactured using card and tin foil, pegs and drawing pins etc.

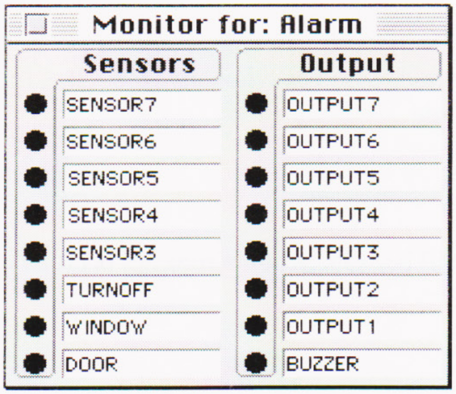
An example project

The following example uses three digital sensors and a buzzer to create a simple security system.

Sensors 0 and 1 are magnetic contact switches which will open as a door or window is opened. A push switch is connected to sensor 2 which is used to switch off the system. The Digital sensors can be 'Labelled' or given an alternative name using the LABEL command in Smart Move. This makes the sensors easier to identify. In the same way some of the Smart Move keywords can be changed. In this example the keyword OFF is changed to OPEN. The reference section gives more details. Type in the following as direct commands:

LABEL OFF open
LABEL SENSOR0 door
LABEL SENSOR1 window
LABEL SENSOR2 turnoff
LABEL OUTPUT0 buzzer

Type in all the procedures as shown then run the *Alarm* procedure.



Macintosh monitor window showing labels

Procedure: Alarm

REPEAT
IF DOOR IS OPEN THEN Soundalarm
IF WINDOW IS OPEN THEN Soundalarm
UNTIL TURNOFF IS ON
Stopalarm

Procedure: Soundalarm

SWITCH ON buzzer PULSE 50,50

Procedure: Stopalarm

SWITCH OFF buzzer

Smart Box Relay

The Smart Box Relay can be used to control electrical devices from the digital outputs on Smart Box. These devices will typically require larger voltages or currents than available direct from the digital output. The relay can switch a maximum of 3 Amps at 40 volts. The unit is protected internally by fuses which can be replaced if required by unscrewing the lid of the case. Fuses must be replaced with a fuse of the same type and rating.

The unit contains a double pole changeover relay. The Smart Box Relay should be connected to one of the digital outputs on Smart Box. When the digital output is 'off' current will be able to flow through the common (COM.) and normally closed (N.C.) connections. When the digital output is switched 'on' the Leds on the relay unit will light and a circuit will be made between the Common and normally open (N.O.) connections.

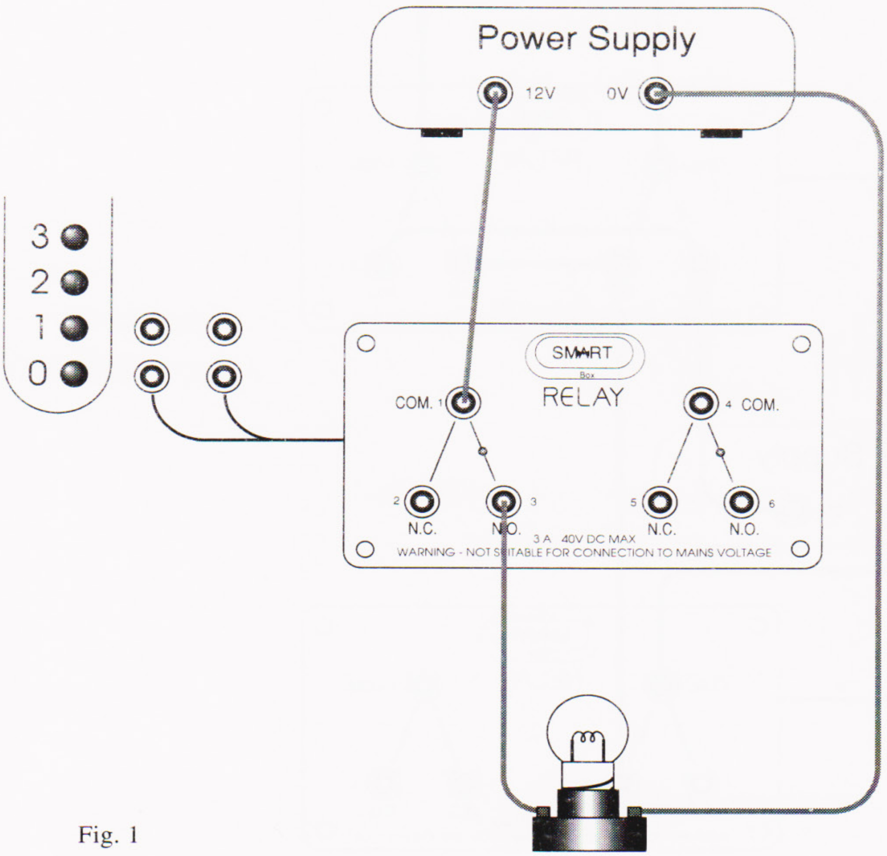


Fig. 1

Warning *The POWER, SPEED and PULSE commands should not be used with the relay as the rapid switching can cause mechanical failure of the unit.*

Fig. 1 shows a typical circuit in which the relay is being used to switch a device on and off. Power from an external supply is connected to one of the Common connections on the unit. The device, in this case a 12 Volt bulb, is connected between the Normally Open socket on the Smart Box Relay and the 0 Volt connection on the power supply.

Fig. 2 shows how two Smart Box Relays can be used to control a motor with on/ off and forward/ backward control. One relay is used to switch the power from the external supply on and off. The second relay controls the polarity the supply to the motor. Connections are made between the N.O. and N.C connections on each pole of the relay. Stacking 4mm plugs should be used to make these connections. In this example switching output 0 off and on will stop and start the motor while output 1 will control the direction of rotation of the motor.

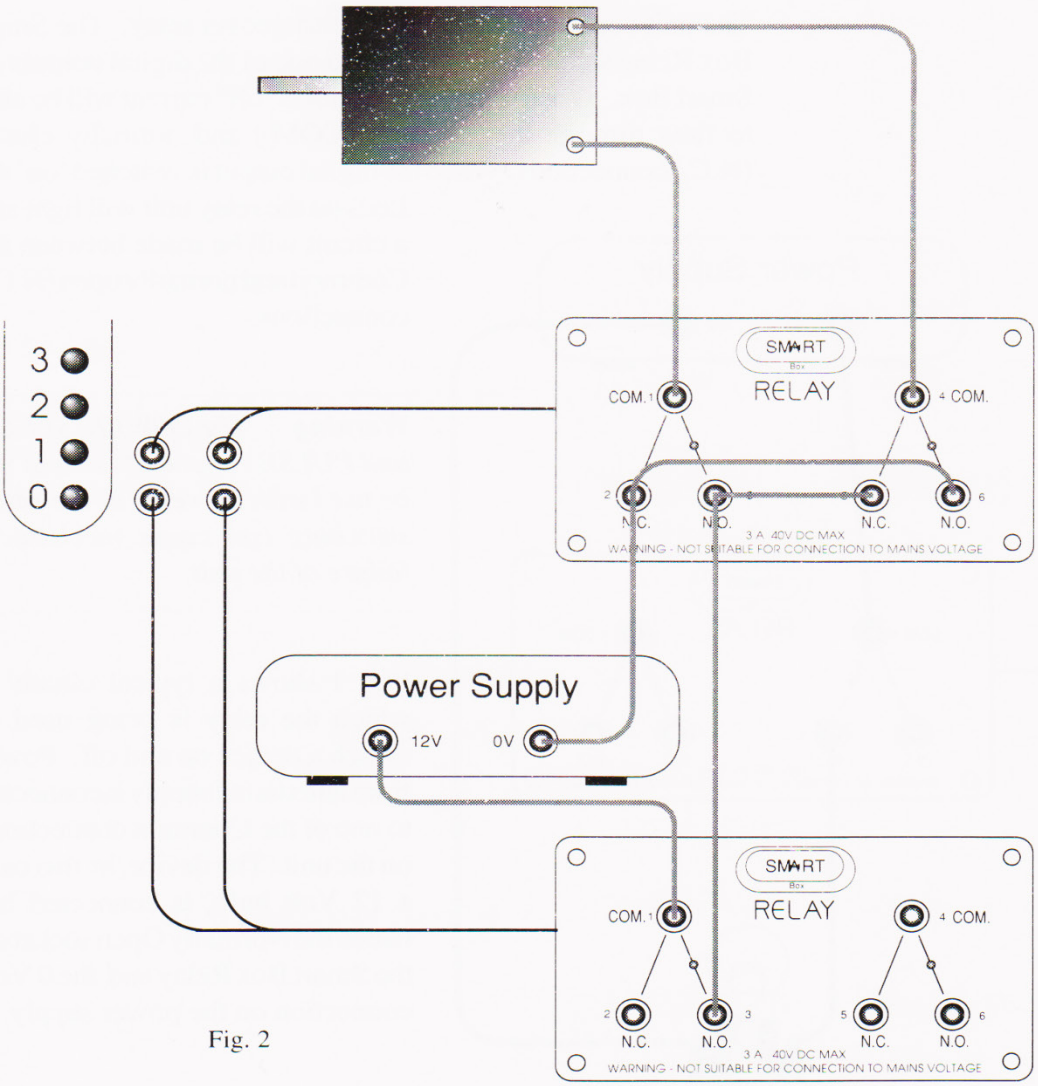


Fig. 2