

# Roamer User Guide

## 1. Introduction

### UNPACKING

Your Roamer package contains:

- The Valiant Roamer (batteries not included)
- The Roamer Activity Book.
- The Roamer User Guide.
- Programming Sheet.
- A set of Face Shapes.

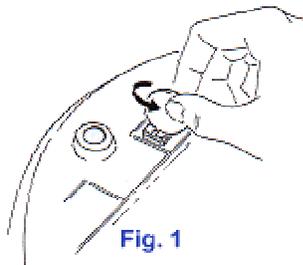


Fig. 1

### BATTERIES

The Roamer uses two 'lantern' <sup>\*</sup> batteries (e.g. Ever Ready PJ996) or the Valiant Rechargeable Battery Pack.

#### WARNING

The Valiant Rechargeable Battery Pack incorporates features essential for the safety and protection of the Roamer. If you use unapproved rechargeable batteries, you may damage the Roamer and will invalidate the warranty.

\* The international specification for lantern batteries is IEC 4R25, four cells in series, 6 volts with spring contacts.

### Loading the Batteries

1. Make sure the Roamer's slide switch, located in a depression at the front of the Roamer, is OFF (towards the Roamer's centre).
2. Turn the Roamer upside down.
3. Loosen the battery cover fasteners with a coin (fig 1).
4. Remove the battery cover.
5. Insert the batteries. Make sure that the springs are correctly located against the battery contacts.
6. Replace the battery cover.
7. Turn and push the battery cover fasteners until they click into place.

### When to Remove the Batteries

If the Roamer is not in use for more than one week remove the batteries to avoid corrosion of the contacts.

Remove the batteries from your Roamer when you are transporting it (e.g. posting it). If the cells leak, the Roamer could be damaged.

### Saving Battery Power

When the Roamer is switched on but not in use, it will sound a warning after five minutes. After six warnings, it will switch to a low power mode and lose the contents of its memory. Switch the Roamer off before restarting.

## Battery Life

The life of the batteries depends on how the Roamer is used. It uses less power when it is being programmed than when it is running around. If you run it on a rough surface like a carpet, or use accessories such as motors or lights, more power is consumed than when the Roamer is running on a smooth surface.

## SWITCHING ON

Once you have loaded the batteries, switch ON the Roamer by sliding the switch away from Roamer's centre. The Roamer should 'beep'. You are now ready to start teaching your Roamer.

### NOTE

If you switch off the Roamer, pause for at least three seconds before switching it on again to allow the microprocessor to reset. If the Roamer does not 'beep' when you switch it on or makes a strange sound, switch it off. Wait for three seconds before switching it on again.

## Switch off the Roamer when it is not in use

## 2. Programming Instructions

### TEACHING THE ROAMER

The Roamer moves forward and back, turns left and right, waits and makes sounds. You teach the Roamer to do this by pressing keys on the top of its body. There is a key for each of these functions, and a set of number keys.

To enter an instruction press a key, followed by a number. This tells the Roamer how far to move, how much to turn, how long to wait or what sound to make.

There are three other function keys: Sense, Two-State Outputs and Stepper Motor.

### ROAMER PROGRAMS

There are two types of program, the **GO Program** and **Procedures**.

The GO Program is a list of instructions carried out when you press **GO**.

A Procedure is a list of instructions with a name. Once you have defined the list, its name (e.g. P1) is used like any other instruction. When you enter its name in the GO Program, the Roamer will carry out the whole procedure list.

### MEMORY

The Roamer has two types of memory, **GO Memory** and **Procedure Memory**.

The Roamer will remember up to 59 instructions, and its powerful programming facilities enable it to carry out hundreds of actions.

When its memory is nearly full the Roamer will sound a warning, similar to the sound you hear when you press **CM**.

### GO MEMORY

Instructions in the GO Memory are carried out when you press **GO**. If you add more instructions after executing the GO Program, they will be added to the GO Program and carried out the next time you press **GO**.

The Roamer waits for two seconds after you press **GO** before executing the GO Program.

## CLEARING THE GO MEMORY

Pressing **CM** **CM** clears the GO Memory and allows you to enter a new GO Program. It does not clear the Procedure Memory.

The first time you press **CM** a warning is sounded. Pressing **CM** a second time clears the GO Memory.

If you press **CM** by mistake, press another key, or wait 10 seconds. The Roamer will then carry on with its GO Memory intact.

## DEMONSTRATION PROGRAM

If you press **GO** after you switch on the Roamer, it will carry out the Demonstration Program. This can be used to show beginners the basic Roamer functions.

**REMEMBER TO CLEAR THE DEMONSTRATION PROGRAM FROM THE GO MEMORY BEFORE PROGRAMMING.**

## FORWARD AND BACK

Pressing  or  followed by a number from 1 to 99 tells the Roamer to move forward or back that number of units. (See **UNITS OF DISTANCE AND TURN**).

 1 0	Moves the Roamer forward 10 units.
 3	Moves the Roamer back 3 units.
<b>GO</b>	Executes the GO Program: forward 10 back 3.

(These examples are illustrated in the style suggested on the programming sheet).

## RIGHT AND LEFT TURN

Pressing  or  followed by a number up to 999 turns the Roamer to the right (clockwise) or left (anticlockwise) that number of units. (See **UNITS OF DISTANCE AND TURN**).

 4 5	Turns the Roamer 45 units to the right.
 1 0 6	Turns the Roamer 106 units to the left.
<b>GO</b>	Executes the GO Program: right 45 left 106.

## **W** WAIT

Pressing **W** followed by a number from 1 to 99 tells the Roamer to be still and quiet for that number of seconds.

 3 0	The Roamer moves forward 30.
 5	The Roamer is still and quiet for 5 seconds.
 2 0	The Roamer moves back 20.

## SOUND

The Roamer has a programmable sound facility. You need to specify how long each note will last (duration) and how high or low the note will be (pitch).

To play a note, press  followed by a number from 1 to 8 for its duration, and another number from 1 to 13 for its pitch.

If you want a rest (silent note), enter 14 for the pitch.

 1 1	A note of short duration at low pitch
 4 1 4	A medium duration rest.
 8 1 3	A note of long duration at high pitch.

## UNITS OF DISTANCE AND TURN

When you switch the Roamer on it moves forward and back in units equal to its body length (30cm) and turns in angular units of 1 degree. You can change the units of distance and turn. This allows the teacher to set the most appropriate units for each teaching period.

Once the units are set, they remain unchanged until you redefine them, or switch the Roamer off. Clearing the GO Memory does not change the last unit setting.

To clarify which bracket opens and closes a list, [ and ] have been used instead of 

### Changing the Units of Distance

To change the unit of distance, press  or  followed by 

Then press a number between 1 and 99. The number determines how many centimetres will be in each unit of distance.

Then press  again.

The GO Program must be run before the change becomes effective.

 ①	Moves the Roamer forward 30 cm (the default setting).
 [① ⑩]	Sets the units of distance to 10cm.
 ②	Moves the Roamer back 20 cm.
 [①]	Sets the units of distance to 1cm.
 ⑤	Moves the Roamer forward 5 cm.
	When GO is pressed a second time, the first instruction  ① moves the Roamer forward 1cm, in accordance with the last setting  [ 1 ].
	

## Changing Units of Turn

To change the unit of turn, press  or  followed by . Then press a number from 1 and 999. This number determines how many degrees will be in each unit of turn.

Then press  again.

The GO Program must be run before the change becomes effective.

 [⑨ ⑩]	Sets the unit of turn to 90
 ②	Turns the Roamer 180 to the left.
	

## Changing the tempo and octave

Roamer sounds may be played at one of five tempos and one of three octaves.

To change tempo or octave press  then .

Press a number between 1 and 5 for the tempo. 1 is fastest, 5 is slowest.

Press a number between 1 and 3 for the octave. 1 is lowest and 3 is highest.

Press  again.

Ⓜ [1 1]	Sets the fastest tempo and the lowest octave.
Ⓜ 1 1	Plays the Roamer's lowest, shortest note.
Ⓜ [5 3]	Sets the slowest tempo and the highest octave.
Ⓜ 8 1 3	Plays the Roamer's highest, longest note.
GO	

When you press **GO**, the Roamer starts with the tempo set at 3 and the octave set at 2.

### Musical equivalents

When defining the pitch, 1 represents C, 2 represents C#, 3 represents D, etc., up a chromatic scale.

When defining the duration, 1 is equivalent to a semiquaver, 2 is equivalent to a quaver, 4 is equivalent to a crochet, etc.

Setting the tempo at 1 produces 170 beats per minute; 2, 3, 4 and 5 produce 140, 100, 80 and 60 respectively.



### SOUND CUES

When you press a key, the Roamer makes a sound. If the instruction is valid the sound is positive. Invalid instructions produce a negative sound.

	Gives two positive sounds.
	Gives a positive, a negative then a positive sound. The Roamer turns right 9 units.
	Gives three positive sounds.
	Gives a positive sound, followed by a 2 second pause, then the Roamer moves forward 1 unit, turns right 9 and then left 90 units.

### STOPPING THE ROAMER

If the Roamer is carrying out a list of instructions and you want to stop it before it has finished, hold down any key until it completes the current instruction. This feature allows the children to stop the Roamer if it is not doing what they wanted it to do.



### CANCEL ENTRY

Pressing  cancels the last instruction or part instruction from the GO Program.

	
	Back key pressed in error.
	Cancels the last entry,  .
	
	
	The Roamer turns left 40, moves forward 3 and turns right 90.
	Cancels the last entry,  .
	The Roamer turns left 40 and moves forward 3.
	Cancels the last entry.
	The Roamer turns left 40.



### REPEAT

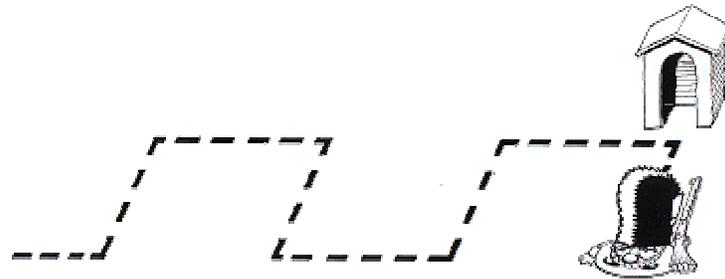
Pressing , followed by a number up to 99, will make the Roamer repeat a following list of instructions, that number of times. The list must be enclosed in a "box" by placing  before and after the list.

<b>R 4 [</b>  <b>GO</b>		Repeat 4 times. Forward 3. Left 90 End the repeat. The Roamer moves through a square.
-------------------------------	--	---

Repeat lists can contain other repeat lists. This is like putting one box inside another. You can place up to five "boxes" within "boxes" i.e., nest repeats 5 deep.

<b>R 4 [</b>  <b>GO</b>		Start the first repeat  Start second repeat. J 1 5 J 8 1 End second repeat. End first repeat. The Roamer turns through 360 units. Every 60 units it stops and plays 2 notes, four times.
-------------------------------	--	---

This program has a nest of repeats 2 deep.



### PROCEDURES

A procedure is a list of instructions named P, plus a number from 1 to 99. The list is enclosed in brackets **(1)**.

P 4 [		Starts the definition of Procedure 4.
	△ 3	First instruction of P4.
	9 0	Second instruction of P4.
]		Ends the definition of P4.
GO		Nothing happens. P4 is not yet part of the GO Program.
P 4		Enters P4 in the GO Program.
GO		P4 is executed: forward 3 left 90.
CM		Removes P4 from the GO Program but leaves it in the Procedure Memory.
CM		

When keying in instructions in a procedure definition, sound cues for valid and invalid instructions are still made. However a different pitch is used to inform the programmer that the instructions are part of a procedure, not part of a GO Program.

### Putting a Procedure Inside Another Procedure

A procedure can be part of another procedure.

P 1 [		Starts definition of Procedure 1.
	P 4	Procedure 4 (already defined) is the first instruction in P1.
	P 9	Procedure 9 (as yet undefined), is the second instruction in P1.
	△ 3	The third instruction in P1.
	]	Ends definition of Procedure 1.
P 1		Puts Procedure 1 in the GO Memory.
GO		The Roamer executes P4. Since P9 is undefined, the Roamer will ignore it before executing the last instruction △ 3.
P 9 [		Starts definition of Procedure 9.
	R 4 [	
	J T 6	
	J T 8	
	]	End of Repeat list.
	]	Ends the definition of Procedure 9
GO		The Roamer executes P4, then P9, and finally moves forward 3 units.

Procedures can be nested to any depth - memory permitting.

### CHANGING A PROCEDURE

To change a procedure, simply redefine it.

P 6 [		Starts definition of P6.
	△ 5 0	
	W 1 0	
	} 2 0	
]		
P 6		Enters P6 into the GO Program.
GO		The Roamer moves forward 50, waits 10 seconds and turns right 20.
P 6 [		Redefines P6.
	▽ 7	
]		
GO		P6 is still in the GO Program and executes the new definition: back 7

## Erasing a Procedure

To erase a procedure from the Procedure Memory, enter **P** and the procedure number, followed by **C** and **C**

P 1 2 [		
	R 4 [	
	△ 1 0	
	} 9 0	
]	]	
P 1 2		The Roamer moves in a square pattern
GO		
P 1 2 [		Erases the definition of P12.
]		
GO		Nothing happens because P12 is now undefined.

---

## RULES ABOUT USING PROCEDURES

### 1. A procedure cannot use itself

Roamer procedures are not recursive.

P 7 7 [	P 7 7	Not allowed Pressing any other key gives an illegal instruction sound.
---------	-------	--

2. You can define only one procedure at a time

3. A procedure cannot include a procedure with a lower number

P 6 2 [	P 6 3	is a valid instruction
]	P 6 1	is an invalid instruction

This means that Procedure 1 can call all other procedures, and Procedure 99 cannot call any.

### 3. Teachers' Section

#### PROGRAMMING ROAMER

The natural response when first meeting a Roamer is to prod a few keys and see what happens. The Roamer's keyboard, sound cues and simple programming structure support this exploratory process. With a minimum of teacher intervention children can achieve remarkable success. They quickly learn how to make the Roamer move about and make sounds. (Some teachers have found it useful to mask some of the keys during this early stage.)

The exploratory process can lead to more purposeful work, where children program the Roamer to behave in a particular way; for example, teaching the Roamer to move through an obstacle course.

As their expertise and confidence grows, children are able to tackle more substantial projects. Again, with its 'Repeat' and 'Procedure' facilities, the Roamer supports these developments.

---

#### WRITING ROAMER PROGRAMS

It is advisable first to write programs on paper, then to key them into the Roamer. There are several positive results to be gained from this discipline:

1. It develops planning and thinking skills.
2. It develops the important skill of recording information.
3. It provides the child and the teacher with a record of their programs and the processes involved in achieving their objectives.

The programming sheet included with your Roamer suggests a technique for writing Roamer programs.

#### BENEFITS OF USING PROCEDURES

Roamer Procedures have four distinct advantages:

##### 1. Procedures support "Bottom Up" programming

The "Bottom-Up" programming approach is to solve new problems using existing knowledge. If the Roamer has been taught a procedure, it can be used in the performance of new tasks.

For example, suppose that Procedure P1 plays a tune which has previously been taught to the Roamer. A student now wants to use the tune to make the Roamer into an ice cream van.

△ ①	Move to the first location.
Ⓟ ①	Play the tune to tell the children the van has arrived.
Ⓜ ③ ①	Serve the ice cream.
△ ②	
Ⓝ ⑨ ①	Move to the next location.
△ ①	
Ⓟ ①	
Ⓜ ④ ⑤	
GO	Delivers the ice cream

## 2. Procedures facilitate "Top Down" problem solving

The "Top Down" approach of dividing a complex problem into smaller, more manageable problems is an important life skill. The Roamer's Procedure facility encourages this approach.

For example, a class project could be divided into group projects. The first group's program is contained in procedure P1, the second group writes P2 and so on. The first group can solve their problem using basic Roamer instructions or procedures P11, P12, P13 ....

Group 2 uses procedures P21, P22, P23 etc. The main procedures, P1, P2 etc., are finally combined in the GO Program.

## 3. Procedures increase the Roamer's action capacity

Ⓟ ②		A procedure consisting of 30 instructions.
Ⓝ ⑨ ①		
Ⓜ ④ [	Ⓟ ②	
]		
Ⓝ ⑧ ⑨		The Roamer will perform 182 actions compared with the 59 action capacity using single instructions.
Ⓟ ②		
GO		

## 4. Procedures simplify the process of changing a program

To change the GO program, the GO memory must be cleared and the whole list must be rekeyed with the modified instructions. If instructions are entered as procedures, only the procedures need to be redefined.

<p>P 3</p> <p>P 5</p> <p>P 6</p> <p>P 5 1</p> <p>GO</p>		<p>Roamer executes P3, P5, P6 and P51.</p>
<p>P 3 [</p> <p>J 8 9</p> <p>J 4 6</p> <p>J 8 1</p> <p>]</p> <p>GO</p>		<p>Redefines Procedure 3.</p> <p>Roamer executes the new P3 and the original P5, P6 and P51.</p>
<p>P 7</p> <p>P 4</p> <p>P 1 0</p> <p>P 2 1</p> <p>GO</p> <p>CM</p> <p>CM</p> <p>P 2 1</p> <p>P 4</p> <p>P 7</p> <p>P 1 0</p> <p>GO</p>		<p>Executes P7, then P4, then P10, and finally P21.</p> <p>Clears GOMemory.</p> <p>Executes P21, then P4, P7 and P10.</p>

## ACTIVITIES USING THE ROAMER

The **Roamer Activity Book** contains tasks and games for children of different ages and abilities. These ideas use the basic Roamer and do not need any Accessory Packs.

Each activity involves several educationally valuable ideas which children approach through experimentation and exploration. Children playing 'Collecting Rubbish' may use arithmetic, linear and angular measurement, programming, trigonometry and geometry.

The activities should not be used prescriptively. They are starting points around which the teacher may create opportunities for pupils to develop their understanding by testing and revising their own solutions to the problems.

The following case study illustrates this approach:

*"Two boys and two girls, from a class of 8 year-olds, played the 'Collecting Rubbish' game from the Roamer Activity Book. They collected a selection of discarded items from the playground and scattered them around the classroom floor. Each child wrote a 'Rubbish Collecting' program. The program was not considered successful unless the Roamer returned, more or less, to its starting point. This involved a process of trial and error. Although the game originally included a competitive element; the children became so absorbed that they began to help each other.*

*The children had to estimate distances and angles. They used words like "perimeter" in the correct context. As their understanding of these ideas grew, their game strategies became more efficient.*

*An interesting development began when children decided which type of litter to collect.*

*Each child wrote a program to collect a particular type of rubbish. The other children had to guess the rationale behind the selection. One child programmed the Roamer around items that "would go rotten". The children disagreed with some of his selections so it was decided to conduct experiments to discover what would rot.*

*Other children selected paper objects or food containers. One child selected large objects, partly because it was easier to drive the Roamer around them. She justified her choice by saying, "They covered more area."*

*During discussions afterwards, one child noticed the similarity between selecting objects in this way and formal Venn diagrams. The children fitted the Roamer with a pen and drew lines around the different sets.*

*The mathematical links were obvious. What was surprising was the interest the children developed in litter. The Roamer caught their imagination. Because they were actively 'doing', the topic was approached at the correct level for their understanding. Their interest grew and they wanted to help other children realise the wider ecological issues involved."*

*Alan Coode, Headteacher, Southmead Junior School, South London.*

## **ROAMER CONTROL FEATURES**

You can expand the Roamer's range of activities with a Roamer Control Box. This provides the Roamer with a Sense Input Line, four Two-State Outputs and a Stepper Motor Output. They are controlled by the following keys:

'S'

for Sense

'T'

for the Two-State Outputs

'M'

for Stepper Motor.

The instructions on how to program with these keys are supplied with the Roamer Control Box.

Connecting sensors and motors greatly extends the scope for Roamer characterisation.

'Mophead' the Roamer guard dog could be programmed to prick up its ears and bark when it hears a sound. 'Fiona' the Fire Engine could flash her light, raise her ladder and play her siren. The Roamer Bee could flap its wings and hum "Flight of the Bumble Bee".

Roamers can be programmed to respond to light, sound, touch, temperature, etc.. The motors can be used to raise and lower a pen, operate a robot arm, raise a flag, stir a cup of tea or even play cricket. If the child can think of it, the Roamer can probably do it.

## **4. Looking after your Roamer**

### **CARE OF THE VALIANT ROAMER**

Check regularly that Roamer's drive axles are free from dust and fibres, particularly if you run Roamer over carpet.

Attach things to the Roamer body with Valiant Suction Pads, sticky tape or Blu-Tack. Don't use adhesives.

Clean the Roamer's shell with a damp cloth. Don't use solvents or abrasives.

It is not recommended that you decorate the Roamer with pens or markers, as they leave an indelible trace.

Glue and marker pens can be safely used on the Roamer Jackets.



**IF YOUR ROAMER IS MISBEHAVING**

Roamer is designed to be simple and reliable. If you have problems, check the batteries. Swap them over. Even if you bought them recently they may have been on your supplier's shelf for some time and be weak. Ask for them to be exchanged. Try another pair in the Roamer.

If you have a Valiant Rechargeable Battery Pack, recharge the cells.

If there is still a problem, telephone the Valiant HELPLINE. Many difficulties can be resolved by speaking to our support staff.

**WARNING**

Do not remove the Roamer's shell. This will invalidate your warranty.