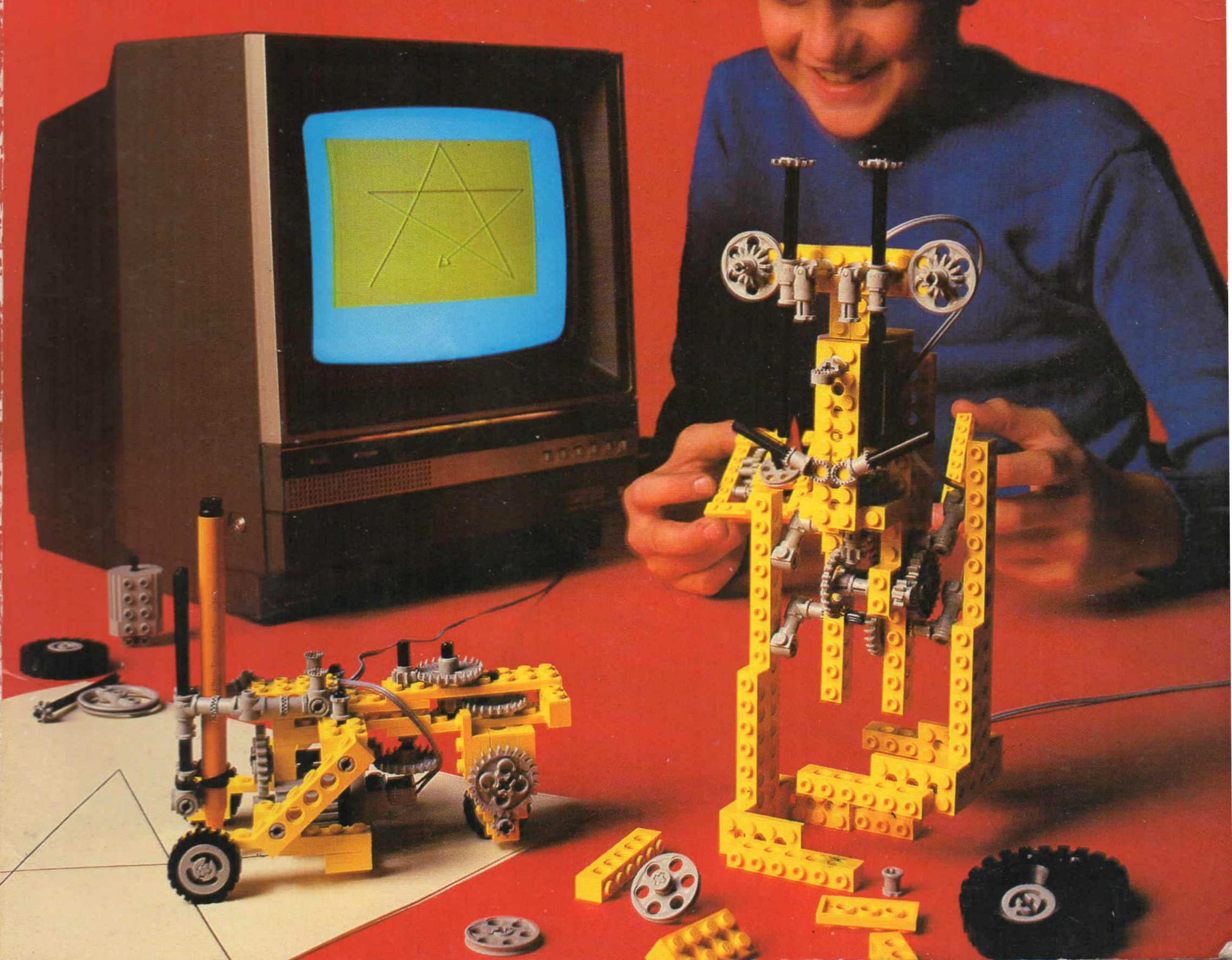
Make And Program Your Own Robots

For the Commodore 64 and Vic 20

WILLIAM CLARK



MAKE AND PROGRAM YOUR OWN ROBOTS

FOR THE COMMODORE 64
AND VIC 20

WILLIAM CLARK

DEVISED BY LIONEL BENDER
DESIGNED BY BEN WHITE

BEAVER BOOKS

CONTENTS

About this book 3
Linking up robots to your computer 4
Programming and robot control 5
Wiring and electrical circuits 6
Using LEGO® 7
Projects
1 Walking Android 8
2 Whirly Turtle 11
3 Lift Operator 15
4 Card Reader 20
5 Mini-arm 25
6 Plotter 32
7 Maxi-arm 40
Useful addresses and information 48

A Beaver Book
Published by Arrow Books Limited
17-21 Conway Street, London W1P 6JD

London Melbourne Sydney Auckland
Johannesburg and agencies throughout the world

An imprint of the Hutchinson Publishing Group
17-21 Conway Street, London W1P 6JD

First published 1985

Printed in Great Britain by
Philip Print Limited, London
Bound by Hunter & Foulis Limited, Edinburgh

ISBN 09 942660 9

Make and Program Your Own Robots
was conceived and produced by
Lionheart Books
10 Chelmsford Square, London NW10 3AR

Editor Lionel Bender
Designer Ben White
Illustrators Hayward Art Group
Typesetting Facet Filmsetting Limited
Robot photography Ian McKinnell

ABOUT THIS BOOK

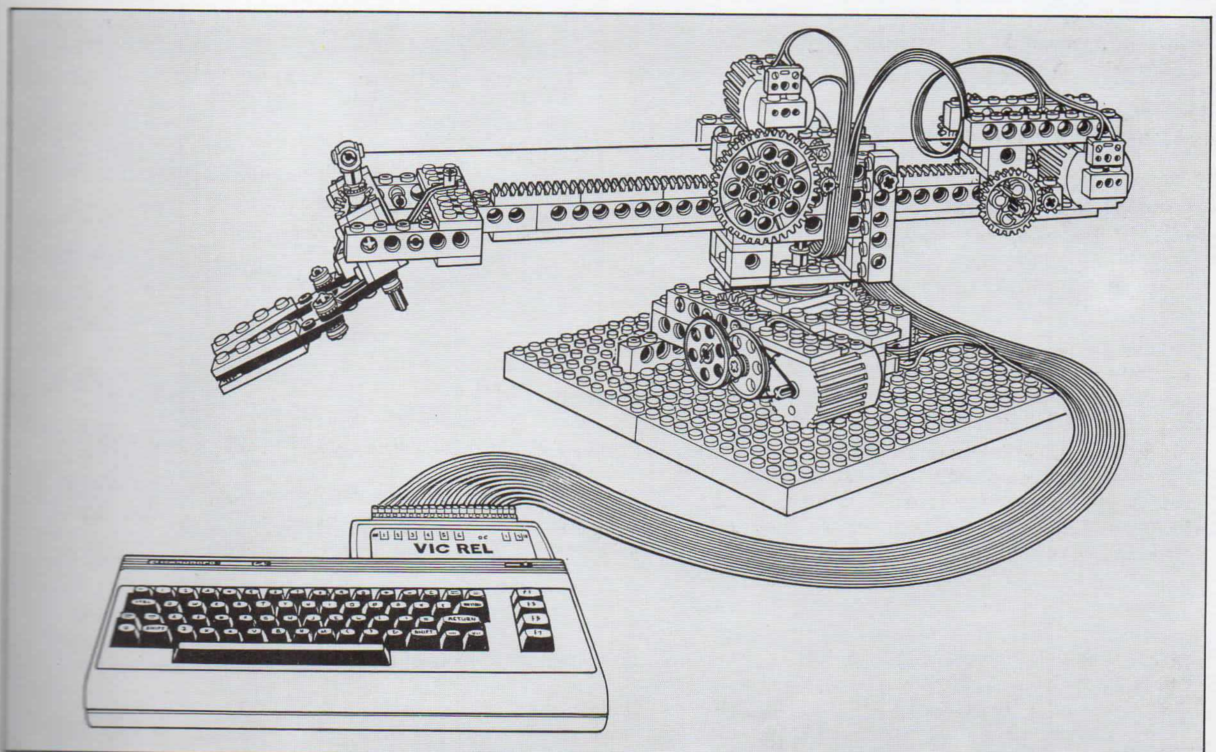
Many of you may think that typical robots are like R2D2 of *Star Wars* fame, the Daleks of *Dr Who* or like machines that walk and talk like a human being. This is mostly untrue, as you will discover from the projects in this book. A robot is any machine that can be programmed by a computer to carry out a task automatically. Most robots do simple jobs that people find boring or too dangerous to do.

There are seven robot projects in this book for you to try. The projects are designed to give you an insight into the basics of how robots are built and work. They are very simple and start with a walking robot and progress to a 'pick and place' robot arm. For each project we provide full details of what the robot can do, how it does it, what you will need to make it, how to build it, link it to your home computer, and program it. After completing the projects, you should be able to build and program robots of your own design.

All our robots are made from LEGO® Technic building sets and motors. Many of you will probably already have a collection of LEGO pieces.

To program the robots, you will need, in addition to your home computer, an interface unit. We have used a readily available purpose-built interface unit but if you are good with electronics or know someone who is, you could make your own from the details we provide. To SAVE the robot control programs, you will also need a cassette tape recorder or disc drive unit, and to display and check your programs, a tv or monitor.

Before starting the projects, read the following four pages to become familiar with the basic techniques and skills you will need to build and program the robots. Then, we recommend that you try the projects one at a time in the sequence they are presented in the book. None of the projects should take you more than two or three hours to complete.



LINKING UP ROBOTS TO YOUR COMPUTER

To control the robots in this book using your computer you will need an interface unit. This is a device that links together the motors in the robots, the battery box power source and the computer. Inside the interface unit is an electronic circuit that switches the power to the motors on and off when instructed by the computer. We have used an interface unit called a VicRel. It can be obtained from any good computer dealer or from one of the suppliers listed on page 48.

The VicRel comes with an instruction leaflet. Study this carefully before proceeding. The unit contains six electronically controlled switches called relays. A relay is a switch in which the contact – the part that does the connecting (closing) and disconnecting (opening) – is moved by the force of an electromagnet. When the magnet is on, the contact moves to the conducting position. When the magnet is off, the contact is pulled away by a small spring and the relay stops conducting. The relays allow the computer to switch on and off devices such as motors, lamps and alarm bells.

USING THE INTERFACE

Turn off your computer (remembering first to SAVE any important programs) then plug the interface unit into the user port on the back of the machine. Now turn the computer back on. First you must set up the user port to transmit signals to the relays and receive signals from the switch inputs. Type

COMMODORE 64	**VIC 20**
POKE56579,63	POKE37138,63

The relays are numbered 1 to 6. Each relay has a control value as follows

Relay	1	2	3	4	5	6
Control value	1	2	4	6	16	32

To switch on any relay, type

POKE56577, control value	POKE37136, control value

For example, to switch on relay, 1 type

POKE56577,1	POKE37136,1

To switch on relay 5, type

POKE56577,16	POKE37136,16

By adding together control values you can switch on combinations of relays. For example, to switch on relays 2 and 6 (control values 2+32), type

POKE56577,34	POKE37136,34

To switch on relays 3 and 5 (control values 4+16), type

POKE56577,20	POKE37136,20

To turn off all the relays, type

POKE56577,0	POKE37136,0

When you type these commands only the relays you want to go on are operated, all the others are turned off (if they are not already off).

The following program will switch relay 1 on and turn on the motor when you press any key:

```
10 POKE56579,63        10 POKE37138,63
       20 GETA$:IFA$=""THEN20
30 POKE56577,1         30 POKE37136,1
```

SWITCH INPUTS

Also on the VicRel there are two switch inputs. These allow you to connect ordinary switches to the interface so that the computer can sense when the switches are open (off) or closed (on). In some of our projects we use simple switches as sensors. You could, though, use switches in, say, floor mats or door handles and use your computer to operate a burglar alarm.

Using the switch inputs

To find if either of the switches is open or closed, you have to 'look into' the computer's memory, where an item of data is automatically changed by the switches turning on and off.

To test if switch 1 is closed (on), type

PRINT PEEK(56577)AND128 (Commodore 64)
PRINT PEEK(37136)AND128 (Vic 20)

If it is closed, the computer will print 0.
If it is open, the computer will print 128.
Testing for switch 2, instead type

PRINT POKE(56577)AND64 (Commodore 64)
PRINT POKE(37136)AND64 (Vic 20)

and the computer will print 0 if it is closed and 64 if it is open.

This happens whatever state the relays or the other switch sensors are in.

To convert the programs in this book for the Vic20 change line 20 in every program to:
20 PRT=37136:POKE37138,63

(In case you cannot obtain a VicRel or should you wish to build your own interface unit, we supply a circuit diagram on page 48.)

PROGRAMMING AND ROBOT CONTROL

When you have built a robot you are going to control using your computer, type in the program printed with the project. If you have not typed in programs before, look in your computer's manual to see how it is done.

The interface unit should be plugged into the user port before you turn on the computer and start typing in the program. Take care to type in the lines correctly. Do not try RUNning the program until you have typed it all in.

When you have entered all the program, SAVE it on tape or disc. Do not RUN the program, unplug the interface unit or turn off the computer.

Having SAVEd the program, VERIFY it to check it is stored properly. Then try RUNning the program. If the program causes the computer to crash, LOAD the copy on the tape or disc back into the computer and check it through on the screen.

Writing your own control programs

After you have done some of the projects, you should start to understand how we use the interface unit to control robots. You should be able to write your own simple programs to switch motors on and off. We use some of the interface unit relays to do the pole reversing (changing the direction of rotation of the motors) and some to switch the motors on and off. To help you understand the structure of our programs and what each line does, opposite is a copy of our Lift Operator project program fully labelled and annotated (see project on pages 15-19).

CHECKING CONTROL SYSTEMS

To check a program, first carefully compare the lines you have typed in with those in the program printed in the book. If you think your program is correct but you are still having problems, refer to the table below. Here we list some of the most common problems which occur with the robots together with the remedies.

Problem	Remedies
None of the motors will run	Check the batteries are fresh. Check the wiring and connections are good. Check that the LEGO plugs are plugged in properly. Make sure you are switching the correct relay.
All the motors run in the wrong direction	Change the position of the pole reverser on the battery box. If you are not using a pole reverser, turn the plugs over on the battery box.
One of the motors runs in the wrong direction	Turn the plug over on that motor.
The motors run but will not stop under computer control.	Check the switches are sending pulses to the computer by watching the green lights flash on the VicRel. Check that when one motor stops, its switch remains in the on position.

```
 10 REM**LIFT CONTROLLER**
 20 PRT=56577:POKE56579,63
 30 UP=9:DW=6:SP=0:CL=1
 40 INPUT"WHICH LEVEL 1-7";NL
 50 IFNL>7ORNL<1ORNL=CLTHEN40
 60 TD=NL-CL
 70 IFSGN(TD)=-1THENPOKEPRT,DW
 80 IFSGN(TD)=1THENPOKEPRT,UP
 90 FORI=1TOABS(TD)*6
 100 IF(PEEK(PRT)AND128)=0THEN100
 110 IF(PEEK(PRT)AND128)=128THEN110
 120 NEXTI:CL=NL
 130 POKEPRT,SP:GOTO40
READY.
```

20 set user port address
30 set variables to drive Up, Down, Stop. Set Current Level to 1
40 ask Which Level. Set Next Level to this value
50 check level between 1 & 7
60 find Travel Distance
70 switch motor on. If travel distance negative:Down
80 switch motor on. If travel distance positive:Up
90 set up program to count 6× travel distance pulses from the interface
100 wait for switch to open
110 wait for switch to close
120 finish counting pulses. Set Current Level to Next Level
130 switch motor off. Go back to line 40

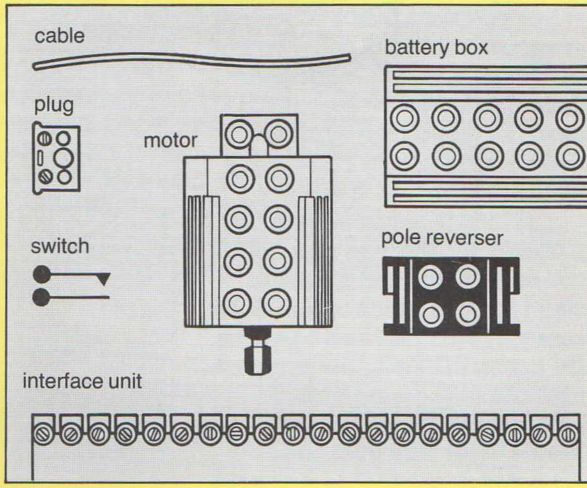
WIRING AND ELECTRICAL CIRCUITS

To connect up your robots to the interface unit you will have to do some simple wiring. For each project we have provided a simple wiring diagram in which we have drawn the components life-like. The illustration opposite shows the components.

The cable we use for most of the projects is called ribbon cable. It is flat like a ribbon and has several strands which run side-by-side. It is best to buy about 2 metres of 20-way (20-strand) ribbon cable which will suffice for projects in the book.

In our wiring diagrams we have shown the interface unit on the left, with the terminals numbered, and the motors, battery box and so on, and the wires coming from them, on the right.

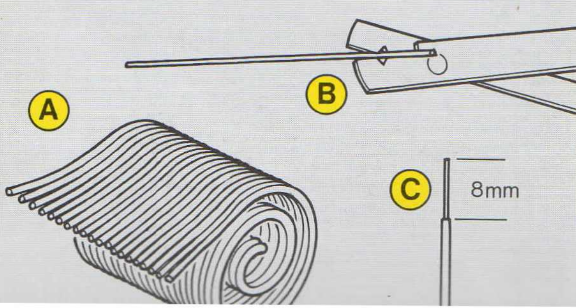
The ribbon cable and all other electrical components are obtainable from any good electrical or electronics store or supplier.



PREPARING THE RIBBON CABLE

In our larger projects we show you in the wiring section how best to cut and separate the individual strands of the ribbon cable to wire up the robot motors and switches. Separate the strands by gently pulling them apart. Some of the projects use non-standard widths of ribbon cable, such as 4-way or 12-way. In these cases, split the 20-way cable along the entire length you wish to use then cut off the appropriate section.

Once you have cut the cable to length, strip the plastic sheath off the ends of the individual strands. It is best to use wire-strippers for this. Close the cutting blades gently onto the strands so you cut the plastic but not the metal wire. Then slide off the plastic. Strip about 8mm of sheath off the end of each strand. Finally, twist together the metal fibres of each wire.

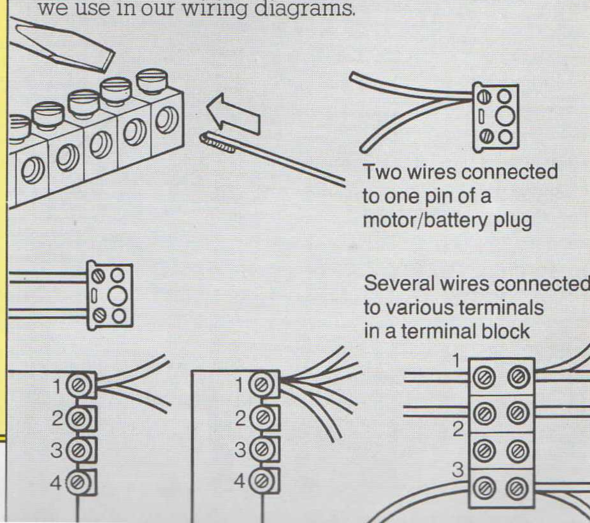


MAKING THE CONNECTIONS

To connect the wires from the motors, switches and so on into the VicRel, undo the little terminal screws using a small screwdriver and into each hole of the unit poke the wire or wires as appropriate. Tighten the screws down firmly but not too hard. It is sometimes useful to bend each twisted wire back against its plastic sheath. This can stop the wire breaking if accidentally tugged when fixed in the VicRel.

Wherever you have to make an electrical connection, the wire should actually touch (make contact with) the metal of the terminal.

Below are examples of some of the connections we use in our wiring diagrams.



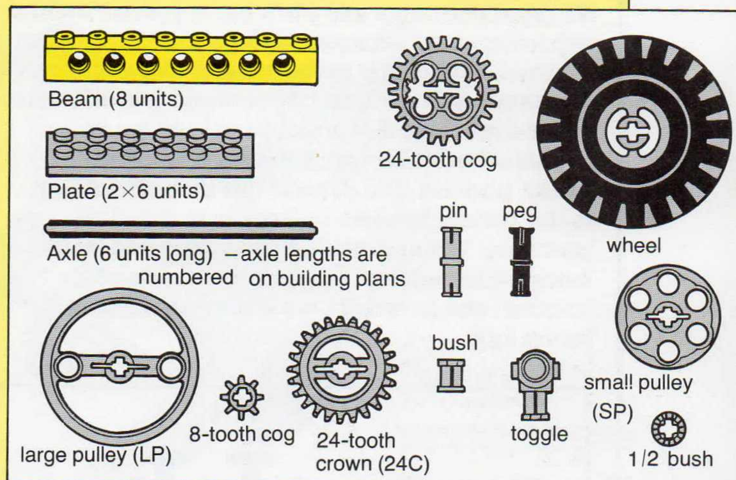
Two wires connected to one pin of a motor/battery plug

Several wires connected to various terminals in a terminal block

USING LEGO

The robots in this book are all constructed from LEGO® Technic pieces. We chose LEGO because it is the most popular, most widely available and most versatile of all children's model construction systems. It is of course possible to make robots with other types of building kits and you might like to try this after making our designs.

To help you follow our building plans, we have adopted the same type of step-by-step illustrations and instructions and the same component names as used in LEGO plans. The illustration opposite shows and labels the main components. Where there may be confusion over the name used for a component or the component is unusual we have illustrated it in the parts chart we provide for each project.



LEGO YOU WILL NEED

The earlier projects in the book are designed so that they can be built with just the parts contained in a standard LEGO kit or combination of standard kits. But if you do not already have the kit we recommend you buy, you can try and follow our basic design and modify the robot to suit the pieces you do have. You may even be able to improve on our designs!

If you need to obtain one or two specific components you can probably obtain them from the LEGO Spares Service. To order from this service you will need an order form printed at the back of a LEGO catalogue. The catalogues are available from toy shops and department store toy departments.

The best kits for building robots are those which contain lots of pins, pegs, gears and toggles as well as beams and plates. The motor starter set No. 8050 is an ideal set to start with. Supplementary set No. 8710 is a very good extension of any LEGO collection if you are going to progress to building large robots. The motors used in the projects are the battery powered 4.5V Technic motors but you can use the 12V Technic motors (as long as you have a power pack) or the large 4.5V motors although you may find it difficult to build these into our designs.

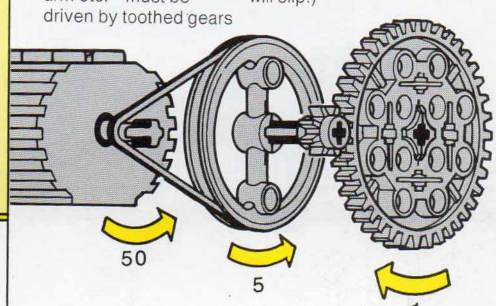
Robots at school

If you plan to build robots at school, it is possible to use the Educational LEGO Spares Service, which is cheaper than the ordinary spares service. For details see page 48.

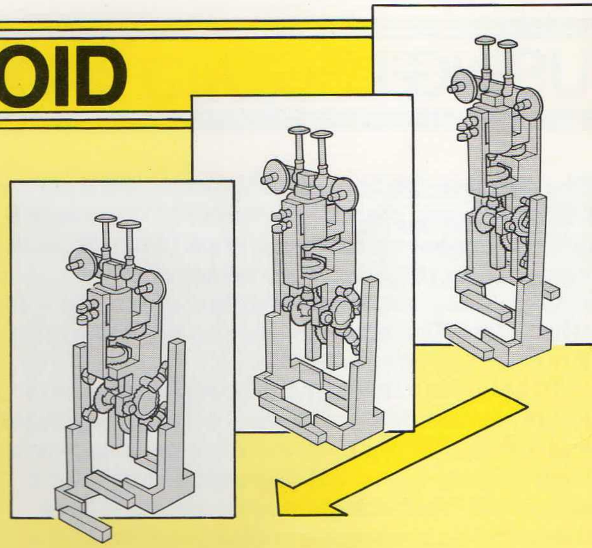
MOTORS AND GEARS

Most robots are simple to control if the moving parts operate slowly. You will have noticed in your LEGO motor instructions that to make things move slowly you need to 'gear down' the power from the motor. We found it best to take the initial power from the motor using a belt driving a large pulley wheel (LP). This causes the motor to do 10 revolutions while the pulley does only 1 – a gearing down of 10 to 1. If one then mounts an 8-tooth cog wheel on the pulley's shaft and uses this to drive a 40-tooth cog wheel, the pulley does 5 revolutions when the 40-tooth cog does only 1 (a gearing down of 40/8 or 5 to 1). The overall effect is that the motor does 50 revolutions while the 40-tooth cog – the output from the motor – does only 1. This is a reduction in speed of 50 to 1. You can see that by using different combinations of gear wheels you can get a series of different reductions.

(When using switch sensors note that the output – wheel, robot arm etc. – must be driven by toothed gears between the switch closing mechanism and the output. Drive belts will slip.)

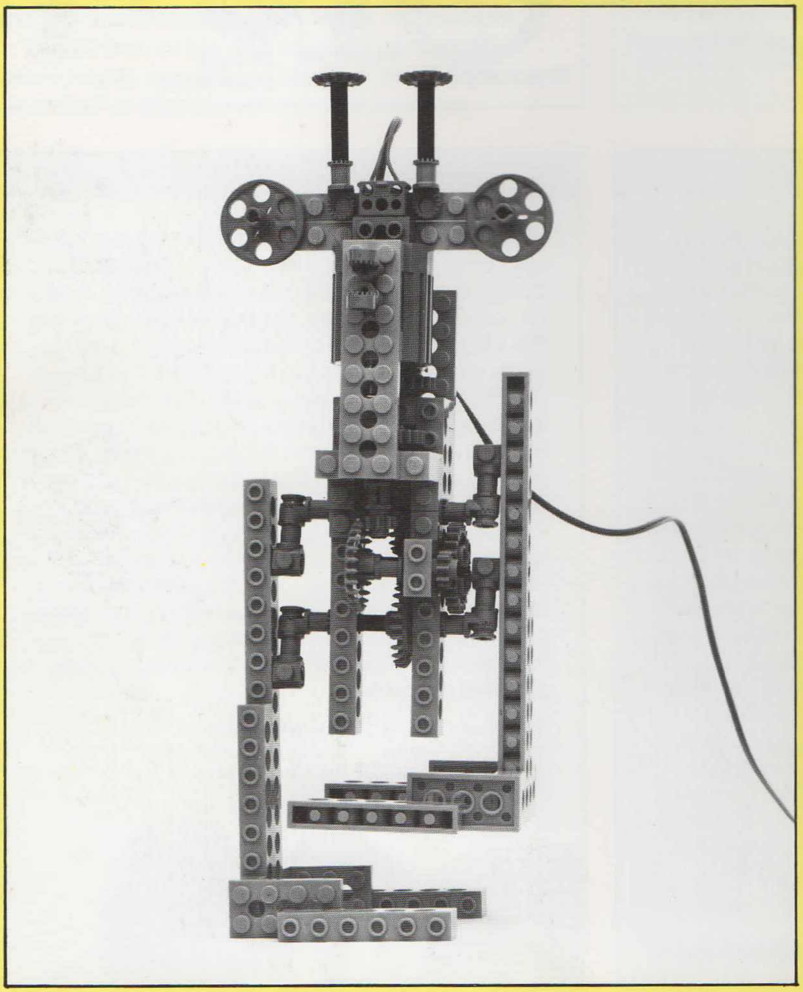


WALKING ANDROID



With just one motor and a few parts, you can build a robot that walks along on two legs. Our design is an android, a robot that looks and acts like a human. But as you will see if you try yourself, it is difficult to make a machine that walks exactly as we do.

This robot walks well because it can balance in any position. You can test this by connecting it to the battery box and stopping it in different positions. The android keeps upright and stable because its legs are rigid, its feet overlap one another, and its weight is always spread over an entire foot.



Up, one two, up, one two

The android walks by slowly raising one leg, moving it forwards, lowering it, putting it down, then repeating the process with the other leg and so on. If you have a friend who has built an android, you could have races!

◁ The android wired up to the computer. The robot will work best on a flat, level surface.

WHAT YOU NEED

All these pieces can be obtained from motor kit No. 8700 (or 880) and the supplementary kit No. 8710.

1 motor

Beams

2	4	6	8	12	16
8	6	4	2	2	2

Plates

1×2	1×3	1×4	1×8	2×4	2×8
6	3	2	2	4	3

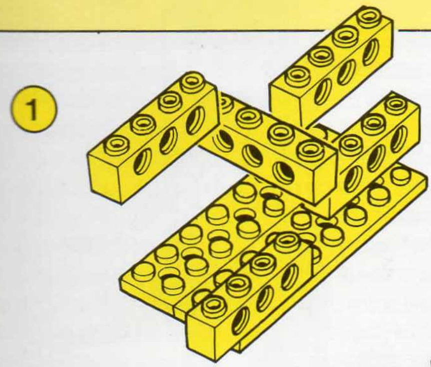
Axles

2	4	6	8
4	5	2	1

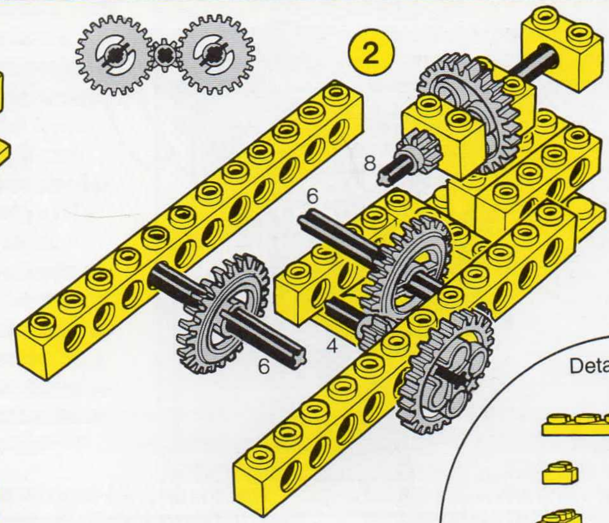
Gears

8	24	14C	24C	PUL
5	3	2	3	2

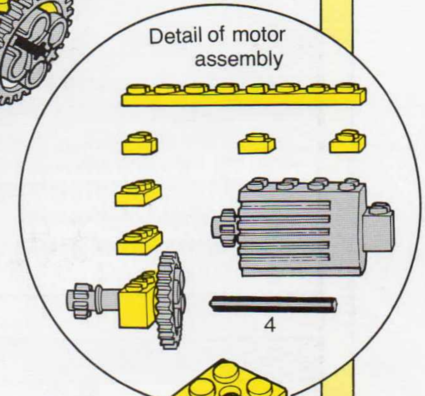
4	6	4	2	4	2

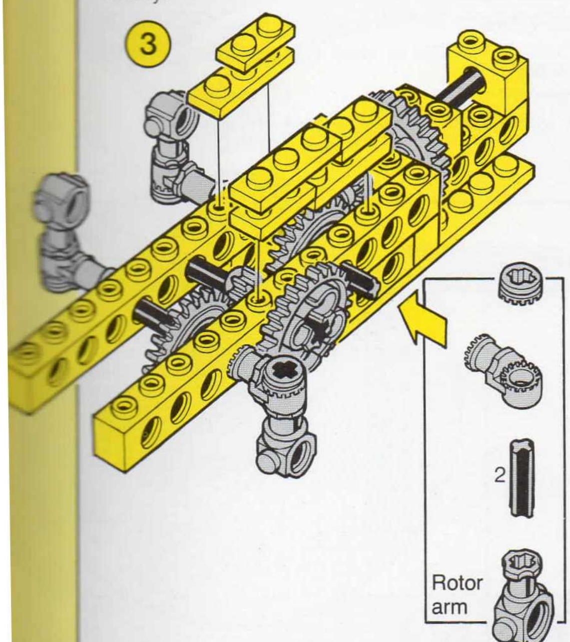


△ Start by building the android's back. Press the pieces together firmly.

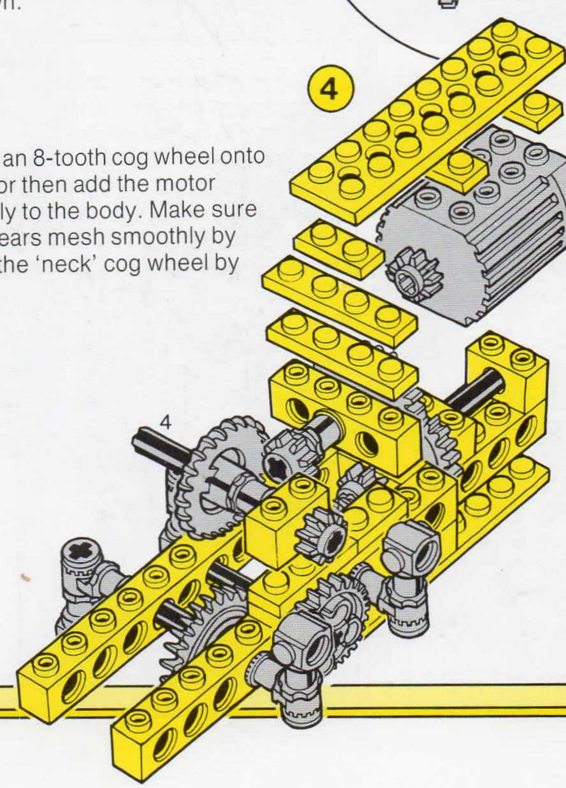


△ Add the waist section bearing the axles and gears. Make sure that the slots in the two 24C-gear wheels are set at the same angle, as shown.



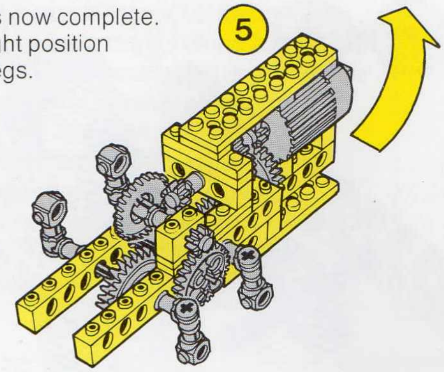


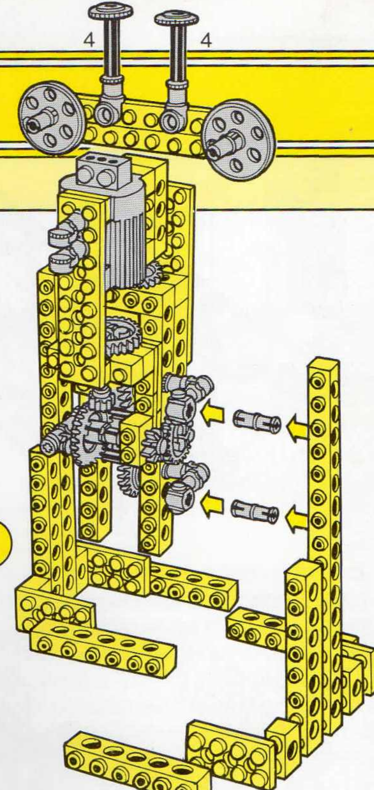
△ Assemble and fit the four leg rotor arms onto the 6-length axles. The arms should be at the same angle on each side of the body but opposite to one another on either side, as shown.



▷ Place an 8-tooth cog wheel onto the motor then add the motor assembly to the body. Make sure all the gears mesh smoothly by turning the 'neck' cog wheel by hand.

▷ The body unit is now complete. Lift it into the upright position ready to add the legs.





▷ Build each leg from the parts shown and attach it to the body using rotor arm pins. The pins fit into the third and seventh holes of the 16-unit beam of each leg.

Add a head of your own design – we've just used a structure with simple sensors and, as a finishing touch, have fitted a 'nose' below.

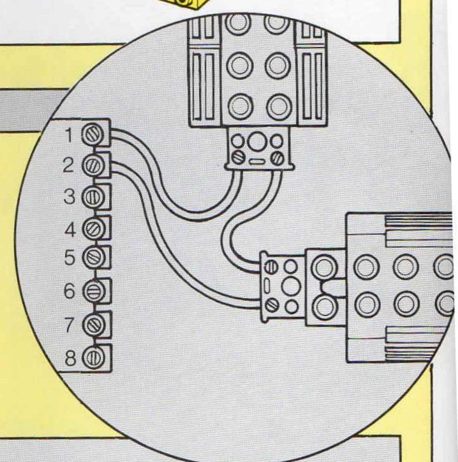
Test the android on a battery box. If it moves stiffly, loosen all the gears. Then connect it to the computer.

PROGRAMMING

Program 1 starts the android when a key is pressed and stops it when another key is pressed. (Any key can be used.) Program 2 starts the android in the same way but stops it after about 7 seconds. You can lengthen or shorten this period by increasing or decreasing the delay value in line 50. Try to adjust this number so that your android takes one slow step at a time. Both these programs can be used to control a motor in any of your own androids or indeed any robot.

```
10 REM ANDROID CONTROLLER
20 PRT=56577:POKE56579,63
30 GETA$:IFA$=""THEN30
40 POKEPRT,1:REM TURN ON MOTOR
50 GETA$:IFA$=""THEN50
60 POKEPRT,0:REM TURN OFF MOTOR
70 GOTO30
READY.
```

WIRING




```
10 REM ANDROID CONTROLLER
20 PRT=56577:POKE56579,63
30 GETA$:IFA$=""THEN30
40 POKEPRT,1:REM TURN ON MOTOR
50 FORTD=0TO5000:NEXT
60 POKEPRT,0:REM TURN OFF MOTOR
70 GOTO30
READY.
```

WHIRLY TURTLE

The walking android in Project 1 did not have a very robotic function except to show how to switch a motor on and off using the computer. In this project, the computer controls the motor in either direction – clockwise or anticlockwise. As a result, the floor robot, or turtle, can perform two functions. It can drive forwards and it can spin round. This is similar to standard turtles except that they can turn to the left or right and drive forwards or backwards.

Our turtle has a holder for a pen, which allows it to draw lines on a sheet of paper as it moves forwards. When the turtle spins, it pivots about the pen and so no line is drawn. Lay a large sheet of paper flat on a table and try and operate the turtle to draw different shapes and patterns. Draw a few lines with a blue pen, then change this for a red or green pen. Fibre-tip pens are the best to use.

WHAT YOU NEED

All the constructional pieces can be obtained from the motor kit No. 8700 (or 880) and the supplementary kit No. 8710.

Beams

2	4	6
8	4	2

Plates

1×2	1×3	1×4	1×8	2×4	2×8
6	4	4	2	4	4

Axles

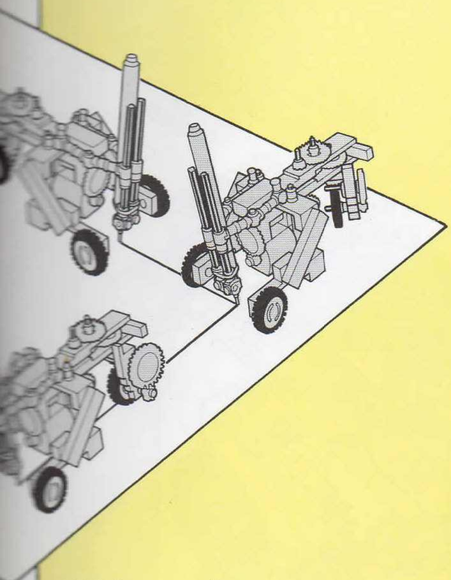
2	4	6	8	10
4	4	2	2	2

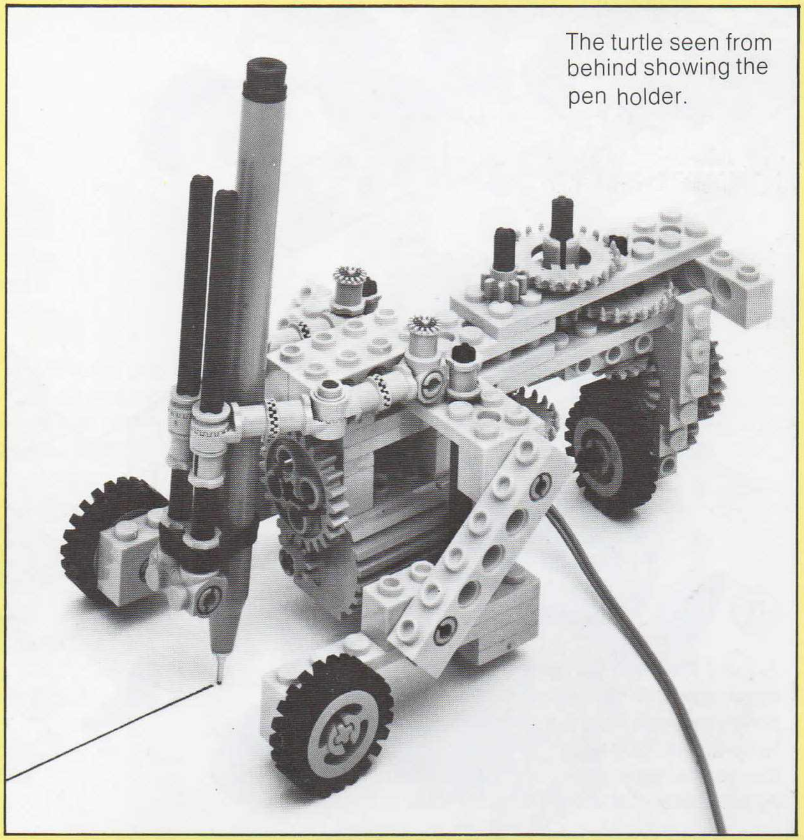
Gears

8	24	24C	14C
5	2	3	2

6	10	2	4	7	2	2

1 motor
3 small wheels



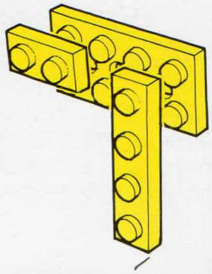


The turtle seen from behind showing the pen holder.

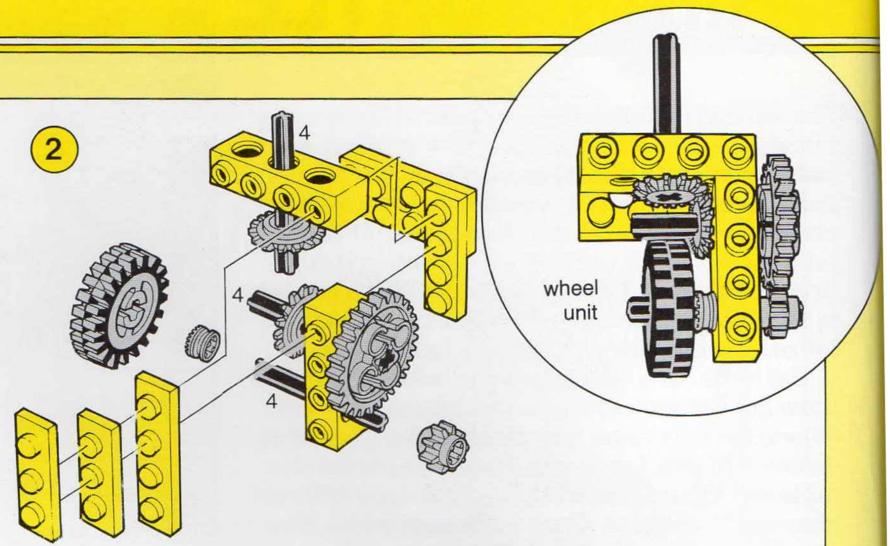
Whirling round

The motor drives the single front wheel. When the motor turns one way, the wheel unit rotates until the wheel points forwards, then the turtle moves off in a straight line. When the motor turns the other way, the unit rotates until it is pointing sideways, then the wheel turns causing the turtle to spin.

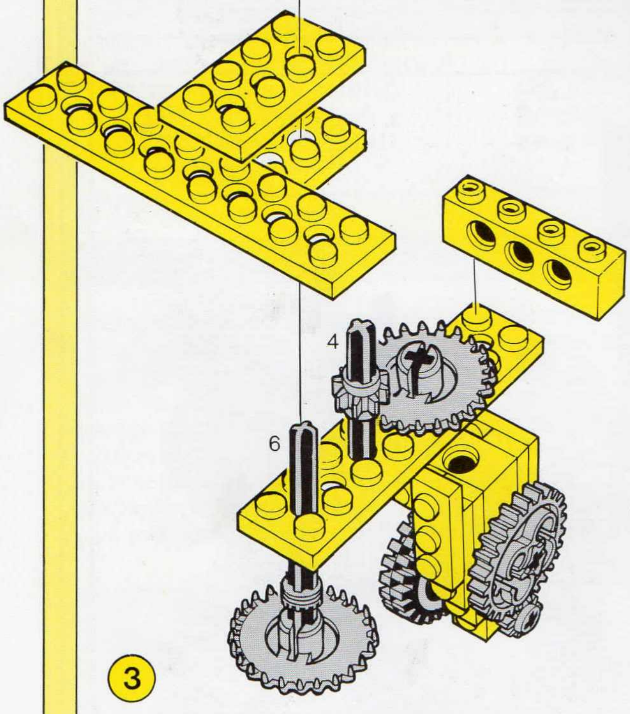
1



2

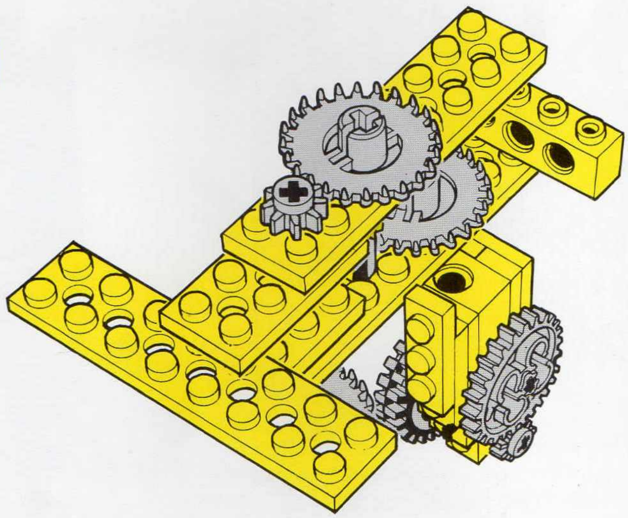


△ Construct the wheel unit as in steps 1 and 2. Make sure all the gears run freely by holding the unit at the top and rolling the wheel on a flat surface.



3

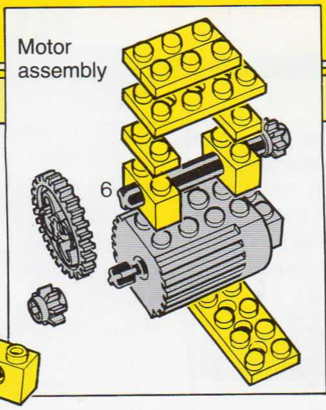
△ Build the main part of the turtle on the wheel unit. The wheel unit axle pokes through hole 3 of the 2×8 plate. Note the various layers of plates and the gear wheels above and below them. Again check that the gears run freely.



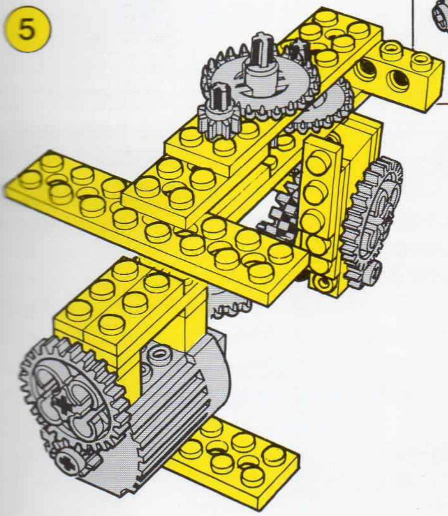
4

△ Add the uppermost plate and, on the axles, the cog and gear wheel as shown.

▽ Now add the motor assembly of plates and gears beneath the large cross-plate. (Note, this is the rear of the turtle.)

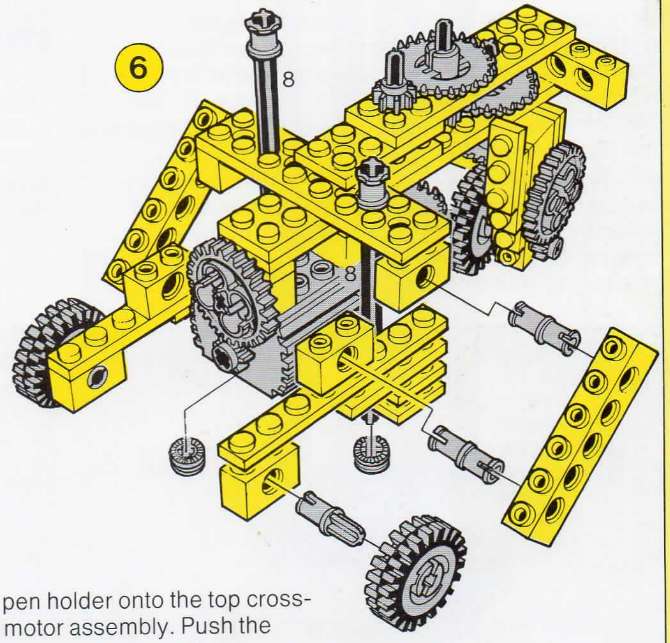


5



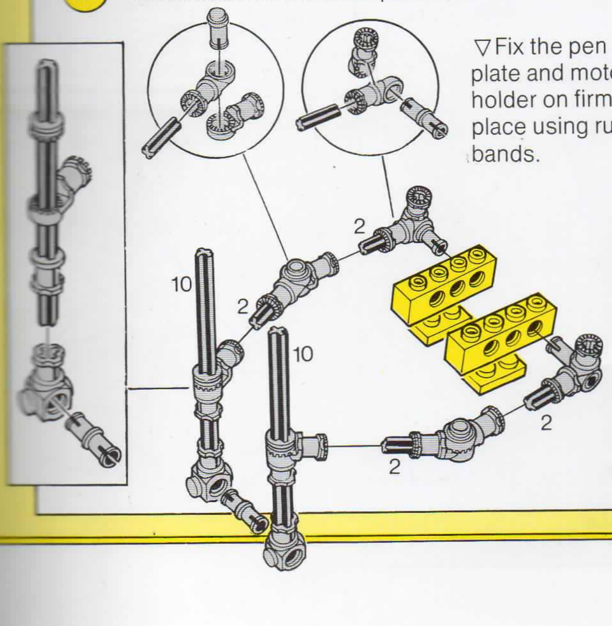
▽ Fit the rear wheels onto half-shafts and slot these into short beams on plates either side of the motor unit. Push two 8-unit axles with bushes at the top through the top and bottom cross-plates and fix half-bushes onto the bottom ends. The axles must not touch the ground.

6



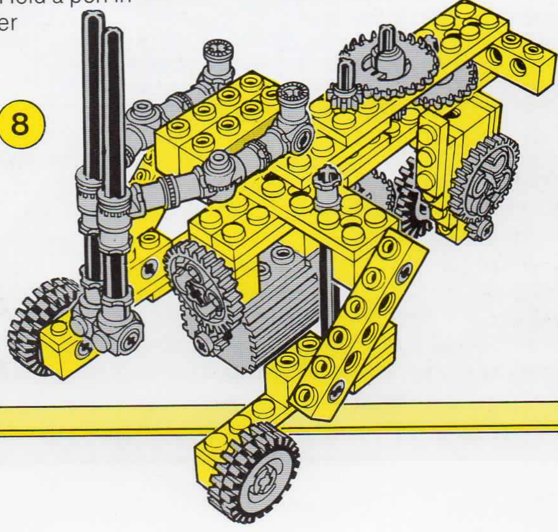
▽ Build the pen holder from toggle joints, bushes, 2- and 10-unit axles (see insets). Each arm of the holder is attached by a pin to one end of a short beam with a small plate below.

7



▽ Fix the pen holder onto the top cross-plate and motor assembly. Push the holder on firmly. Hold a pen in place using rubber bands.

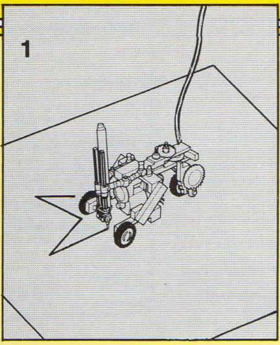
8

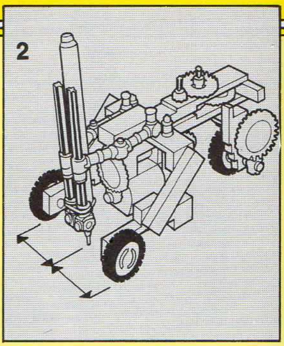


Operating hints and tips

1 As the turtle whirls about, the wire from the motor can sometimes get tangled in the wheels. Strap the wire to one of the bushes on top of the vehicle using a rubber band and hold the wire so that it hangs down to the turtle.

2 The pen tip should rest directly between the two rear wheels or when the turtle spins it will draw a little circle.



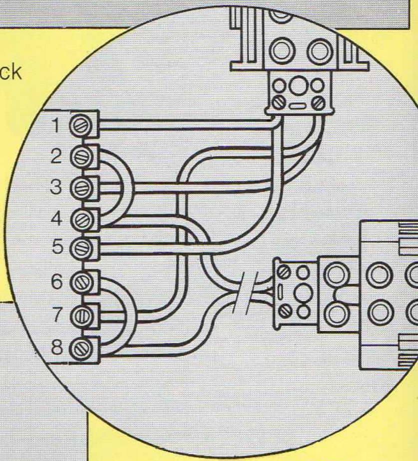


PROGRAMMING

The first program allows the whirly turtle to be controlled directly from the keyboard using the S key to spin and the F key to drive forwards. The Pattern Drawer program allows you to instruct the whirly turtle to perform several moves which it then repeats. The DATA program line contains pairs of command numbers. The first number tells the turtle how many units to spin, the second how far to drive forwards. The main program draws a pentagon. The series of moves are ended by putting 0, 0 at the end of the DATA line. We have provided some alternative DATA lines. Experiment by inserting your own command numbers.

WIRING

The interface block is being used as a pole reverser to drive the motor in two directions.



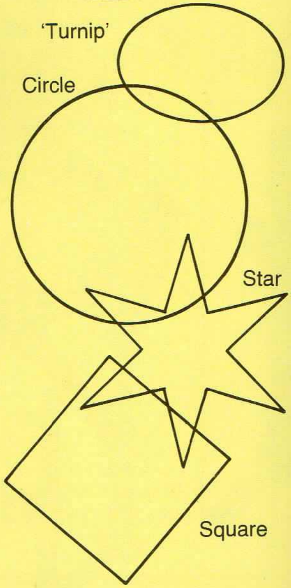

```
 10 REM WHIRLY TURTLE CONTROLLER
 20 PRT=56577:POKE56579,63
 30 NK=64:FK=21:SK=13
 40 A=PEEK(197)
 50 IFA=NKTHENPOKEPRT,0
 60 IFA=FKTHENPOKEPRT,6
 70 IFA=SKTHENPOKEPRT,9
 80 GOTO40
READY.

 10 REM WHIRLY TURTLE PATTERN DRAWER
 20 PRT=56577:POKE56579,63
 30 CT=350:SN=9:FW=6
 40 READA,D:IFA=0ANDD=0THENEND
 50 A=A*10:D=D*10:IFA=0THEN70
 60 POKEPRT,SN:FORPS=1TOCT+A:NEXT
 70 POKEPRT,FW:FORPS=1TOCT+D:NEXT
 80 POKEPRT,0:GOTO40
 100 DATA100,100,100,100,100,100,100,100,100,100,0,0
READY.



 100 DATA80,100,80,100,80,100,80,100,0,0
READY.

 100 DATA80,10,240,10,80,10,240,10,80,10,240,10,80,10,240,
     10,80,10,240,10,0,0
READY.
```



LIFT OPERATOR

Problem: How do we control a lift without a human lift operator? It is fairly simple to use a computer to drive a motor in either direction (see Whirly Turtle project), but how does the computer know when to stop the motor in a lift for each level or floor? Perhaps we could find the time it takes the lift car to travel between floors and use that as a guide? But this time varies depending on the load in the car – the greater the load the faster it descends. A better way is for the computer to 'know' how far the lift car has travelled. In this project we program it to do that.

You will have noticed that on your interface block there are connectors called switch inputs. If you connect a switch to one of these the computer can sense when the switch is in the on position or is 'closed'. We use a mechanism which closes a switch as the lift winch-axle rotates. The computer simply counts how many times the switch is closed and from this it finds out how far the motor has driven the lift car. The switch is made from foil, a band, paper clip and LEGO pieces.

Our design allows you to extend the lift shaft to any height and the lift car can carry small vehicles or toy people. So you could build this 'robot' as part of a model skyscraper or multi-storey car park.

WHAT YOU NEED

The motor and building pieces can be obtained from universal set No. 8050. The switch is home-made.

2	4	6	8	12	16	
4	2	4	5	4	6	Beams

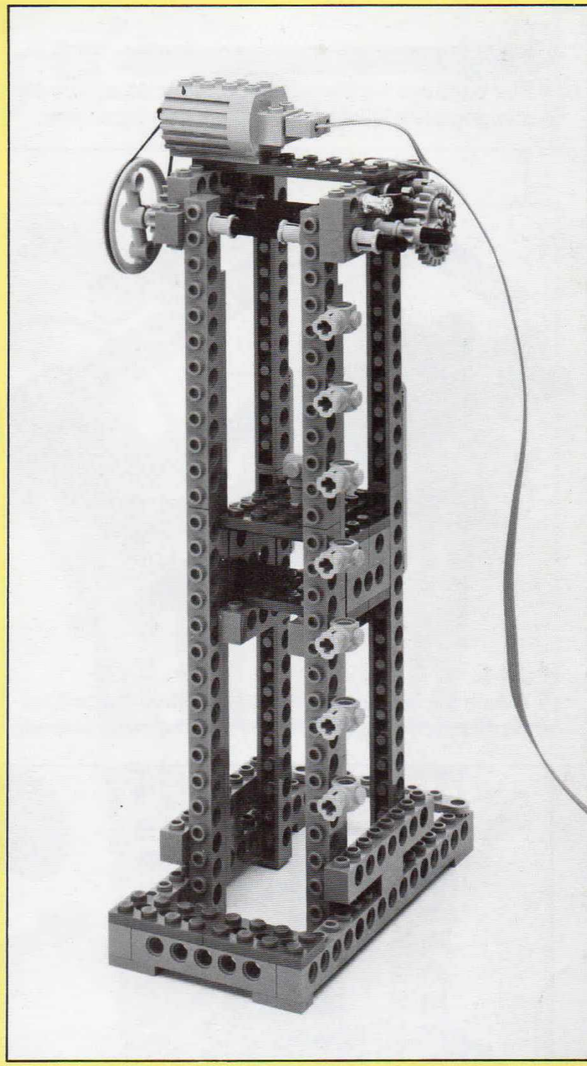
1×2	1×6	2×4	2×6	2×8	4×6	
8	8	2	4	1	1	Plates

10	12	
1	2	Axles

8	24	LP	
2	2	1	Gears

10	7	8	7

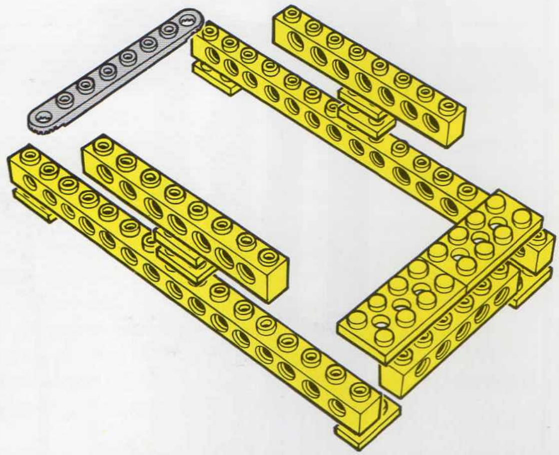
1 motor, 3 1×8 steering plates, 2 drive belts, 30cm length of string, 3 2.5cm-squares of cooking foil, a paper clip, 50cm length of 4-way ribbon cable.



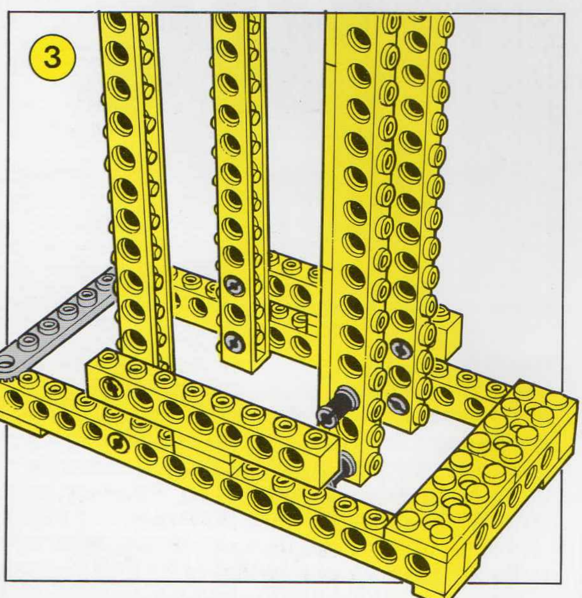
Fourth floor–going up
The lift car is winched up on a piece of string wound round the axle at the top of the shaft. The motor, which is geared down, drives the axle back and forth. As the car goes up or down, the switch mechanism closes the switch six times between each level. The switch sends pulses to the computer, which is set up to stop the car after 6, 12, 18, 24 pulses and so on. So to go from the 4th to the 6th floor the computer counts 2 × 6 = 12 pulses.

1

∇ The base is a simple assembly of plates and beams. A steering plate is used at the back as a cross-link.

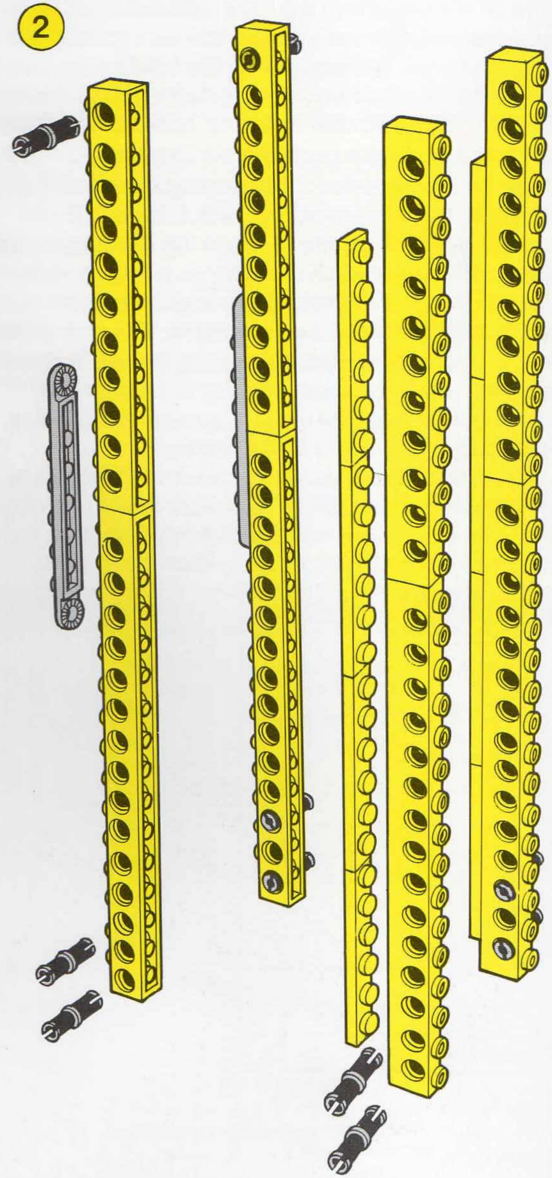


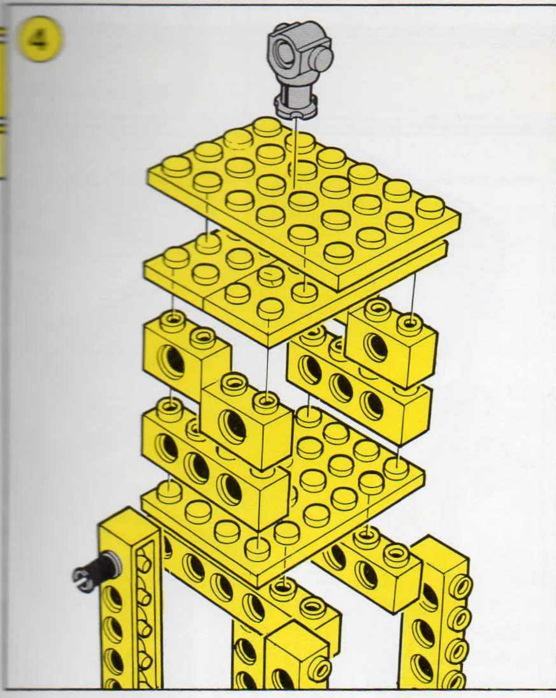
∇ Attach the supports to the base. (Only the bottom ends of supports are shown.) Push the pegs in firmly.



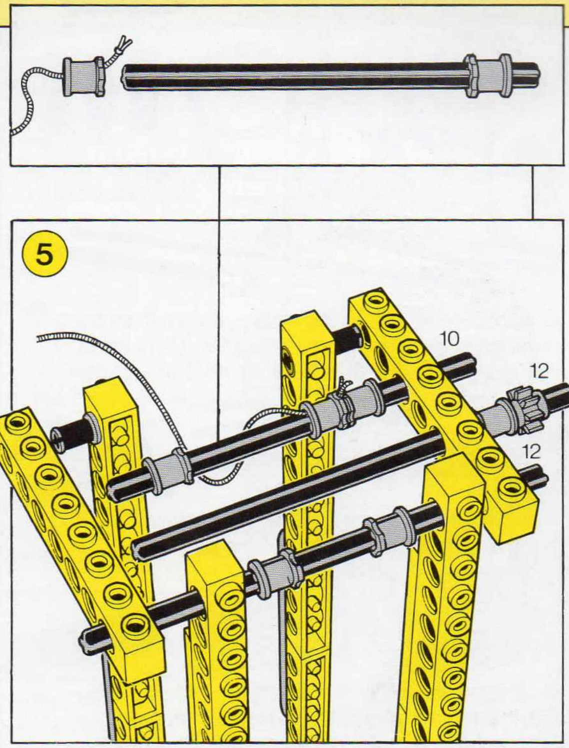
∇ Build separately the four main upright supports. Fit pegs top and bottom as shown.

2

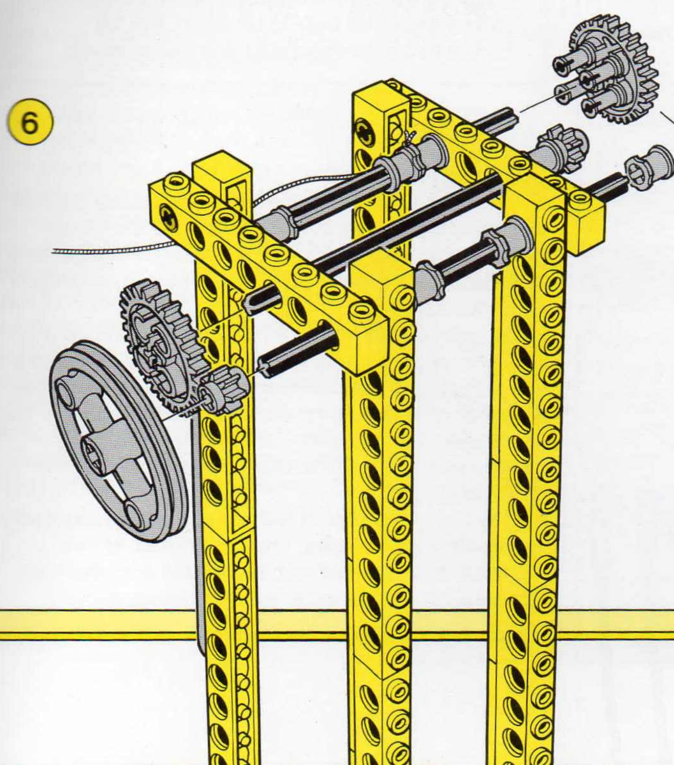


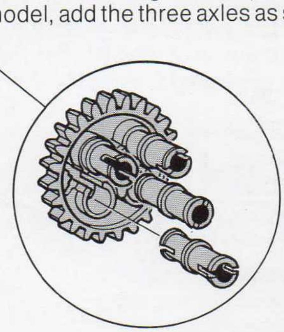


△ Build the lift car from beams and plates. If you wish, make the car taller by adding an extra layer or two of beams. Allow the car to drop into the top of the lift shaft. The car should slide freely between the supports. If it does not, check you have followed the instructions correctly. Do not proceed until you have solved the problem.



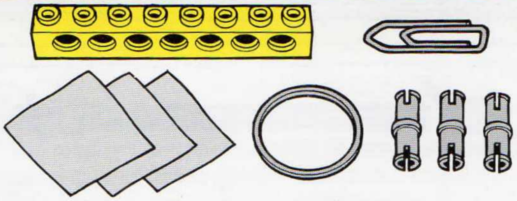
△ Attach the piece of string to the winching axle by poking it through a bush and then sliding the bush onto the axle. Working at the top of the model, add the three axles as shown.



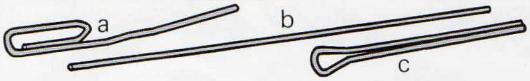


◁ The gear wheels and pulley wheel fit onto the ends of the axles. If the wheels do not turn freely, ease them slightly.

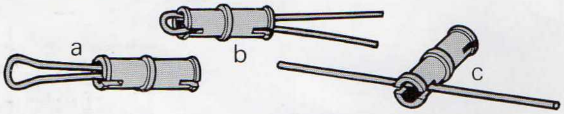
MAKING THE SWITCH



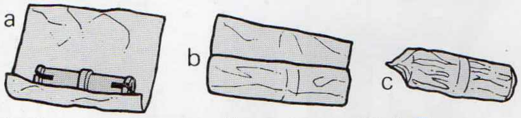
1 Straighten out the paper clip with your fingers or, better still, with pliers.



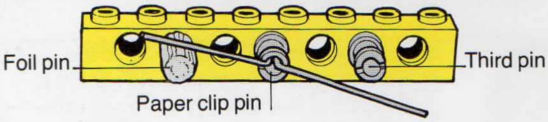
2 Bend the loop into the shape shown then poke it through a pin with a hole through it. The loop at the end of the clip should be about 3mm across.



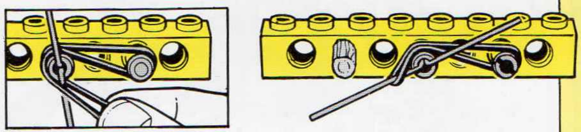
3 Bend back the ends of the clip outwards and back and through the slots in the pin.



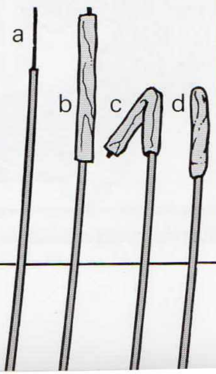
4 Wrap another pin with a square of foil. Then push the three pins into the holes indicated.

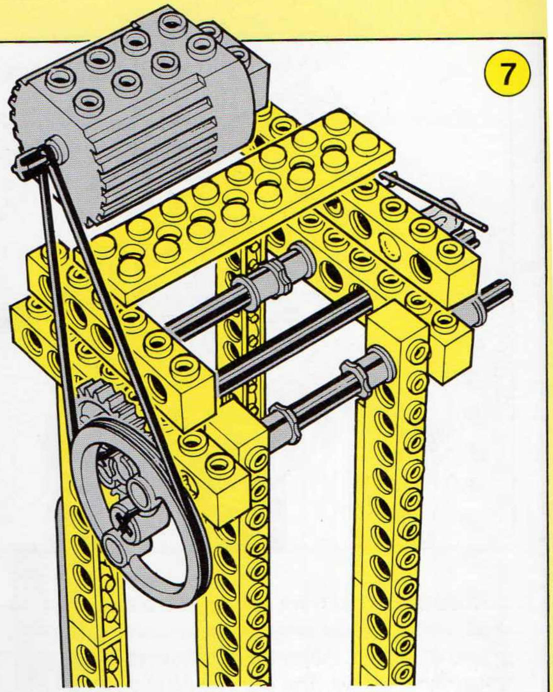


5 Hook the belt onto the third pin. Pass it behind the paper clip, pull it down round the middle pin, then pass the loop of the belt over the lower end of the clip. (See step 8, right, for final view.)



6 Make the plugs. Strip one end of each of the two wires by 1cm. Wrap a square of foil round the end of each wire so that 2mm of bare wire poke out. Fold the foil over and roll the end of each wire to form a short plug. Attach the switch to the model and fit the plugs.





△ Add the switch beam, plain beam, cross-plate and motor. (See left for details of how to make the switch.) The paper clip on the switch should be pulled over to be above the foil. Fix the belt between motor and pulley wheel.

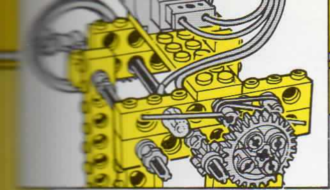
Operating hints and tips

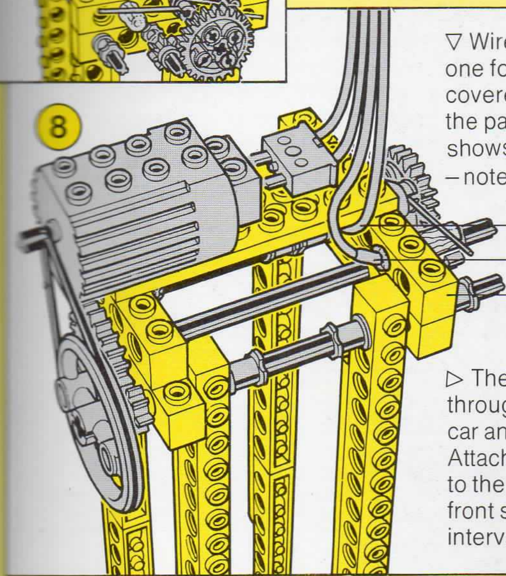
1 Make sure that the four main supports are pushed well together but not in such a way that the lift car gets caught. Also make sure the string is not too long and there are no loose ends which could get tangled in the works.

2 Sometimes the lift car, when empty, does not descend very smoothly. To help it down add some extra weight to it – a piece of plasticine or some LEGO bricks, say.

3 We've used a pole reverser on the battery box. This is not essential, but if the car starts travelling too far, you can override the computer and turn off the motor as necessary.

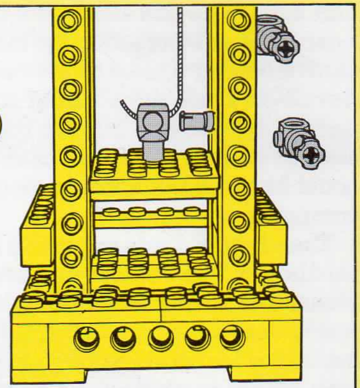
4 The paper clip should flick up and down repeatedly as the drive axle rotates. The clip should, at one moment, be in contact with the foil pin and, the next moment, be positioned a few millimetres above it.

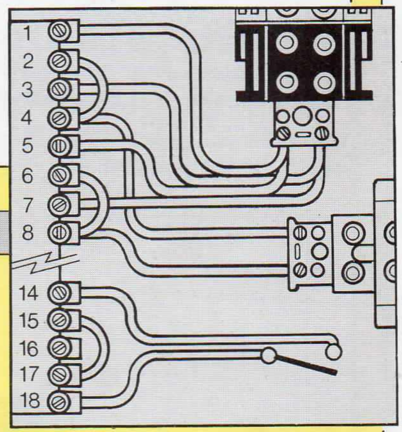




▽ Wire up the motor and switch, with one foil plug pushed into the foil-covered pin and the other plug into the paper clip pin. The inset shows the switch from the far side – note the position of the paper clip.

▷ The loose end of the string goes through the toggle on top of the car and is secured with a pin. Attach toggles (or small plates) to the outside edge of one of the front supports, at 2-hole intervals, to indicate each level.





PROGRAMMING

Our program will move the lift car to any level. When you RUN it, the computer assumes the lift car is at Level 1 (ground floor). If it is not, connect the motor to the battery box and move the car to the bottom of the shaft. Then reconnect the motor to the interface. The program will ask you 'WHICH LEVEL?' Type in a number from 1 to 7 and press RETURN. The car will move to that level and automatically stop. It does this by driving the motor in the correct direction and then counting the pulses from the switch. (If you wish to use a taller lift shaft you will have to change the value 7 in line 50 to the number of levels in your model.) The program will then again ask the question. Note that if you type in, say, '2', the lift car will move to level 2 and not just go up or down two floors.

WIRING

The wiring is similar to that for the Whirly Turtle project except there is also the home-made switch and a pole reverser is fitted to the battery box.

```
 10 REM**LIFT CONTROLLER**
 20 PRT=56577:POKE56579,63
 30 UP=9:DW=6:SP=0:CL=1
 40 INPUT"WHICH LEVEL 1-7";NL
 50 IFNL>7ORNL<1ORNL=CLTHEN40
 60 TD=NL-CL
 70 IFSGN(TD)=-1THENPOKEPRT,DW
 80 IFSGN(TD)=1THENPOKEPRT,UP
 90 FORI=1TOABS(TD)*6
 100 IF(PEEK(PRT)AND128)=0THEN100
 110 IF(PEEK(PRT)AND128)=128THEN110
 120 NEXTI:CL=NL
 130 POKEPRT,SP:GOTO40
READY.
```

CARD READER

This simple device allows the computer to identify a cardholder by reading the number encoded on a card. It could be used as the basis for a security device on a safe or on a door so that only certain people are allowed into the room. Or it could be used as a card reader in a cash dispenser, and you could have a bank account record in your computer.

The identity cards are made from ordinary cardboard. The number on each card is encoded in binary as a row of punched holes. You may have encountered binary numbers in your computer's manual or learnt about them at school or computer club. Binary numbers use only two figures, or digits, 0 and 1, to represent numbers. So 1 in binary is 1, 2 is 10, 3 is 11, 4 is 100 and so on. In our machine cards, a hole punched through the card represents 1 and an absence of hole (solid card) represents a 0.

As a card feeds through the reader, its code number is read by the computer.

WHAT YOU NEED

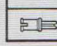
The majority of pieces can be obtained from motor starter set No. 8050.

2	4	6	8	12	16	
4	2	2	2	4	4	Beams

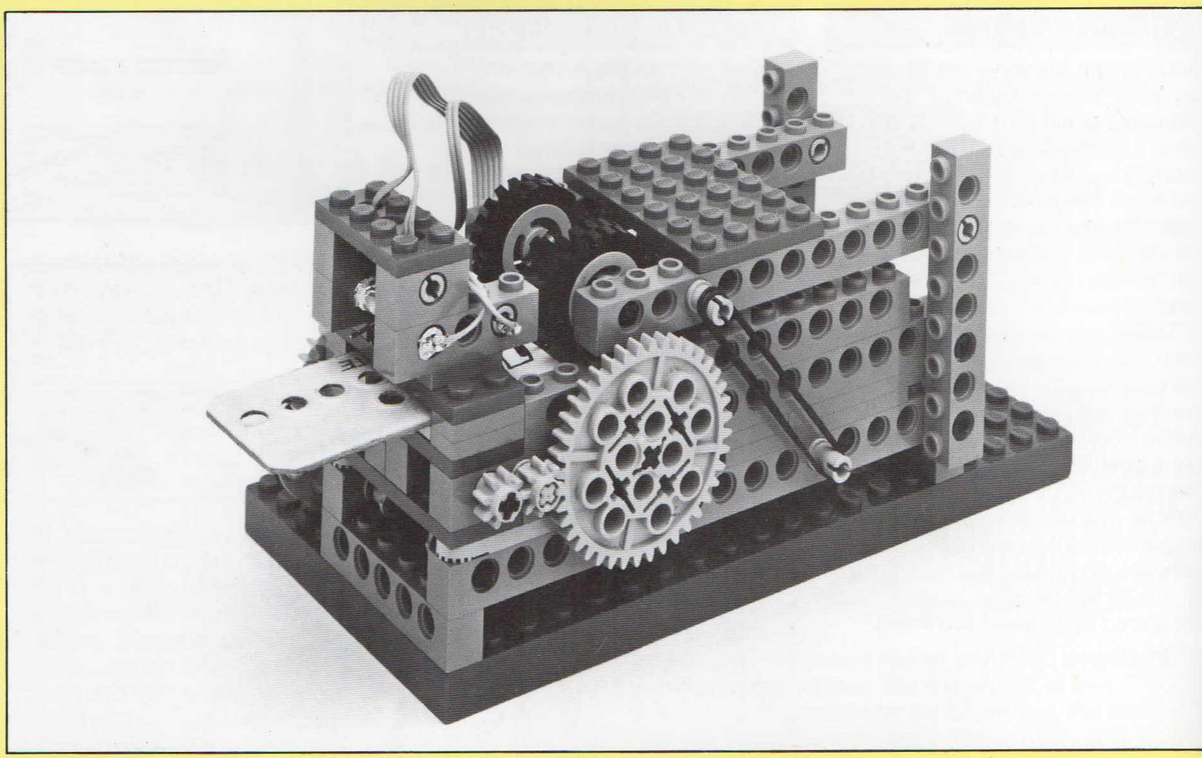
1×2	1×6	2×4	2×6	4×6	
8	6	1	3	1	Plates

6	8	
2	3	Axles

8	24	40	SP	
3	1	1	1	Gears

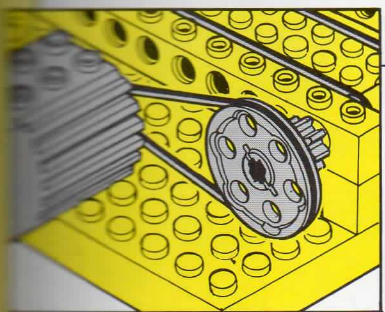
1	10	4	4
			

1 motor, 4 small wheels, 1 10×20 base plate, 6 drive belts, 1 1×8 steering plate, 2 1×2 smooth plates, 50cm length of 6-way ribbon cable, 6 2.5cm squares of cooking foil, 2 paper clips.



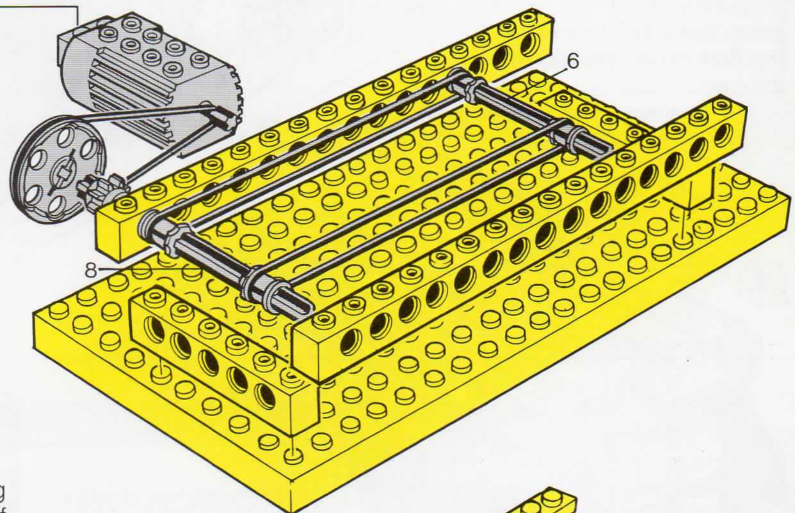
This is a security check!

Provided the card is put in the correct way round, the right-hand switch in the reader is opened and the computer turns the motor on. The card is drawn in by the roller wheels and the holes pass under the switches. The left-hand row of holes allows the computer to sense how far the card has moved through the machine. The right-hand row of holes contains the binary encoded number. When the computer has finished reading the code, the motor is left running for a short while and the card is fed out of the reader. The motor is then turned off.



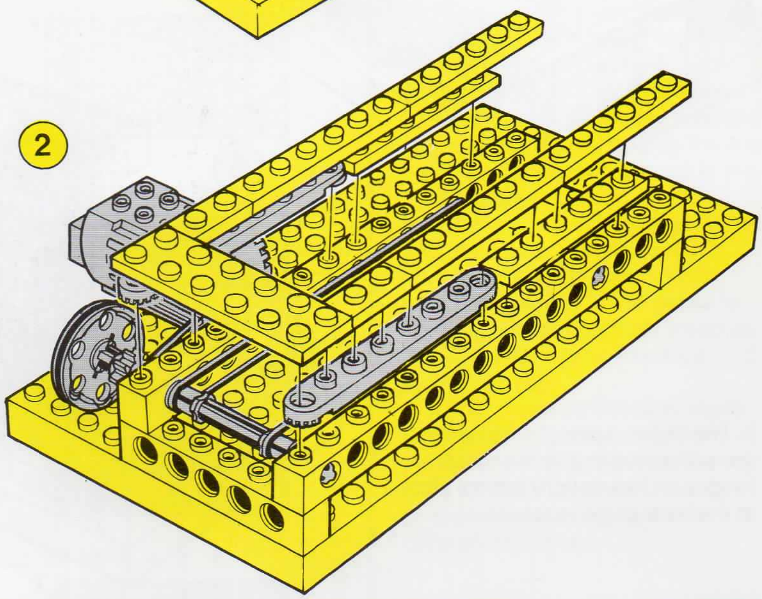
Detail of motor attachment to base plate

1



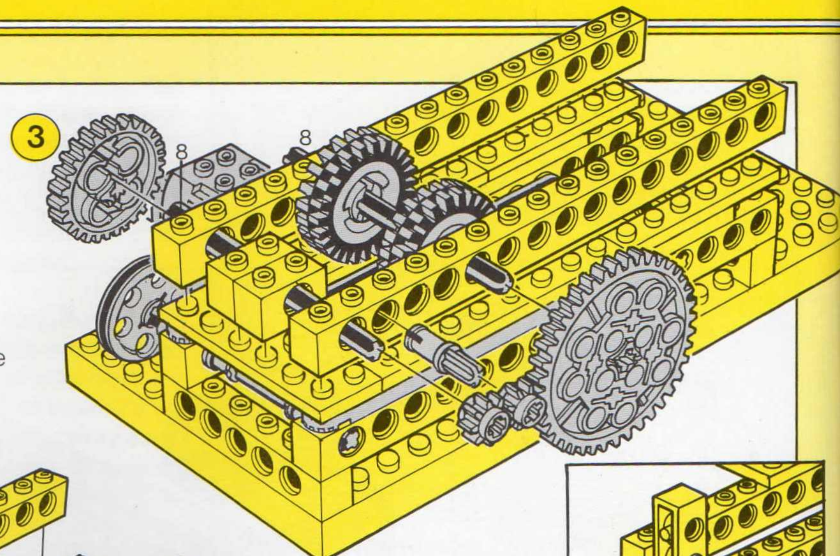
▷ Fix the two cross-beams on the base blate. Push the bushes on the axles, slide the axles into one of the long beams and place the belts over the bushes. Add the other long beam and fix the assembly on top of the cross-beams. Add the gear wheel, pulley, motor and drive belt.

2

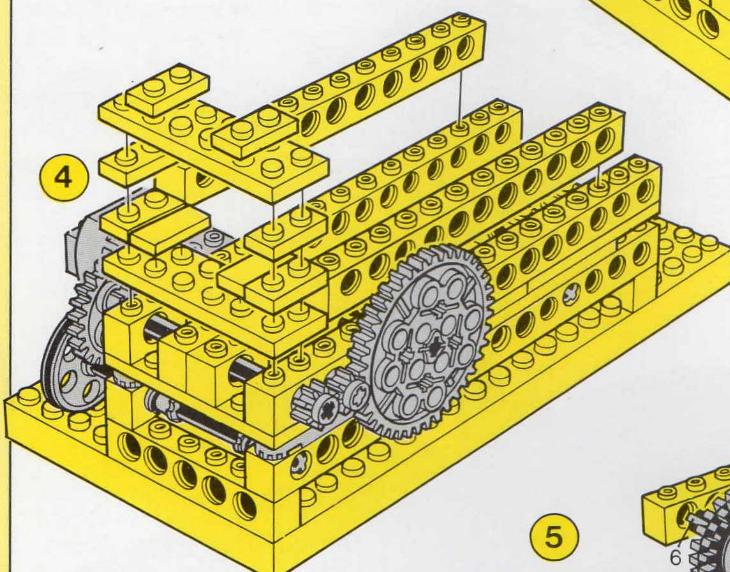


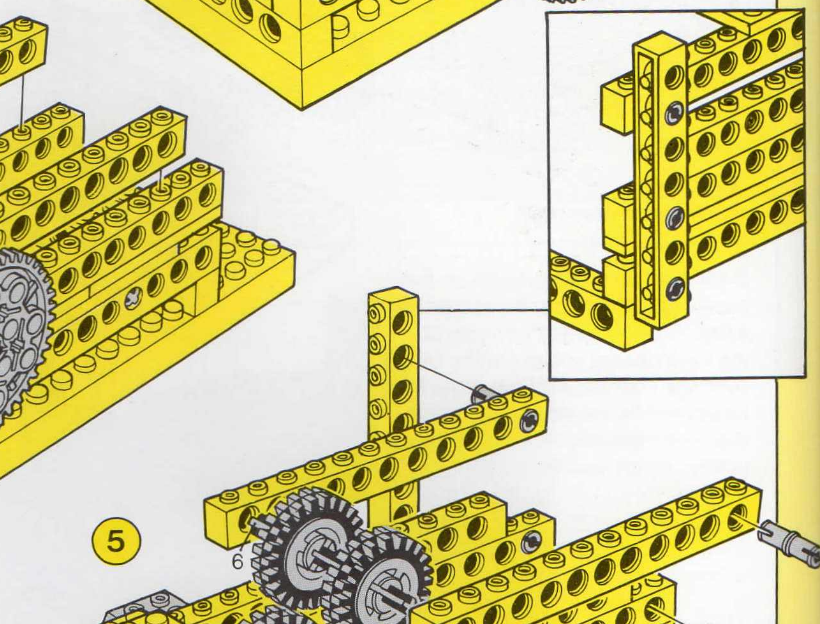
▷ Add the steering plates and other plates as shown.

▷ Make up the next level of beams, axles, gears and small wheels, and place it in position on the first level.

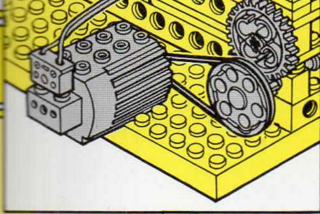


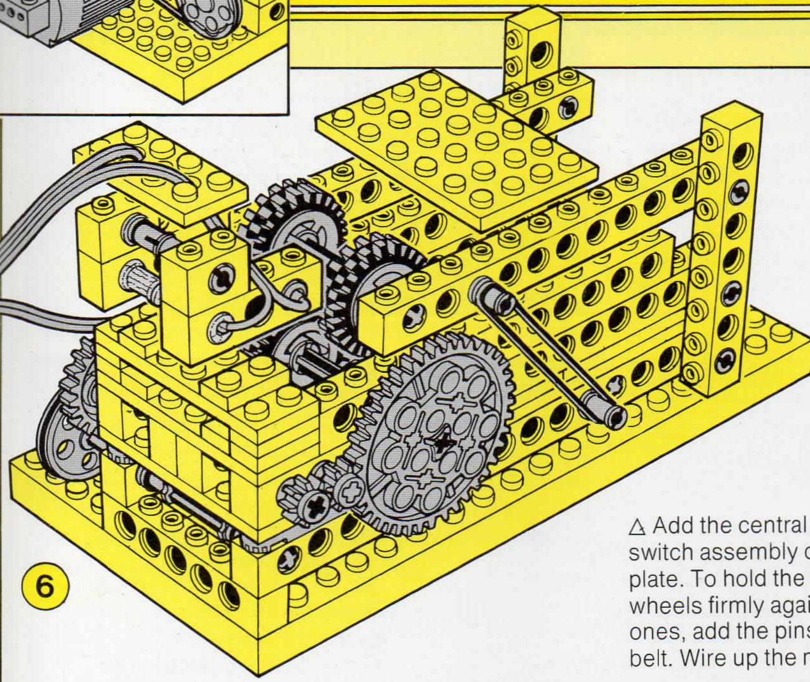
▽ Add the two 12-unit beams and the plates that at the front of the machine make up the slot for the card.

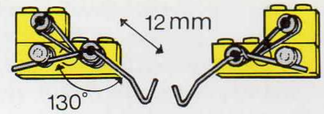




▷ The second pair of small wheels are supported in a frame which hinges on two upright beams placed at the back of the machine.







△ The switches are of a different design to switches used elsewhere in the book. The arms of the paper clips should be set to the angle shown.

For details of how to make the switches see page 18

△ Add the central top plate and the switch assembly of beams and cross-plate. To hold the top card-roller wheels firmly against the bottom ones, add the pins and tension belt. Wire up the model.

MAKING THE CARDS

You will need a sheet of stiff card, a hole punch, scissors, ruler and pencil.

1 From the sheet of card, cut some pieces the size of a LEGO 4×8 plate (some 64×32mm).

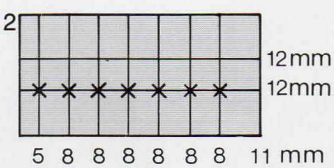
2 Mark the back of the card as shown. Where you have made crosses, punch a hole in the card. The centre of the holes should be over the centre of the crosses. Cut out the right-hand end hole to make a slot.

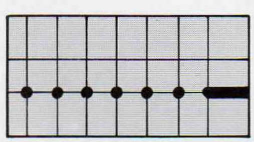
3 Number the other points on the card as indicated. To give the card its identity, punch holes at the correct points. The card's number is given by all the punched numbers added together. (In a binary number, values double from right to left, 1, 2, 4, 8 etc.)

To make the card run more easily in the reader, snip off the corners with scissors.

1

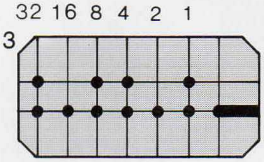
2





32 16 8 4 2 1

3



32 + 8 + 4 + 1 = 45

Operating hints and tips

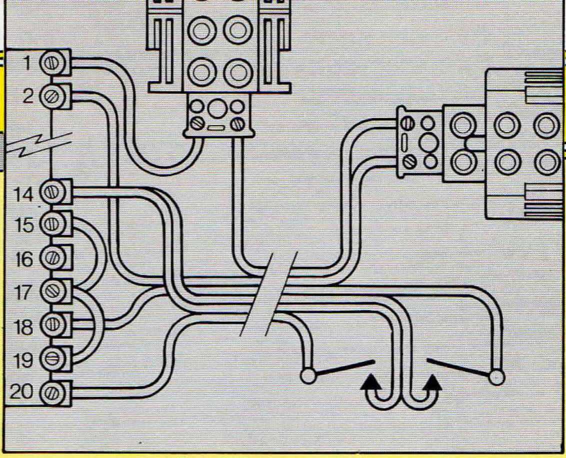
1 If the card reader motor does not turn off after feeding a card through, try changing the angle between the paper clip arms in the *left-hand* switch. The switch should make definite clicks as the card passes.

2 If the machine does not read the card correctly, try changing the angle between the paper clip arms in the *right-hand* switch.

3 If the motor does not start when you poke a card into the slot, check that only the right-hand switch is opening. If the switch is opening, check you have wired up the switches, motor, interface, and so on, correctly.

4 If your cards do not pass easily through the machine, trim them carefully with scissors or a Stanley knife so that they are a loose fit, or experiment using card of a different thickness.

WIRING



Type in the program and SAVE it. When you RUN the program, the screen should clear and the computer print a message asking you to insert a card. If a card is inserted the correct way round, the motor will be turned on, the card will travel through the machine, and its encoded number be read by the computer. If the computer knows a password for the card owner, it will ask for it. Then type in the password for the computer to tell you that you may enter. If it does not know the password, it will ask you to enter your name and password.

PROGRAMMING

```
10 REM CARD READER SECURITY
20 PRT=56577:POKE56579,63
30 M=1:S=0:T1=128:T2=64
40 DIMN$(63),P$(63)
50 D$="■■■■■■■■■■■■■"
100 REM WAIT FOR CARD
110 PRINTD$;"       PLEASE INSERT YOUR CARD"
120 IF(PEEK(PRT)ANDT1)=0THEN120
130 IF(PEEK(PRT)ANDT2)=T2THEN120
140 POKEPRT,M
150 IF(PEEK(PRT)ANDT2)=0THEN150
160 REM READ CARD
170 X=0:FORI=0TO5
180 IF(PEEK(PRT)ANDT2)=T2THEN180
190 IF(PEEK(PRT)ANDT1)=0THENX=X+2↑I:GOTO210
200 IF(PEEK(PRT)ANDT2)=0THEN190
210 IF(PEEK(PRT)ANDT2)=0THEN210
220 NEXTI:FORTD=1TO1500:NEXTTD
230 POKEPRT,S
300 REM TEST FOR PASSWORDS
310 IFP$(X)=""THEN400
320 PRINTD$;:INPUT"    WHAT IS YOUR PASSWORD";PW$
330 IFPW$<>P$(X)THEN500
340 PRINTD$;"          YOU MAY ENTER ";N$(X)
350 GOSUB1000:GOTO100
400 REM ADD NEW PERSON TO FILE
410 PRINTD$;"YOU ARE NOT ON FILE AS A CARD HOLDER"
420 PRINT" WE WOULD LIKE YOUR NAME AND A PASSWORD"
430 INPUT"YOUR NAME        ";N$(X)
440 INPUT"YOUR PASSWORD    ";P$(X)
450 PRINT"      MAKE A NOTE OF YOUR PASSWORD"
460 PRINT"  DO NOT LET ANYONE KNOW YOUR PASSWORD"
470 GOSUB1000:GOTO100
500 REM WRONG PASSWORD ENTERED
510 PRINTD$;"   THAT IS NOT THE CORRECT PASSWORD"
520 PRINT"                TRY AGAIN"
530 GOSUB1000:GOTO100
1000 REM WAIT FOR SPACE BAR
1010 PRINT"      PRESS SPACE TO CONTINUE"
1020 GETA$:IFA$<>" "THEN1020
1030 RETURN
READY.
```

If you type the wrong password, the computer will tell you this and ask you to try again. Passwords can be any word, code or number you like. Once you have entered your name and password into the computer, it will not forget them unless you stop the program or turn off the computer.

MINI ARM

Here is a simplified industrial robot arm for you to build. Industrial robots do simple jobs that people find boring and repetitive, such as packing and sorting. They can also perform dangerous tasks, like mixing chemicals and handling radioactive materials. Our robot arm can pick up small objects, swivel round on its base, then place them down anywhere within its reach.

Our design uses two motors to do the job that in larger robots is normally performed by three motors. One motor drives the arm round on the base, while the other motor raises and lowers the arm and operates the gripper. The base and arm – often called the waist and shoulder – both have built-in switch sensors. These, together with the program, allow you to 'teach' the arm a set of moves which it can then repeat completely under the control of the computer.

The robot arm, with a small cube held in its gripper, swivelling round on its base.

WHAT YOU NEED

All pieces can be obtained from motor starter set No. 8050, supplementary set No. 8710 and motor unit No. 1175 (from the spares service).

2	4	6	8	12	16	
4	5	7	7	3	3	Beams

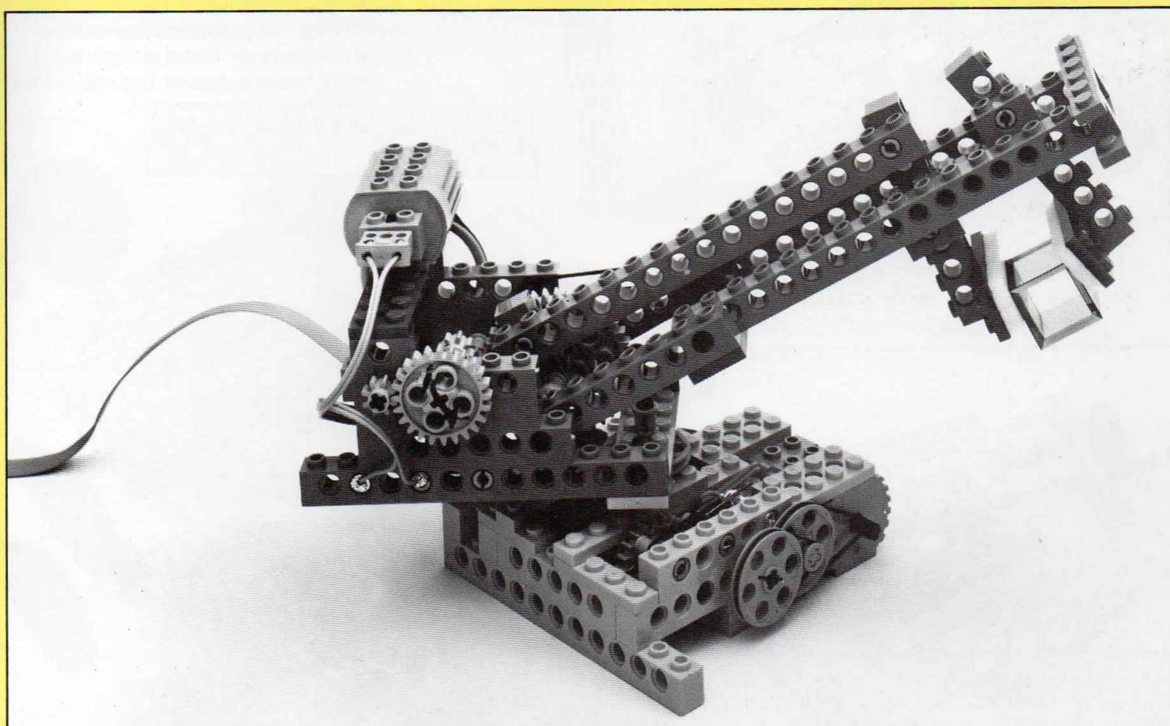
1×2	1×3	1×4	1×6	1×8	2×4	2×6	2×8	
5	2	3	8	2	5	1	3	Plates

8	24	24C	40	SP	LP	
6	4	1	1	3	1	Gears

4	6	8	10	
3	1	2	2	Axles

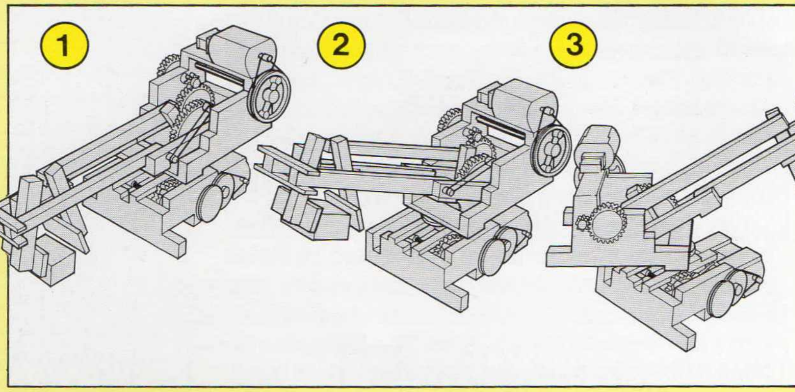
2	5	5	18	2

2 motors, 3 small drive belts, 3 large drive belts, 50cm length of 8-way ribbon cable.
For the switches – belts, cooking foil, paper clips.

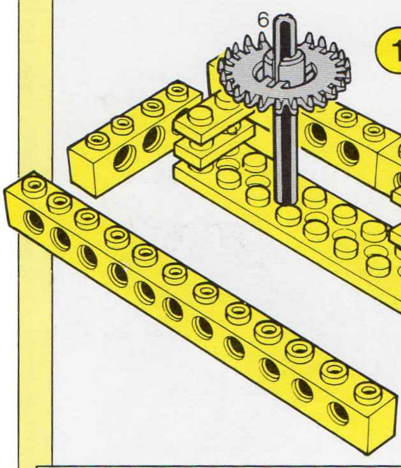


Up, round, down; up, round, down
As the arm turns at the 'waist' it can stop at any one of 36 set positions, and as it moves up and down at the 'shoulder' there are 7 or 8 stop positions from bottom to top.

The gripper closes its fingers on an object. When the fingers cannot close any more, the arm raises up to the chosen position. Then the arm turns, lowers and, as the object touches the tabletop, the grippers open. This is a 'pick-and-place' action.



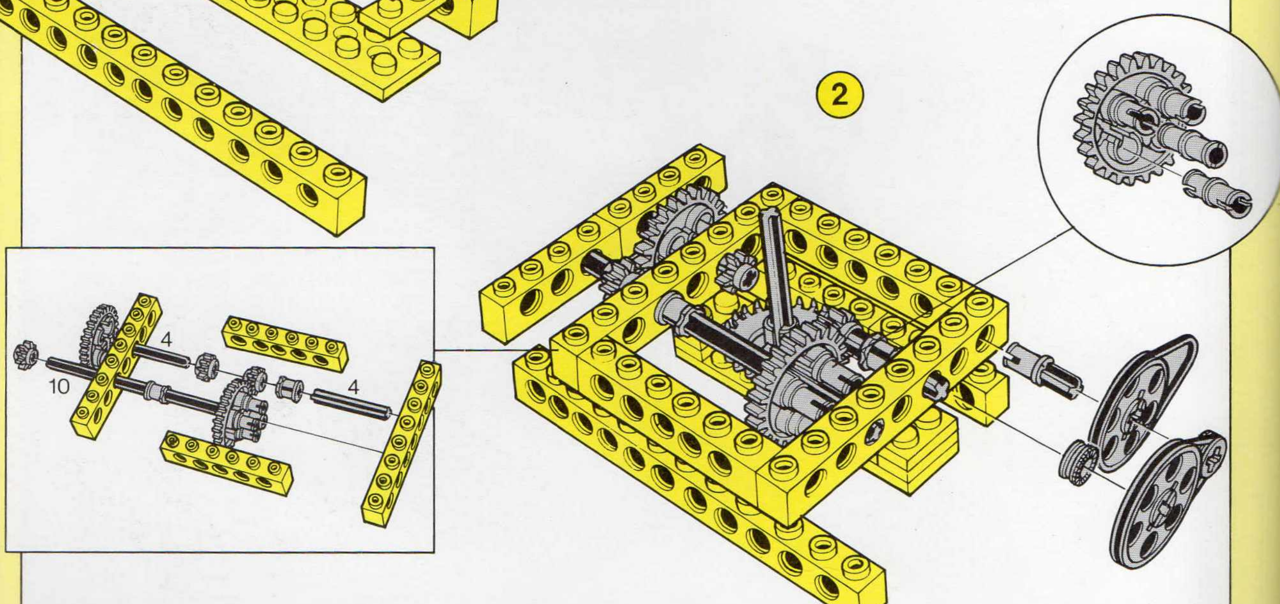
THE BASE



1 ▽ Construct the bottom part of the base with beams and plates from the supplementary set. Add the upright axle and the gear wheel.

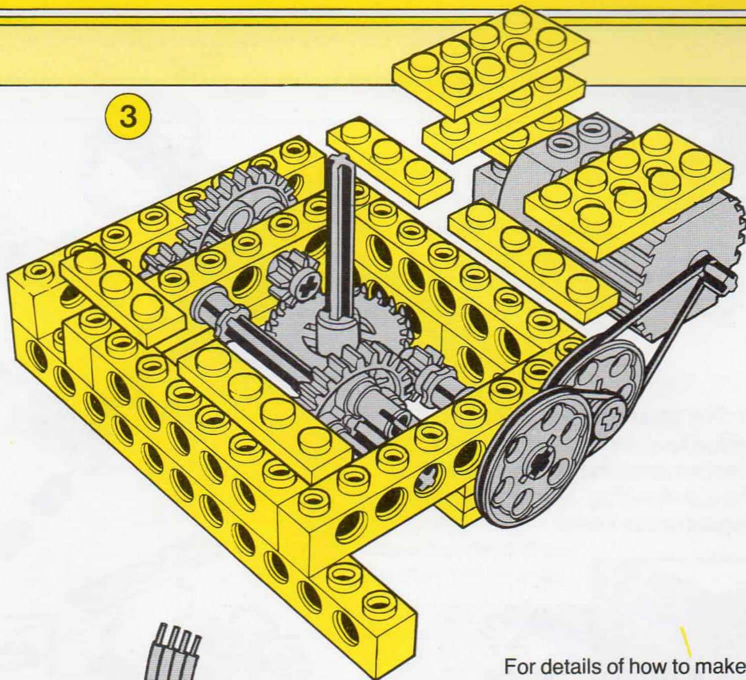
▽ Assemble and add the next level, with gear wheels, axles and so on, making sure you have used the correct pieces (see insets for details). Place small drive belts on the pulleys. Make sure the gears mesh properly and move freely.

2



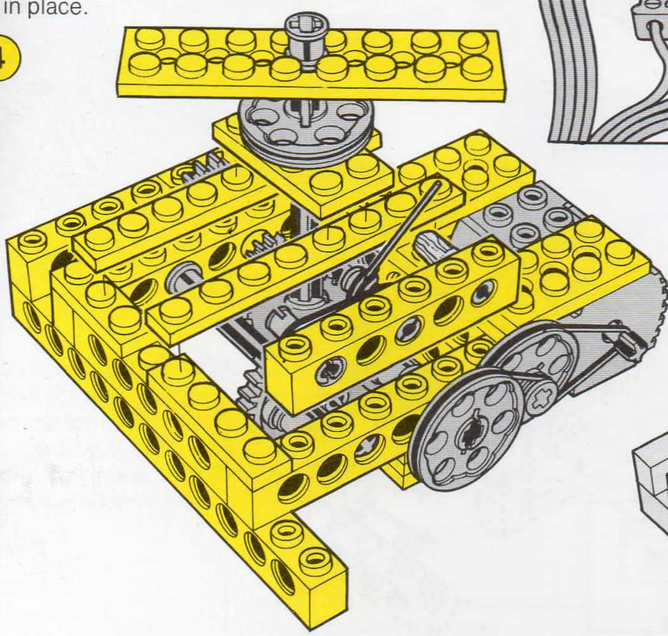
▷ Another layer of plates is added front and back. The motor unit that turns the arm is fixed at the back of the base with more plates. Hook the loose drive belt onto the motor spindle.

3

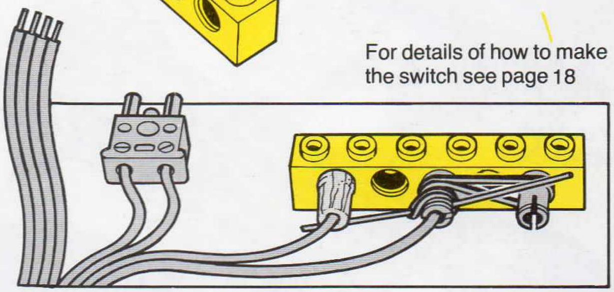


▽ As a support for the shoulder of the arm, add two 1×8 plates and, fitting on the axle, a 2×4 plate and bush. Thread the wires for the base switch sensor under the motor fixing plate and up, and fix the switch beam in place.

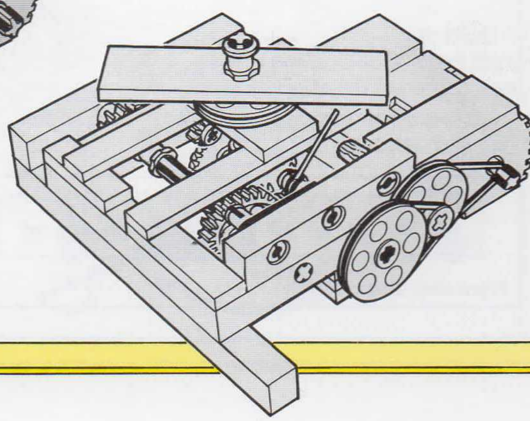
4



For details of how to make the switch see page 18

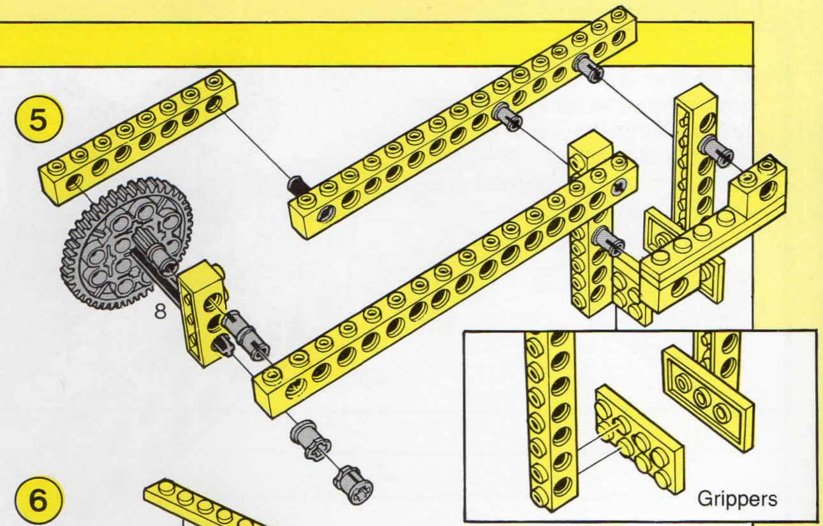


▽ The complete base unit. Put the plug into the motor and push the foil plugs into the switch pins as shown above.

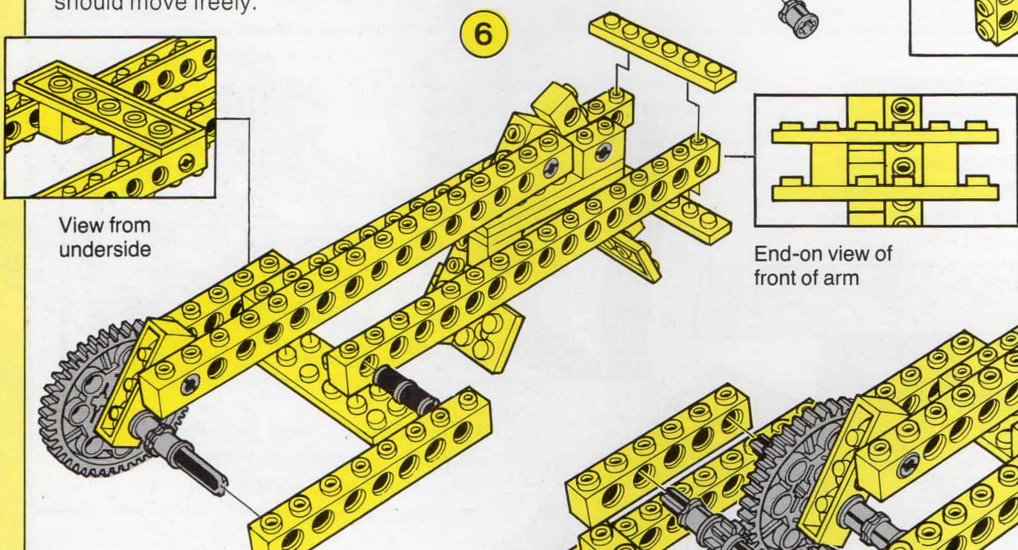


THE ARM

▷ The arm is made from the motor starter set. First build one half of the arm and the gripper shown here in exploded view.

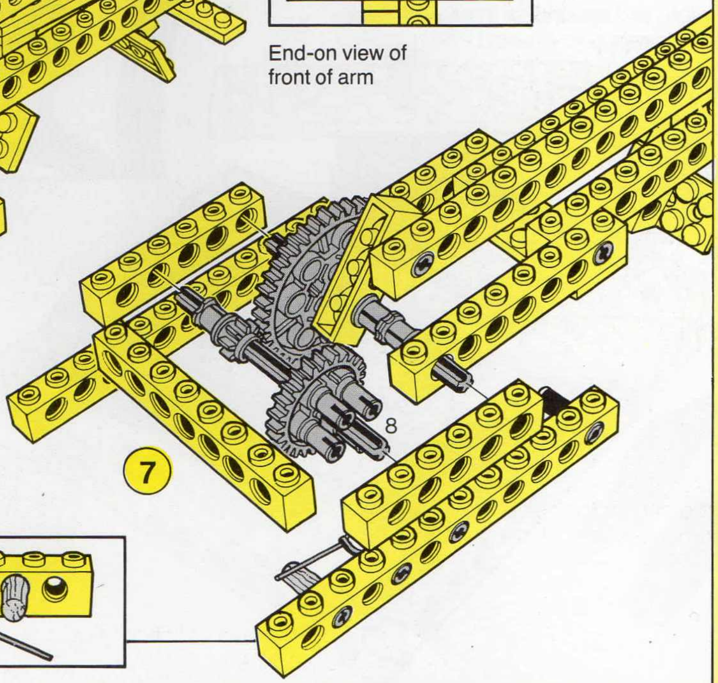


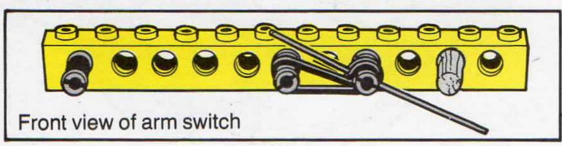
▽ The other half of the arm is added to complete the structure. Use the detail diagrams to check the construction. The mechanism should move freely.



▷ Build the switch and the drive shaft that sits behind the arm, then construct the shoulder unit.

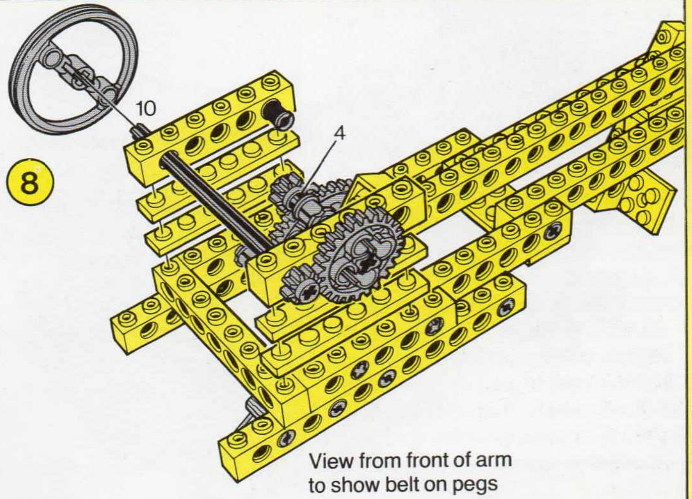
For details of how to make the switch see page 18



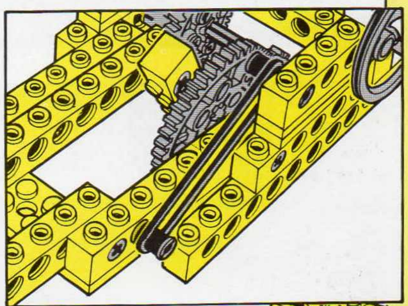


▷ Add the last layer of gears and beams to the top of the model. The gears should move freely and smoothly.

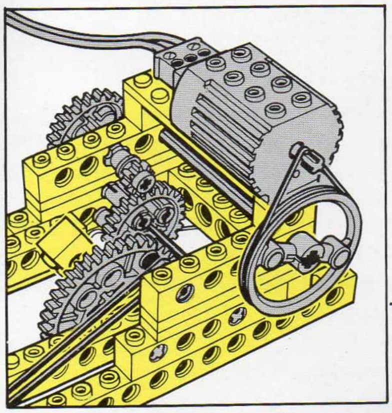
8



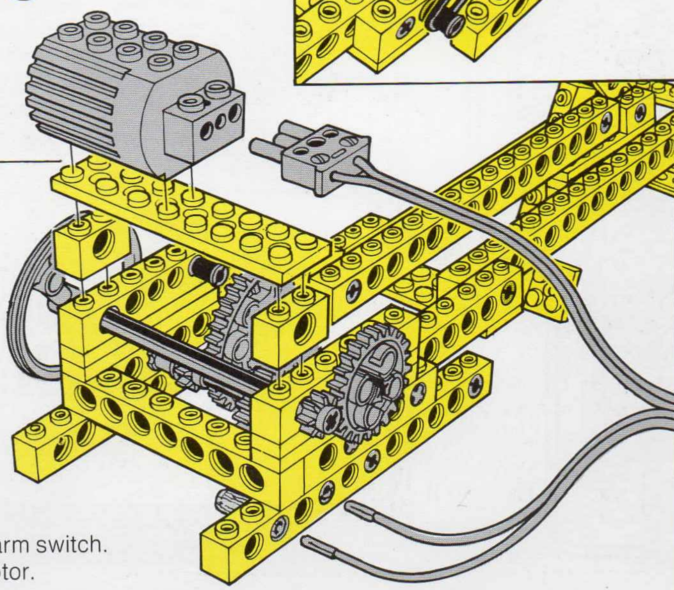
View from front of arm to show belt on pegs



View of motor-pulley assembly

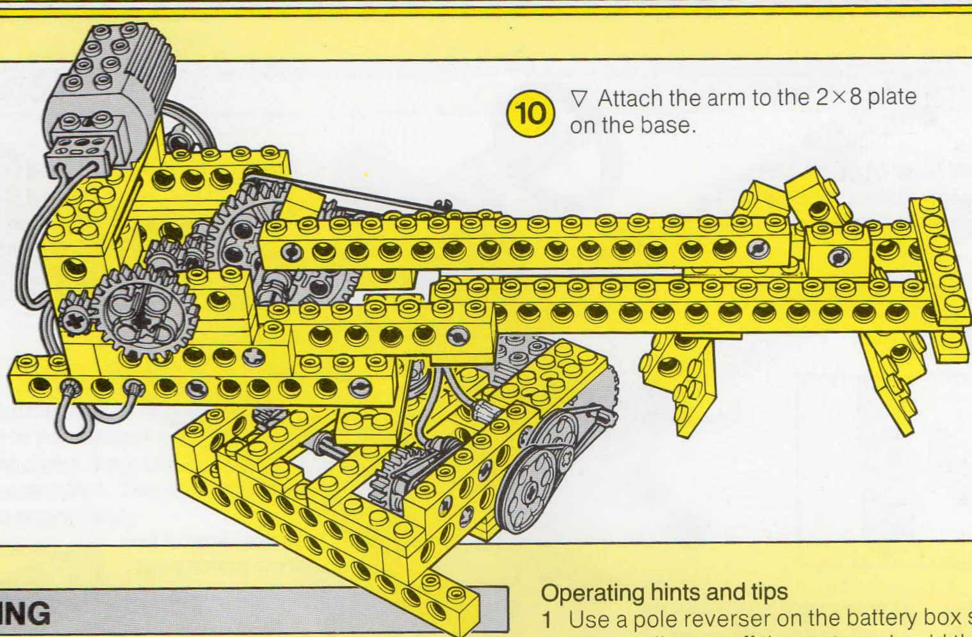


9



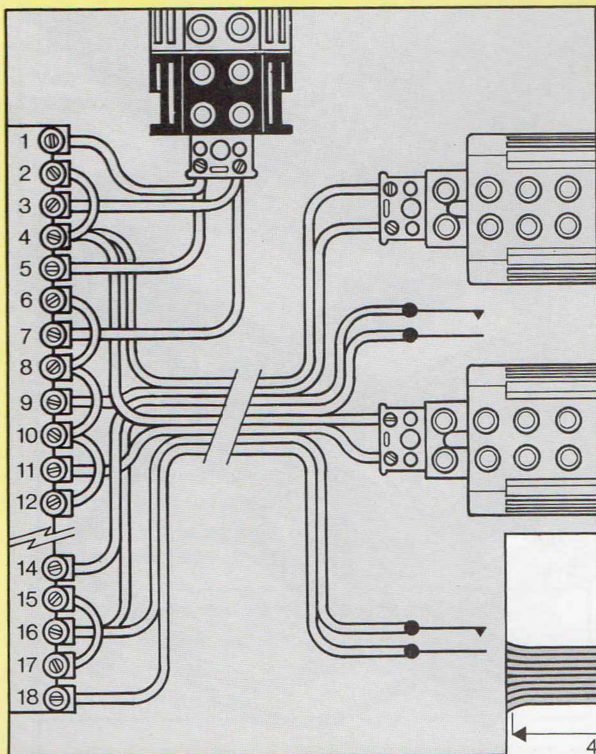
△ The shoulder motor fits onto a 2×8 plate set on two small beams on the top of the model. Use one large drive belt on the pulley and one on the pegs to help the arm move up and down.

Plug the wires into the arm switch. Put the plug into the motor.



(10) ∇ Attach the arm to the 2×8 plate on the base.

WIRING



Operating hints and tips

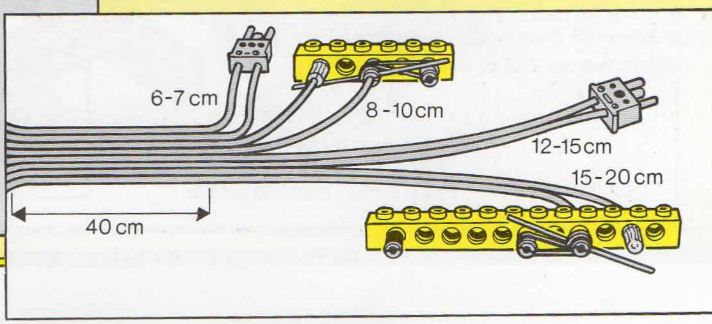
1 Use a pole reverser on the battery box so that you can manually turn off the motors should the arm move too far.

2 To allow the arm to grasp objects more easily, add some small pieces of draught excluder tape to the fingers of the gripper.

3 Once you have connected the arm to the interface, attach the main wire to the back of the base with Blu-Tac or tape to keep it neat and out of the way.

4 Both switches must function properly at all times since they are connected to the same switch input on the interface block. Each must remain closed when its driving motor has stopped.

∇ Suggested separation of the 8 wires in the ribbon cable: 2 wires to each of the motor plugs and 2 to each of the switches.



```
 10 REM MINI ARM CONTROLLER
 20 PRT=56577:POKE56579,63
 30 DR(0)=6:DR(2)=9:M1=16:M2=32:T=128
 40 DIMR1(50),R2(50)
 100 REM MENU
 110 PRINT"{CLR}MINI ARM PROGRAMMER"
 120 PRINT"{DOWN}SELECT:-"
 130 PRINT"{DOWN}1 - TEACH MOVES"
 140 PRINT"{DOWN}2 - REPEAT MOVES"
 150 PRINT"{DOWN}3 - SET HOME POSITION"
 170 GETR$:IFVAL(R$)<1ORVAL(R$)>3THEN170
 180 ON VAL(R$)GOSUB200,300,400
 190 GOTO110
 200 REM TEACH MOVES
 210 PRINT"{CLR}TEACHING MODE":RP=1:A$=""
 220 X$=A$:GETA$
 230 IFA$=""ANDPEEK(197)<>64THENA$=X$
 240 IFA$=CHR$(13)THENRETURN
 250 IFA$="S"THENGOSUB270
 260 GOSUB500:GOTO220
 270 REM SET A POSITION IN ROUTINE
 280 R1(RP)=C1:R2(RP)=C2
 290 RP=RP-(RP<50):RETURN
 300 REM REPEAT MOVES
 310 PRINT"{CLR}REPEATING MOVES"
 320 FORI=0TORP-1:N1=R1(I):N2=R2(I)
 330 GOSUB600:NEXTI:RETURN
 400 REM HOME ARM
 410 PRINT"{CLR}HOME ARM MODE"
 420 GETA$:IFA$=""THEN420
 430 IFA$=CHR$(13)THENC1=0:C2=0:RETURN
 440 GOSUB500:GOTO420
 500 REM CALCULATE DRIVE VALUES
 510 N1=C1+(A$="1")-(A$="Q")
 520 N2=C2+(A$="2")-(A$="W")
 530 GOSUB600:RETURN
 600 REM DRIVE TO N1,N2
 610 SP=N1-C1:IFSP=0THEN630
 620 POKEPRT,M1+DR(SGN(SP)+1):GOSUB700
 630 SP=N2-C2:IFSP=0THEN650
 640 POKEPRT,M2+DR(SGN(SP)+1):GOSUB700
 650 C1=N1:C2=N2:RETURN
 700 REM COUNT PULSES
 710 FORQ=1TOABS(SP)
 720 IF(PEEK(PRT)ANDT)=0THEN720
 730 IF(PEEK(PRT)ANDT)=TTHEN730
 740 NEXTQ:POKEPRT,0:RETURN
READY.
```

PROGRAMMING

After typing in the program, SAVE it on tape. When you RUN the program, the menu will appear on the screen. Select an option by pressing either the 1, 2 or 3 key. To TEACH the arm a series of moves, use the 1 and Q keys to make the arm turn to and fro, and 2 and W to make it go up and down. At the end of each move, press S to set the position in memory. At the end of a series of moves press RETURN to go back to the menu. Pressing 2 will repeat the moves that you have taught the mini arm. HOMING the arm allows you to use the programming keys to set the first position for the series of moves.

PLOTTER

You can program this drawing robot, or plotter, to draw graphs, patterns, diagrams and even letters on a sheet of paper. There are two main types of plotter: X-Y and drum. On an X-Y plotter, the paper is laid flat and the pen moved up and down and side to side over it. On a drum plotter, the paper is attached to a drum which is rotated, and the pen is moved from side to side only. Our model is a drum plotter. It uses two motors, one to move the pen, the other to move the drum. A home-made electromagnet raises and lowers the pen.

Doodle, doodle, doodle

The motor at the front drives the penholder to the left and right across the paper to draw horizontal lines. When the drum is rotated but the holder kept still, the pen draws vertical lines. Diagonal lines are drawn using a combination of small vertical and horizontal lines. To move the pen to another point on the paper without drawing a line, the holder is raised. This is done by turning on the electromagnet, which attracts a small metal paper fastener fixed on the holder.

WHAT YOU NEED

2	4	6	8	12	16	
9	7	3	4	3	8	Beams

1×2	1×4	1×6	1×8	2×4	2×6	4×6	
5	3	7	6	8	1	1	Plates

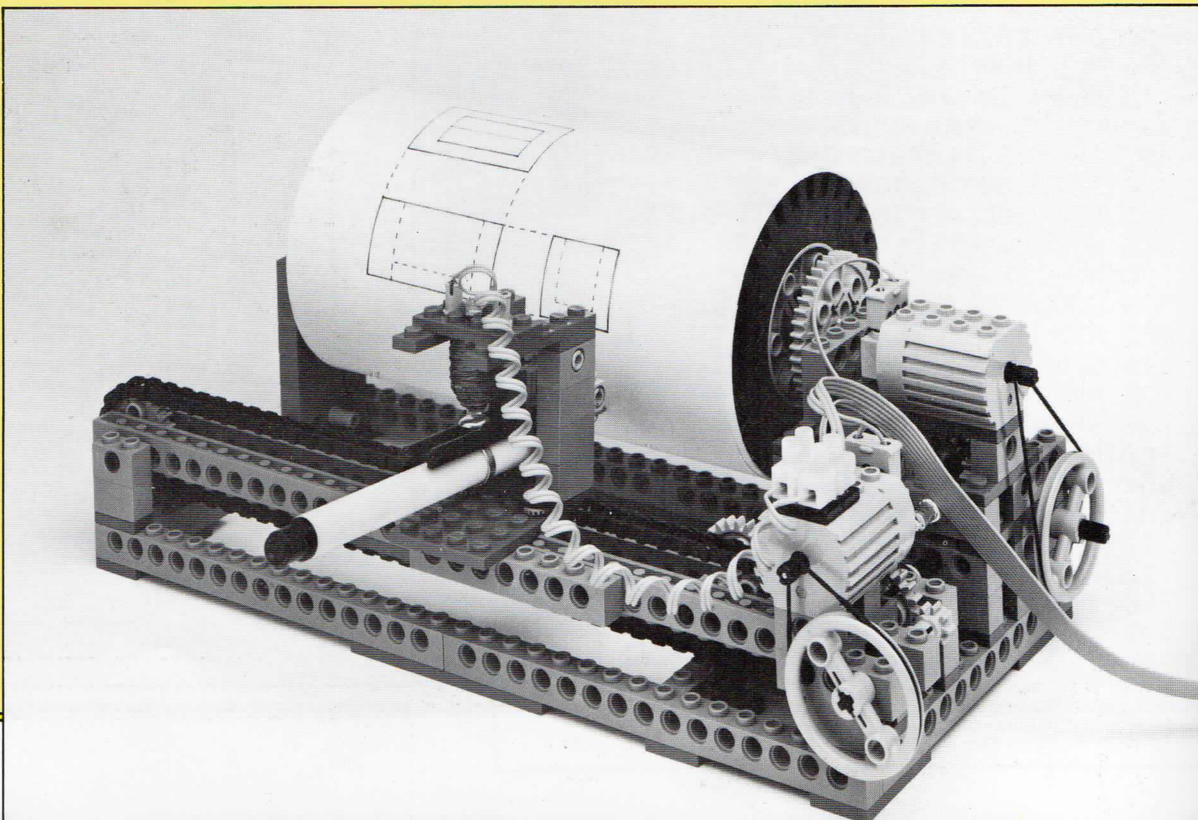
8	24	40	SP	LP	
7	6	2	4	3	Gears

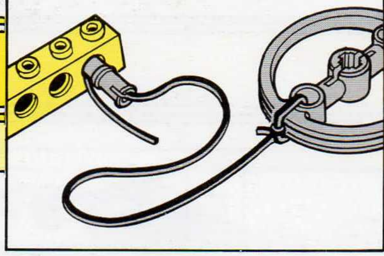
4	22	2	2	5	3	2

2	4	8	
11	10	2	Non-holed beams

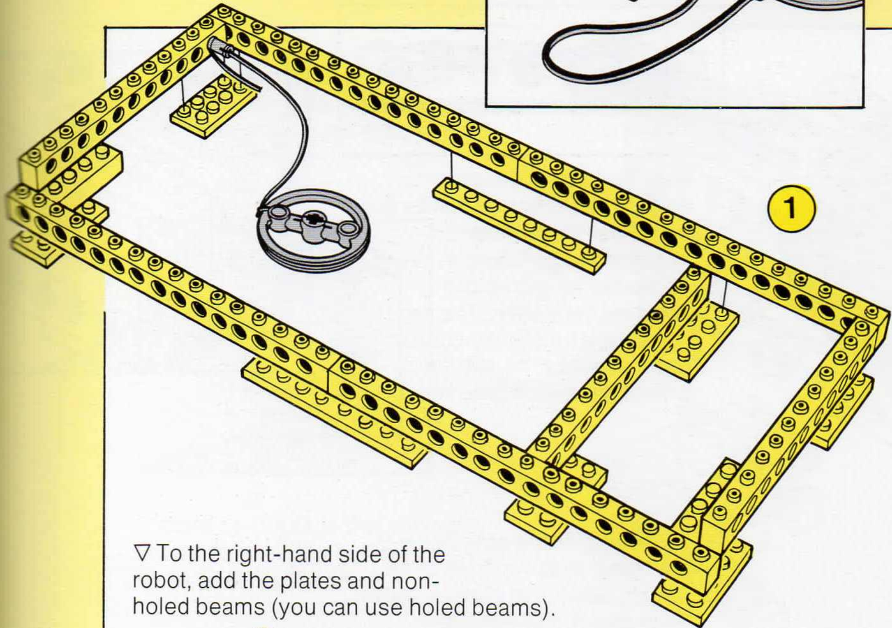
2	4	6	8	
2	1	7	2	Axles

2 motors; 4 small belts, 3 medium; 2 large wheels; 4 plugs, 2 battery boxes; 50cm length of 10-way ribbon cable, 1 paper fastener, paper clips, 1 steel metric-5 25mm-long bolt with nut; 34 s.w.g. enamelled copper wire (obtainable from electronics shops); thin card, double-sided sticky tape, Blu-Tack, 2-way terminal block.



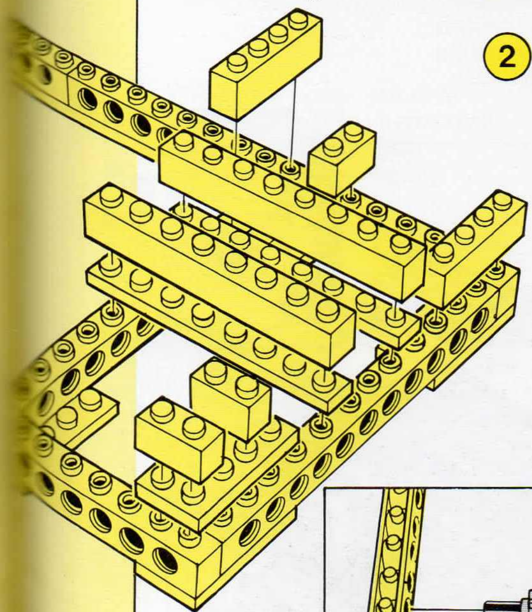


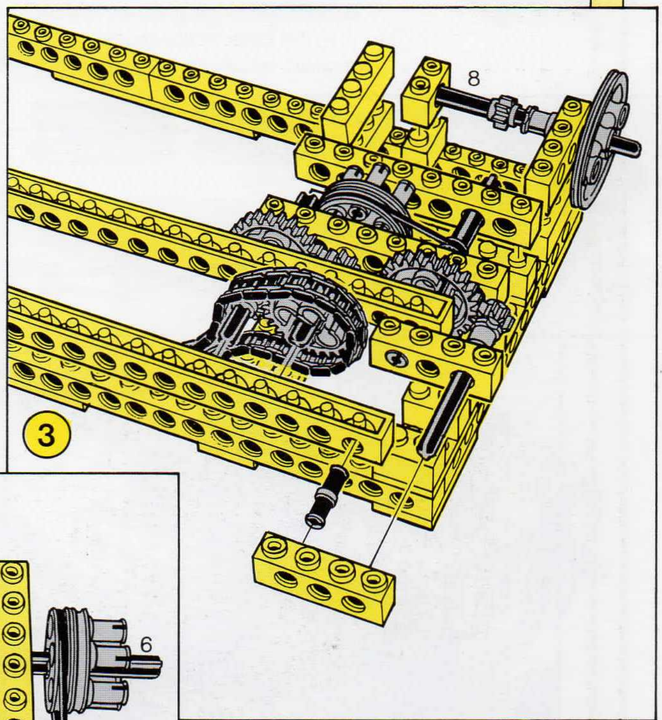
◁ Build the frame for the base. Break a medium belt, tie one end through a rim-hole on the pulley and make a knot, and secure the other end to the cross-beam using a pin.

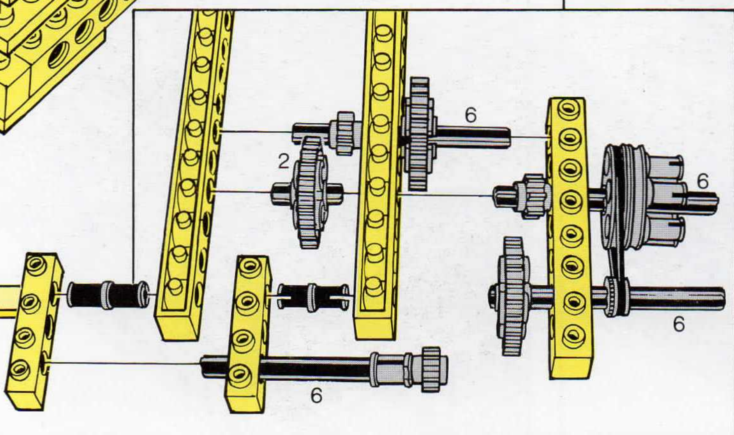


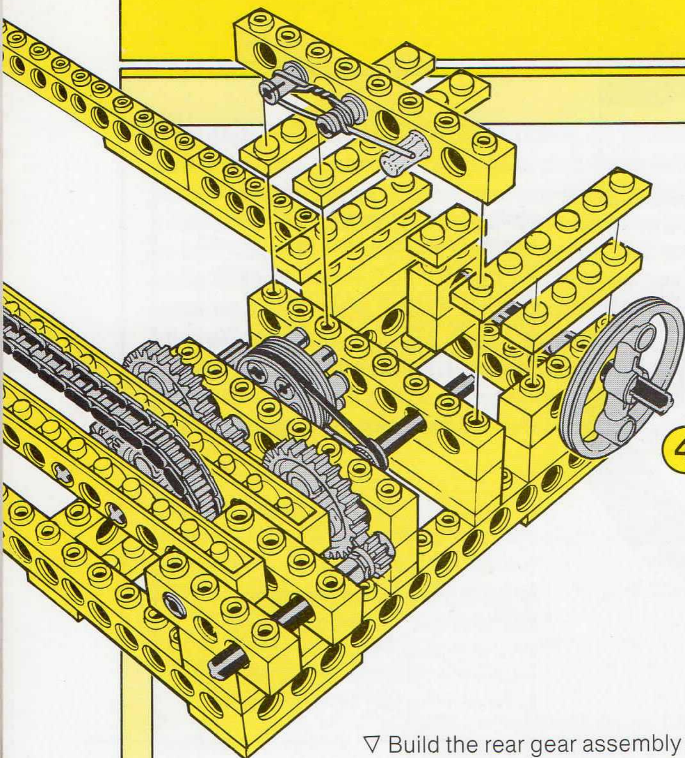
▽ To the right-hand side of the robot, add the plates and non-holed beams (you can use holed beams).

▽ Build the gear assembly as below (inset). Use a small belt on the pulley. Place the chain round the 24-tooth cog which sits between the two 16-unit beams.

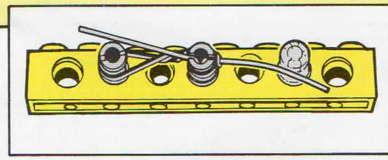




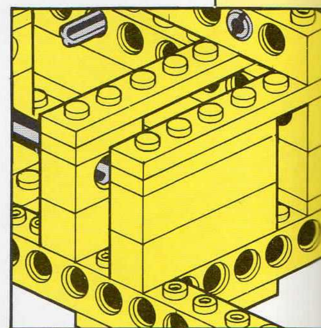




4



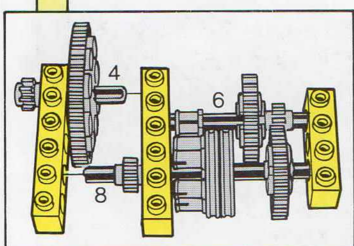
◁ At the back of the assembly, add the plates and switch for the belt drive. Remember to pull the paper clip arm over above the foil pin before securing the switch. The paper clip should be slightly curved on what will be the left-hand side of the switch (see inset).



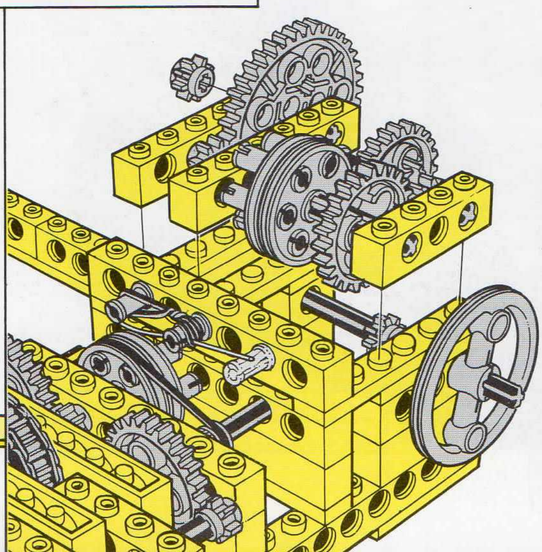
Rear of drive assembly

▽ Build the rear gear assembly and fix it to the back of the model. The gears must run smoothly.

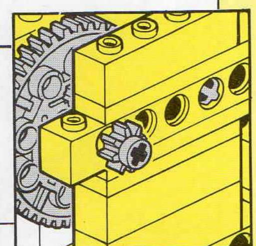
▽ The rear 4-unit-long beam is not holed. Add the drum drive switch, beam and plate as shown. As in step 4, pull the paper clip arm over the foil pin and note the left-hand part of the clip must be curved.

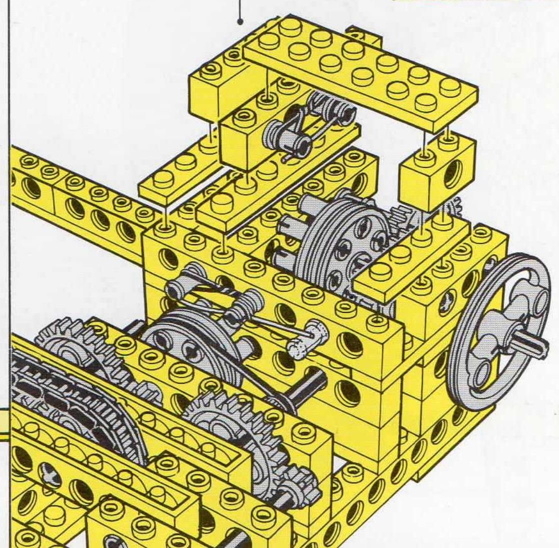


5

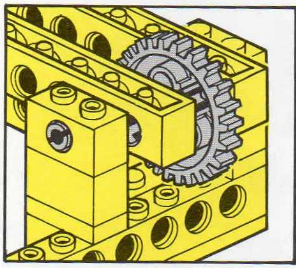


6

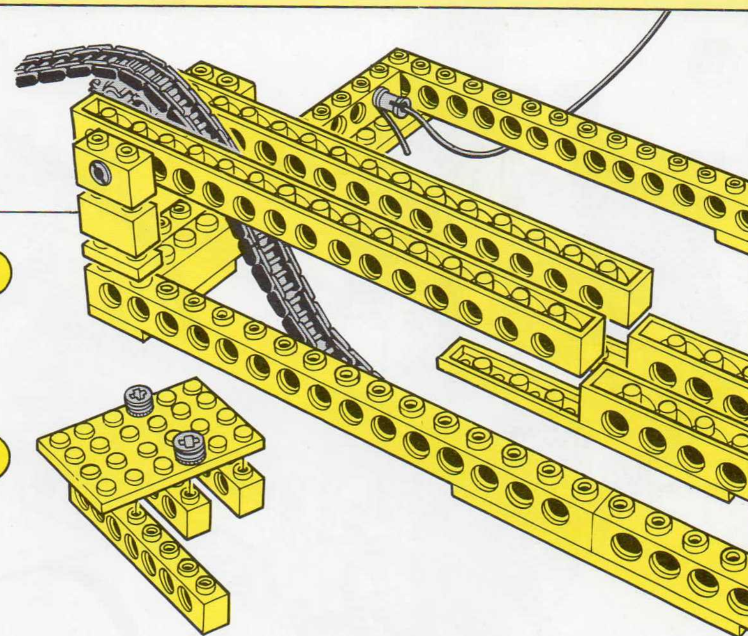




▷ Working again at the front left-hand side, add the other half of the chain drive assembly. Allow the chain to pass under the 24-tooth cog wheel.



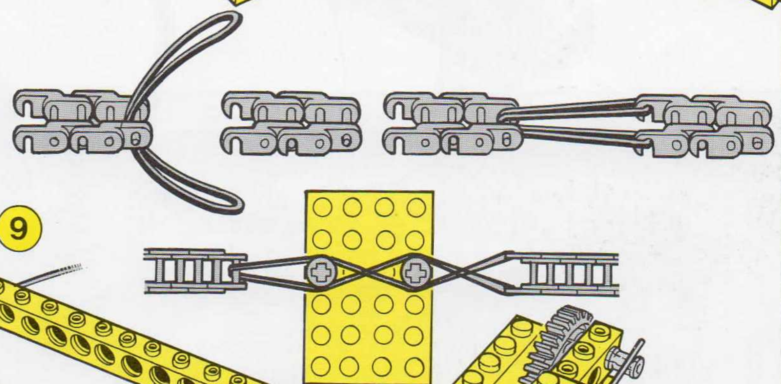
7



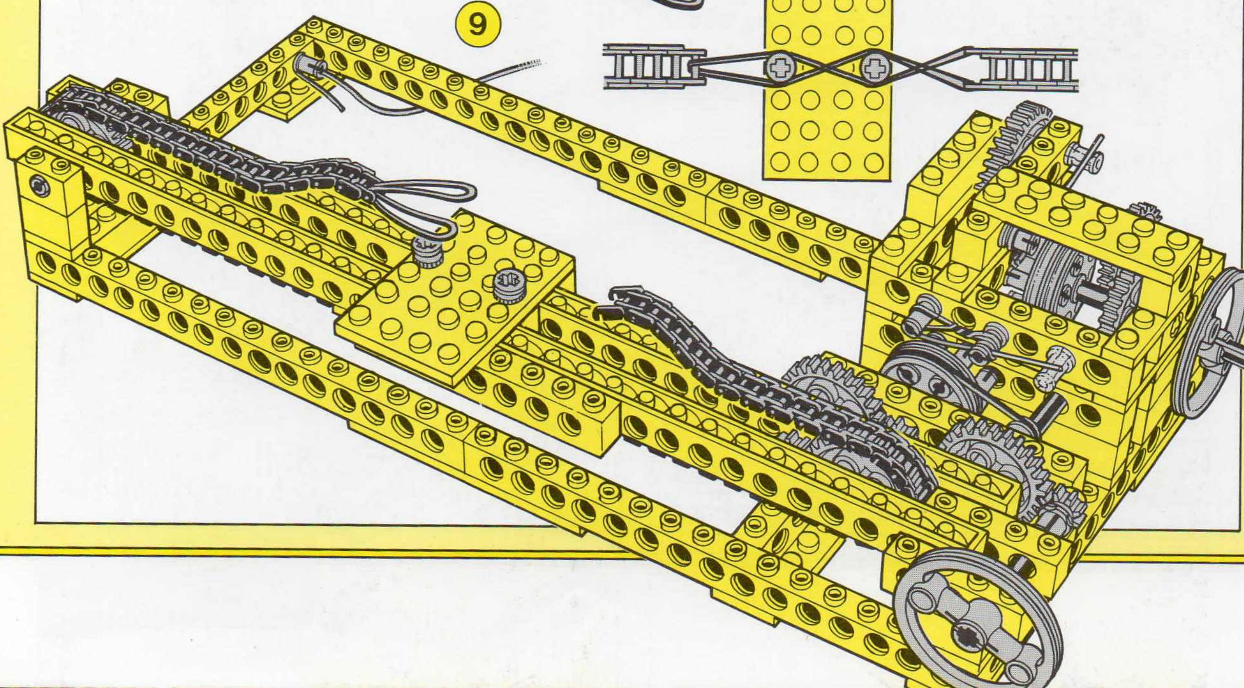
8

▷ Make up the slide that supports the penholder. The half-bushes sit on studs on the plate.

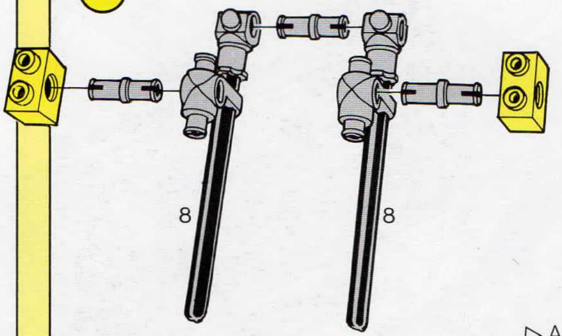
▷ Place the holder-unit slide on the model. Using a small belt, link the ends of the chain across the slide as shown.



9

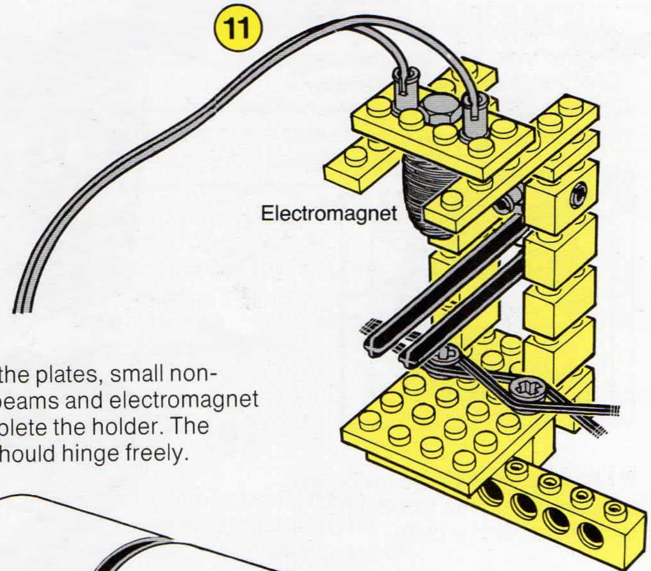


10



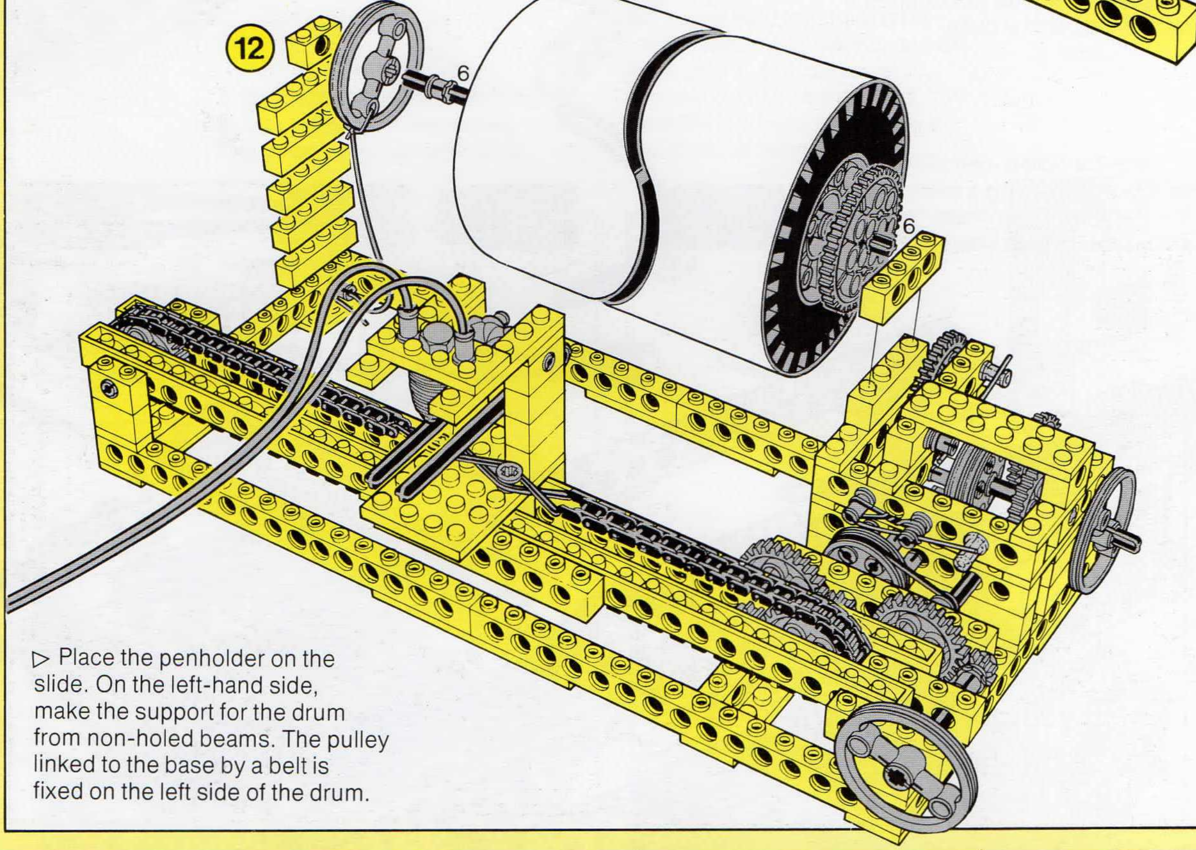
△ Build the penholder arm units from toggles, pins, axles and steering arms.

11

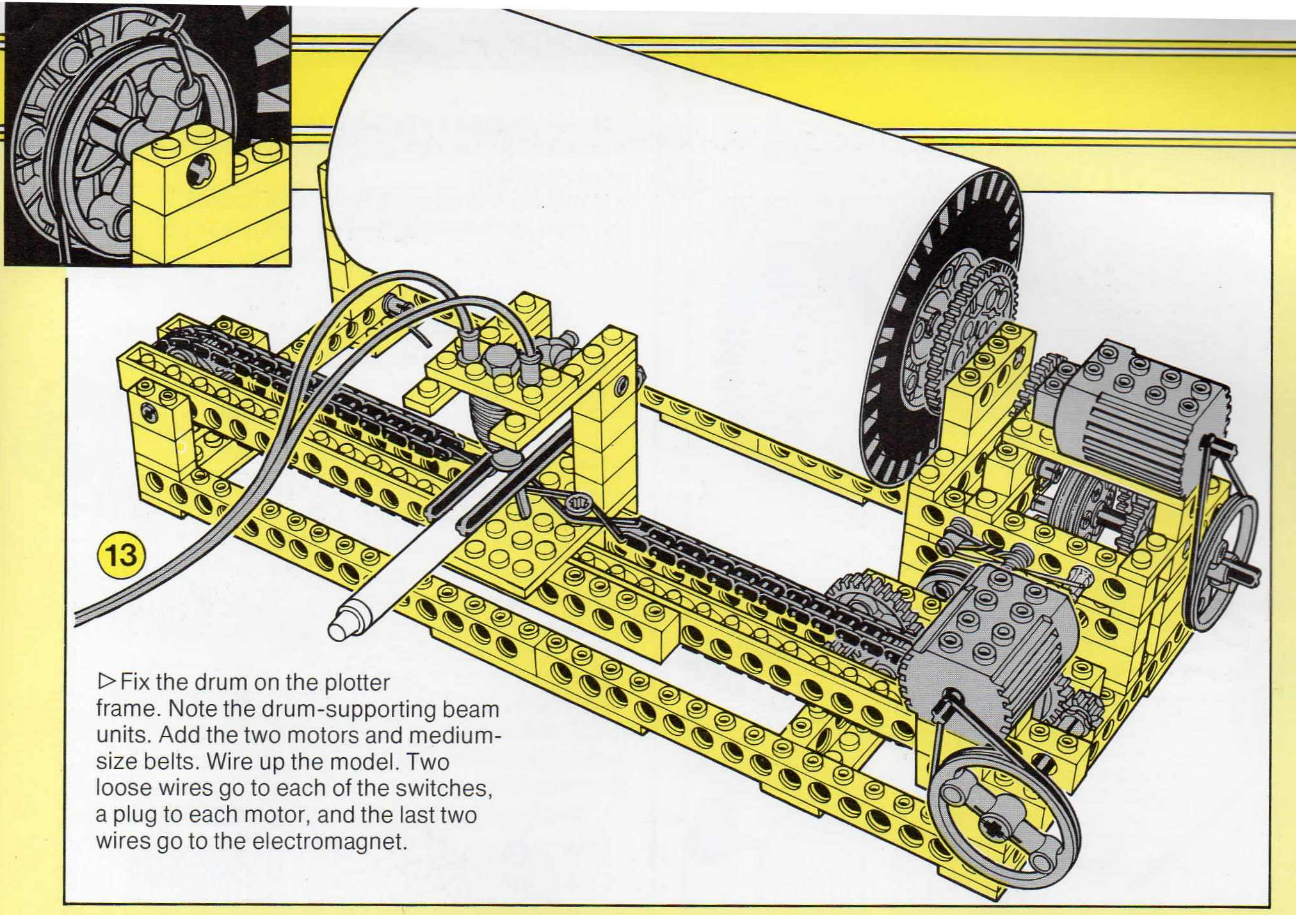


▷ Add the plates, small non-holed beams and electromagnet to complete the holder. The axles should hinge freely.

12

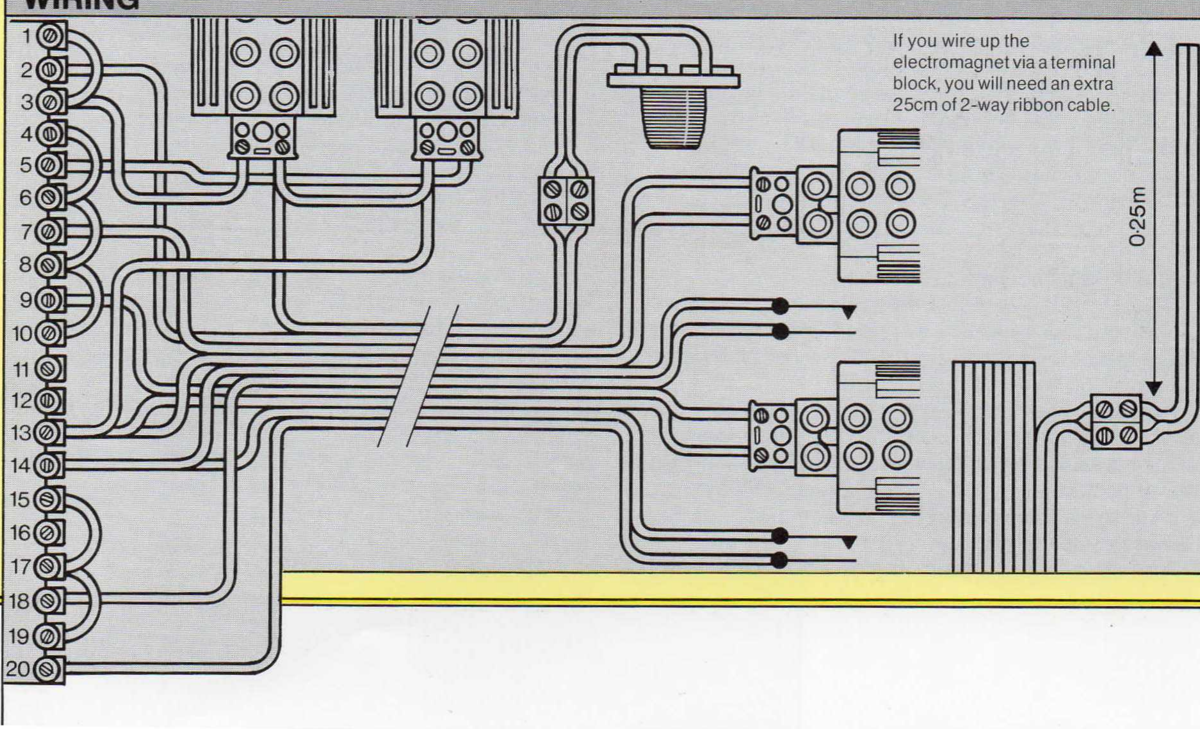


▷ Place the penholder on the slide. On the left-hand side, make the support for the drum from non-holed beams. The pulley linked to the base by a belt is fixed on the left side of the drum.



▷ Fix the drum on the plotter frame. Note the drum-supporting beam units. Add the two motors and medium-size belts. Wire up the model. Two loose wires go to each of the switches, a plug to each motor, and the last two wires go to the electromagnet.

WIRING



If you wire up the electromagnet via a terminal block, you will need an extra 25cm of 2-way ribbon cable.

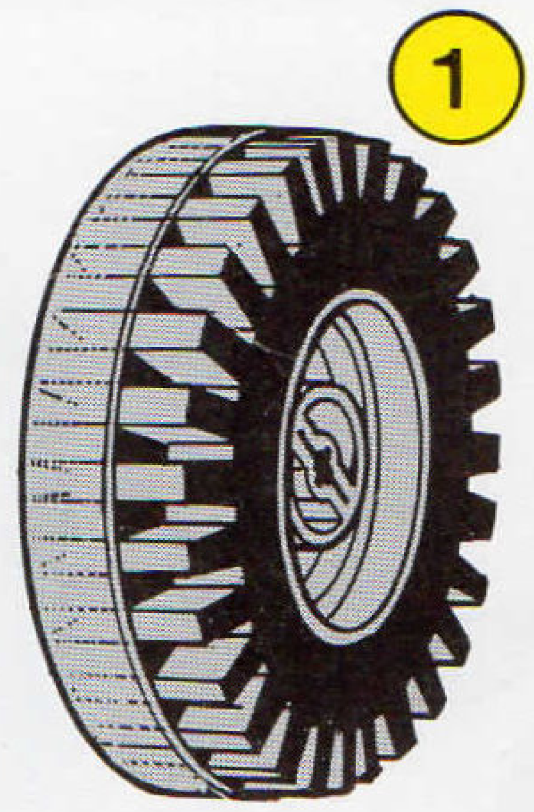
MAKING THE DRUM

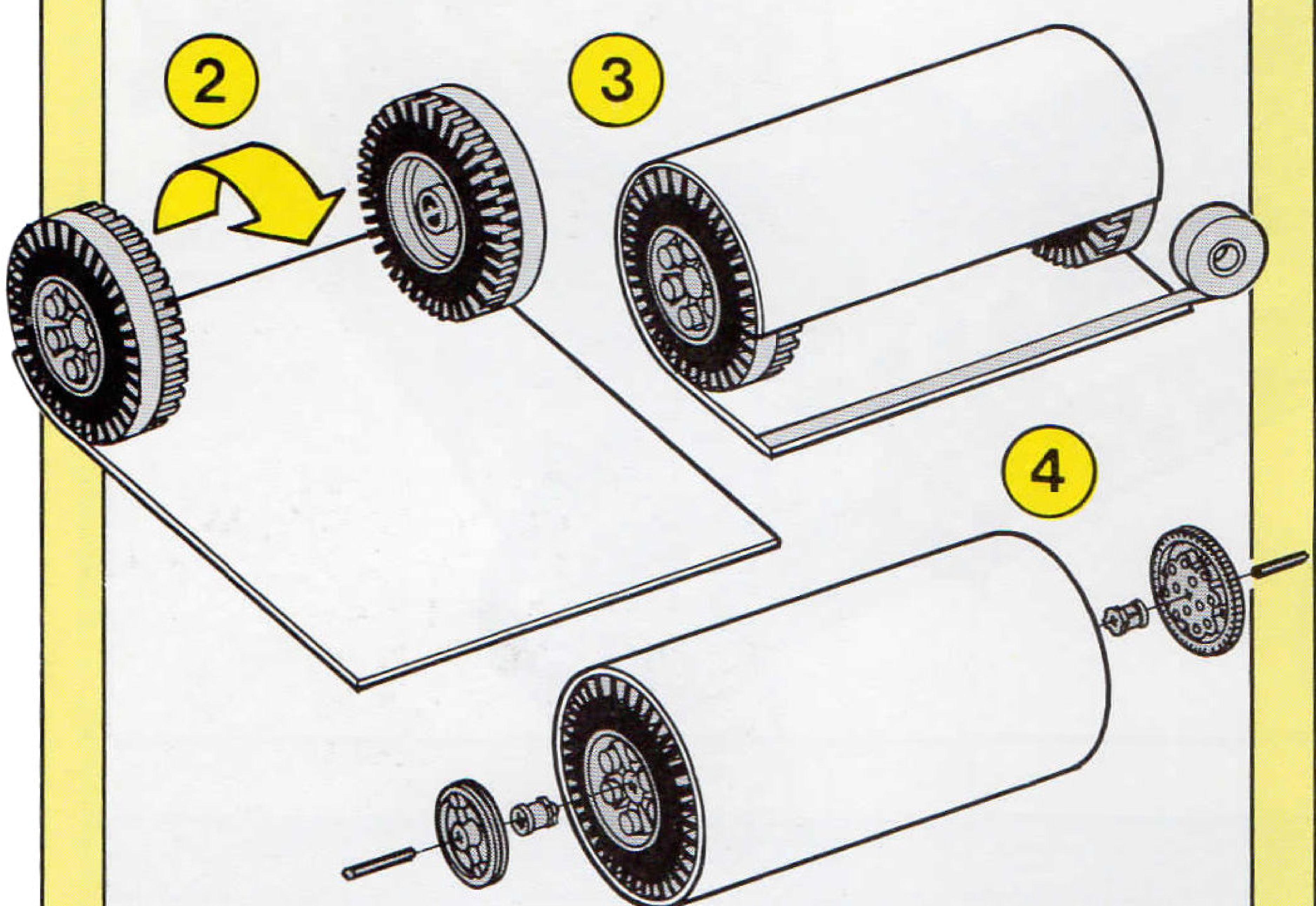
You will need 2 large LEGO wheels, a piece of card 160×280mm and double-sided sticky tape.

1 Wrap double-sided sticky tape round the outer edge of each wheel.

2 Peel off the tape backing and roll the card round the wheels. The edges of the card must align with the edges of the wheels.

3 When you have nearly finished rolling the card round, place a strip of sticky tape across the front end of the card. Peel off the backing, then stick the end down to complete the drum.





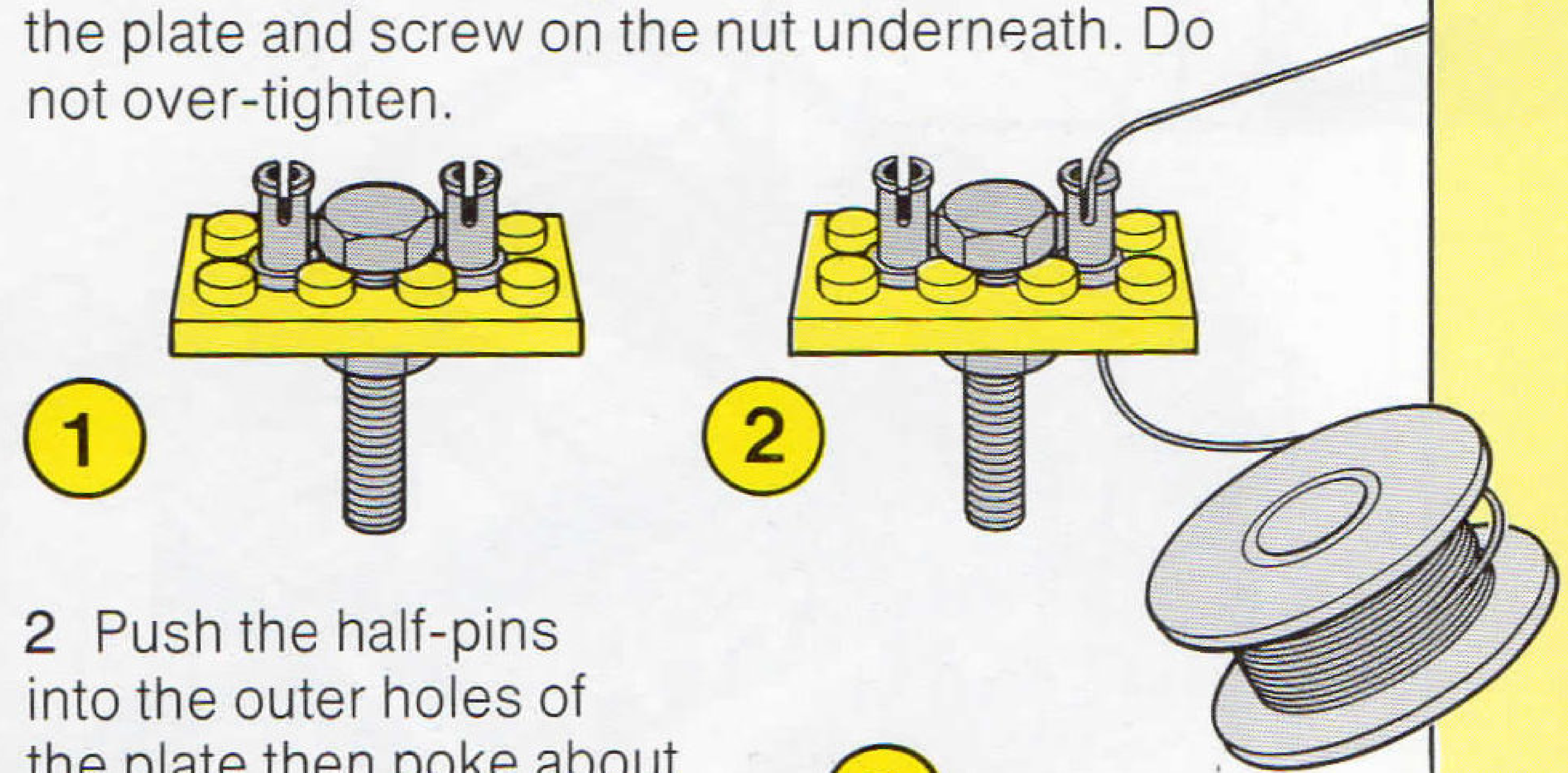
4 Into each wheel, push a 6-length axle bearing a bush and a 40-tooth cog wheel. The axles fit through the end beam-stacks on the plotter.

Wrap a sheet of drawing paper round the drum and stick down the ends using Blu-Tack. When you have finished your drawing, simply remove the paper and replace it with a new sheet.

MAKING THE MAGNET

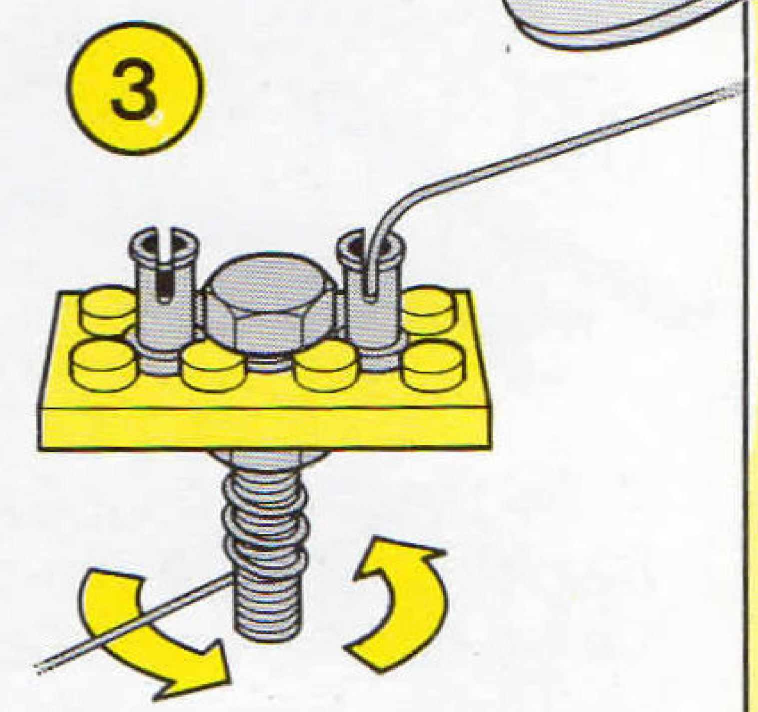
You will need a 2×4 holed plate, 2 half-pins, 1 steel M5 25mm bolt with nut, and some 34 s.w.g. enamelled wire.

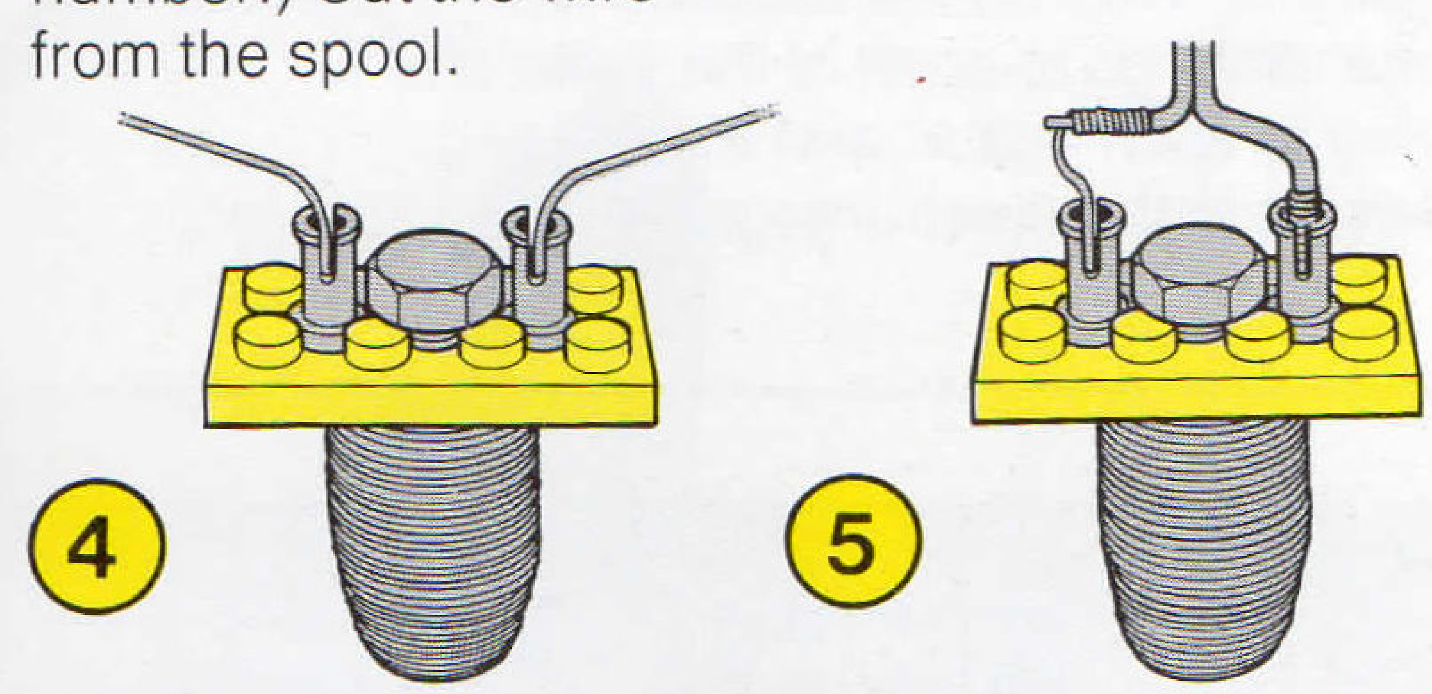
1 Poke the bolt through the central hole of the plate and screw on the nut underneath. Do not over-tighten.



2 Push the half-pins into the outer holes of the plate then poke about 10cm of wire through one of the half-pins.

3 Wind the wire round the bolt about 800 turns. (Write down the numbers 1 to 8 and each time you reach 100, cross off the next number.) Cut the wire from the spool.





4 Poke the loose end of the wire through the other half-pin. Cut the ends to leave long 'tails'. 5 Gently scratch the enamel off the ends of each tail, twist the bare ends with the bare wires of the ribbon cable and poke the 'plugs' into the half-pins as shown.

Operating hints and tips

1 The pen should rest on the paper so that the paper fastener is about 1mm from the electromagnet. Adjust the gap between fastener and magnet by sliding the pen in and out of the holder.

2 To keep the electromagnet wiring neat, make a junction on top of the front motor to connect the magnet to the ribbon cable. Place two small smooth plates on the motor and, with double-sided sticky tape, attach a 2-way piece of terminal block (obtainable from an electrical shop). See main photo on page 32.

3 The belt used in the chain drive gears (step 3) should be small and very tight. You may find it best to use a medium-size belt doubled-up.

4 To support the pen in the holder, wrap a belt round the pen and holder arms (axles). Slip the paper fastener between them. Use a fine-point felt tip pen.

5 Do not leave the electromagnet on for too long. It will get quite hot and will drain the batteries very quickly.

6 You can use any kind of paper to draw on but for best results use a good quality cartridge paper such as typing paper.

```
10 REM PLOTTER CONTROLLER
20 PRT=56577:POKE56579,63
30 MX=8:MY=16:D(0)=2:D(2)=4:PS=0
40 XC=0:YC=0:XN=0:YN=0
50 XS=128:YS=64
100 REM DEMO, CIRCLE
110 GOSUB1500
120 FORA=0TO2*π STEPπ/20
130 XN=R*COS(A):YN=R*SIN(A)
140 GOSUB2000:GOSUB1000:NEXTA
150 GOSUB1100:END
1000 REM PEN DOWN
1010 PS=0:POKEPRT,PS:RETURN
1100 REM PEN UP
1110 PS=1:POKEPRT,PS:RETURN
1500 REM RESET FROM KEYBOARD
1510 PRINT"RESET - USE CURSOR KEYS TO"
1520 PRINT"MOVE, THEN PRESS RETURN"
1530 GOSUB1100
1540 GETA$:IFA$=""THEN1530
1550 XN=XN+(A$="▐")-(A$="▌")
1560 YN=YN+(A$="⌐")-(A$="▀")
1570 IFA$=CHR$(13)THEN1590
1580 GOSUB2000:GOTO1540
1590 XC=0:YC=0:RETURN
2000 REM MOVE TO XN,YN
2010 XN=INT(XN):YN=INT(YN)
2020 DX=XN-XC:IFDX=0THEN2100
2030 POKEPRT,MX+D(SGN(DX)+1)
2040 FORI=1TOABS(DX)
2050 IF(PEEK(PRT)ANDXS)=0THEN2050
2060 IF(PEEK(PRT)ANDXS)=XSTHEN2060
2070 NEXTI:XC=XN:POKEPRT,PS
2100 DY=YN-YC:IFDY=0THEN2160
2110 POKEPRT,MY+D(SGN(DY)+1)
2120 FORI=1TOABS(DY)
2130 IF(PEEK(PRT)ANDYS)=0THEN2130
2140 IF(PEEK(PRT)ANDYS)=YSTHEN2140
2150 NEXTI:YC=YN:POKEPRT,PS
2160 RETURN
READY.
```

PROGRAMMING

The main program contains subroutines to control the plotter. We have given 3 examples which draw different pictures. The routines are fairly easy to use. To raise the pen in a program, type GOSUB 1100. To lower the pen, type GOSUB 1000. To reset the machine back to its starting point, type GOSUB 1500. To move the pen to the position X, Y, where X is a position on the horizontal scale and Y on the vertical scale, type XN=x; YN=y: GOSUB 2000.

```
100 REM DEMO, SQUARE
110 GOSUB1500:GOSUB1000
130 INPUT"LENGTH OF SIDE";L
140 XN=L:GOSUB2000
150 YN=L:GOSUB2000
160 XN=0:GOSUB2000
170 YN=0:GOSUB2000
180 GOSUB1100:END
READY.
```

```
100 REM DEMO, SINE WAVE
110 GOSUB1500
130 FORXN=0TO80
140 YN=((80-XN)/6)*SIN(XN/3)
150 GOSUB2000:GOSUB1000
160 NEXTXN:GOSUB1100
170 END
READY.
```

MAXI ARM

We've called this robot a maxi-arm because it is both larger and more versatile than our mini-arm machine. It can not only move in a circle, it can also extend and shorten in length from the 'shoulder' to pick and place objects anywhere within a large area. It uses three motors and is built from a fairly large assortment of LEGO Technic pieces. (If you don't have enough LEGO pieces yourself, perhaps you can build the robot with a friend who has some LEGO Technic kits.) The base unit is the same as that used for the mini-arm (see pages 25-31), but to make the robot more sturdy, the base must be placed on two 10×20 plates.

Reaching out, picking up

Under control from the computer, the top plate of the base unit that supports the arm can rotate and stop at any one of 36 positions. The arm can slide in and out at the shoulder, stopping at any one of about 20 positions. The gripper at the end of the arm closes its fingers on an object and then raises the object off the ground. At the wrist, the gripper has about 15 stop positions. The arm-support, shoulder and gripper all have a switch sensor which allows the computer to 'sense' how far each has driven.

In effect, the gripper can be moved to pick up an object anywhere within a sphere-shaped area around the shoulder. The technical name for this type of robot arm is a polar arm.

WHAT YOU NEED

1×2	1×3	1×4	1×8	2×4	2×8	4×6	
7	6	7	2	9	7	1	Plates

8	14C	24	24C	40	SP	LP	
8	4	4	1	1	4	1	Gears

2	4	6	8	12	16	
9	8	12	10	1	2	Beams

2	4	6	8	10	
3	4	4	4	1	Axles

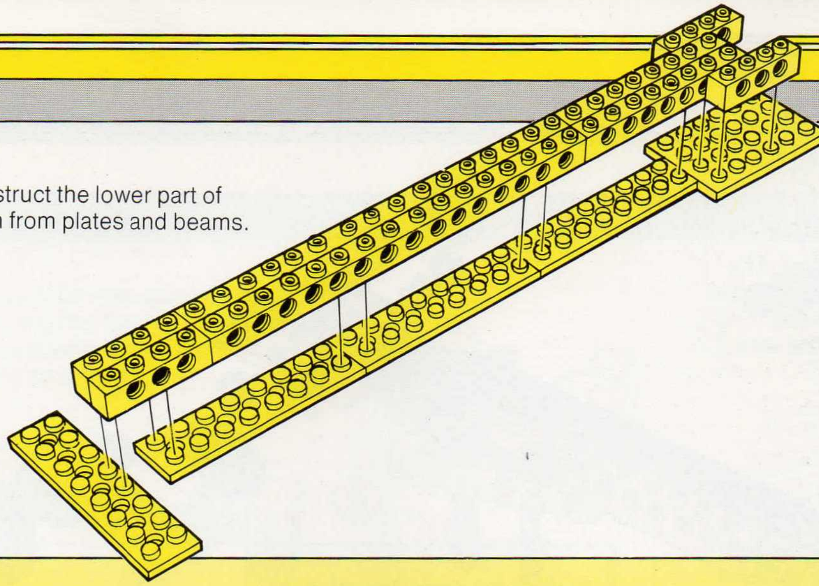
1	1	8	12	2	25

1	6	2	2	2

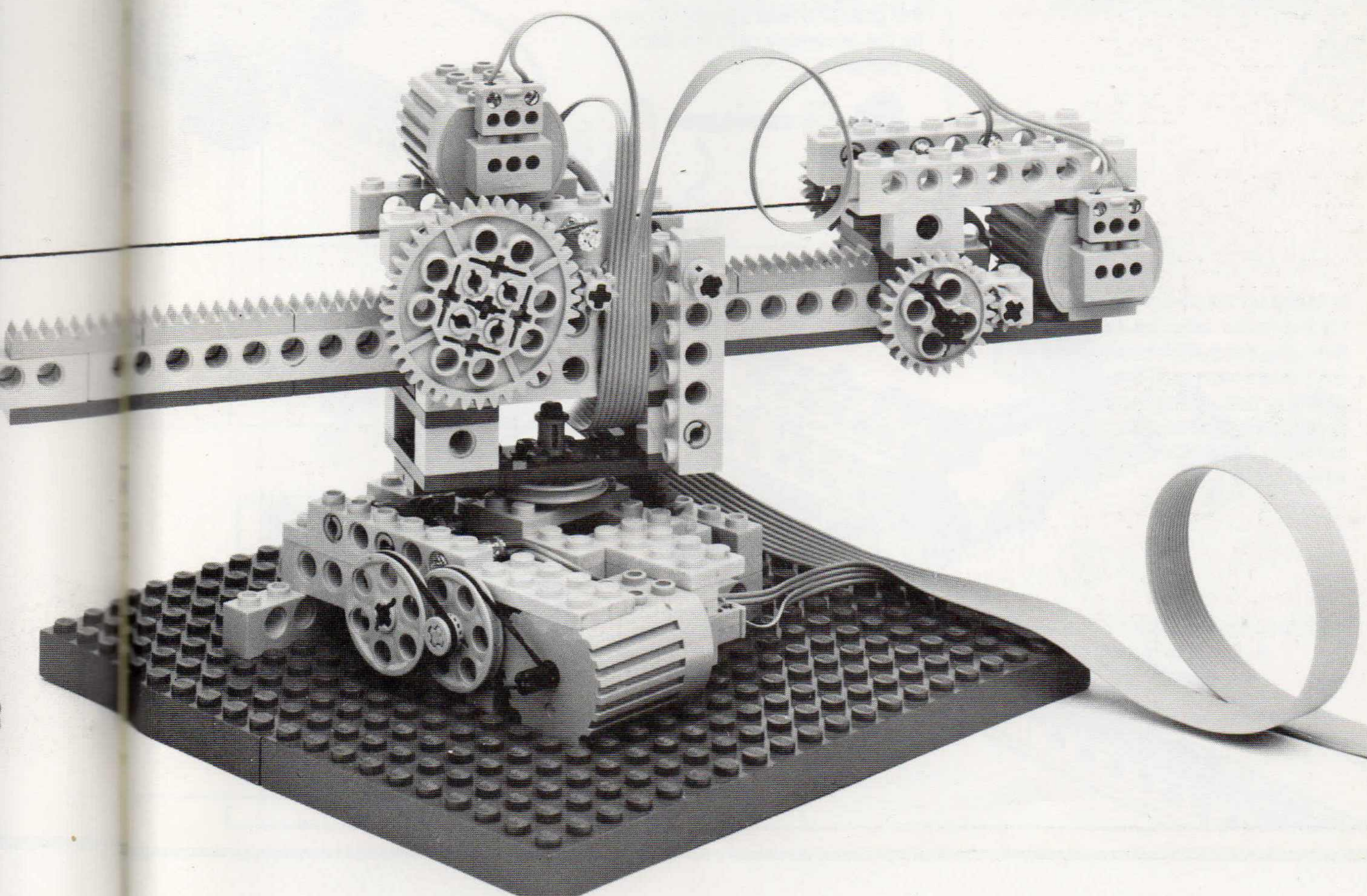
3 motors, 2 battery boxes, 6 small belts, 2 medium belts, 5 plugs, 2 10×20 base plates, short length of draught-excluder tape, 100cm of 12-way ribbon cable, 3 2.5cm squares of cooking foil, paper clips, 30cm length of string.

With this arm, you could move the pieces in a game of chess if pieces and board are quite small.

THE ARM

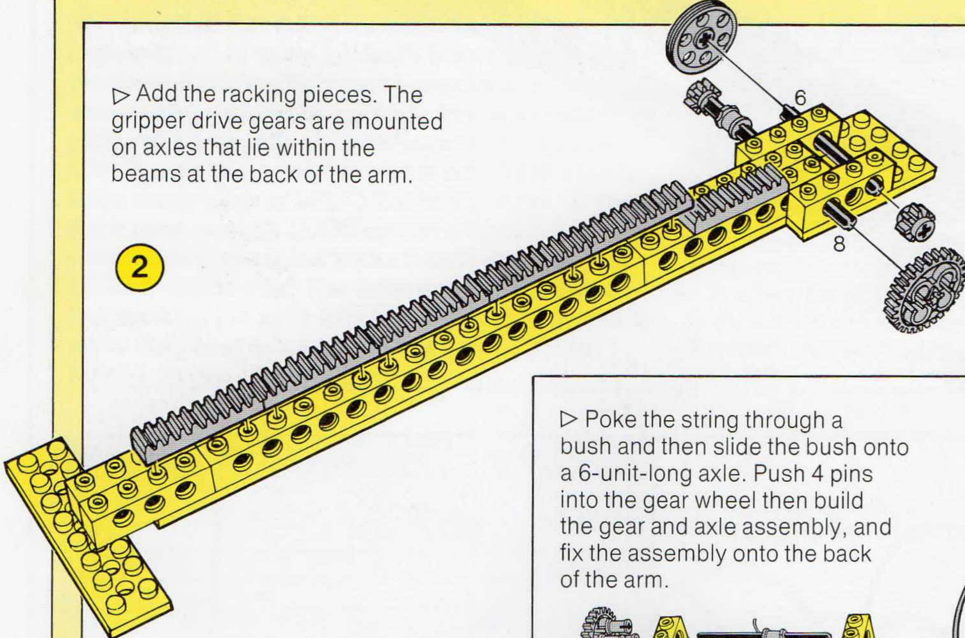


1 ▷ Construct the lower part of the arm from plates and beams.



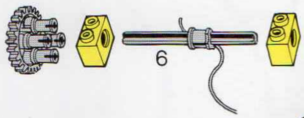
▷ Add the racking pieces. The gripper drive gears are mounted on axles that lie within the beams at the back of the arm.

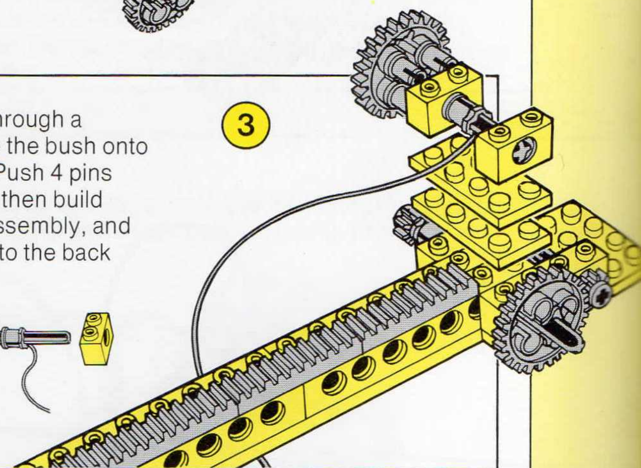
2



▷ Poke the string through a bush and then slide the bush onto a 6-unit-long axle. Push 4 pins into the gear wheel then build the gear and axle assembly, and fix the assembly onto the back of the arm.

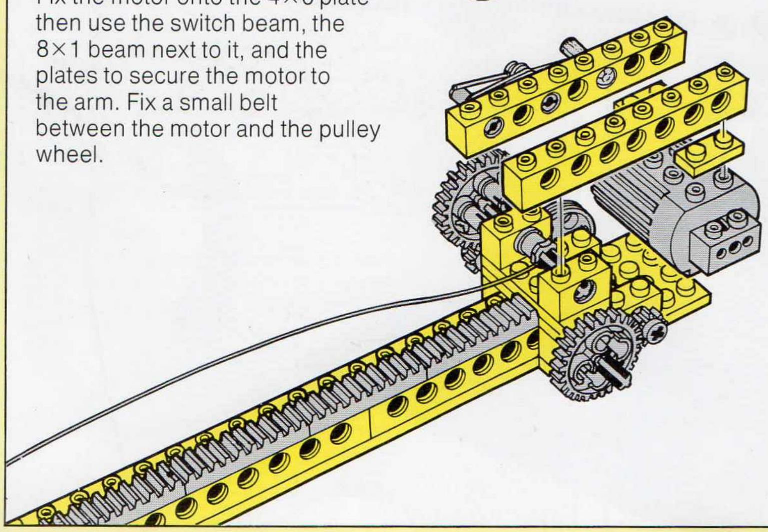
3

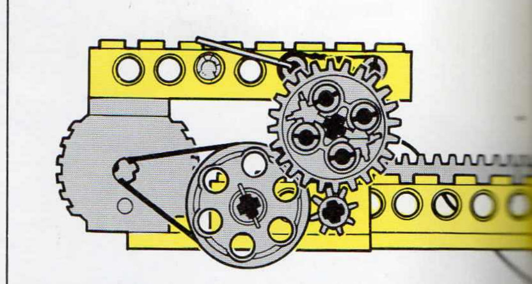




▷ Make up the switch as shown. Fix the motor onto the 4×6 plate then use the switch beam, the 8×1 beam next to it, and the plates to secure the motor to the arm. Fix a small belt between the motor and the pulley wheel.

4



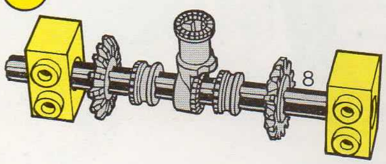


Motor/switch assembly from far side

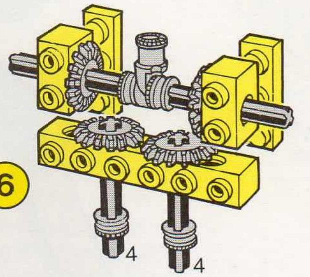
THE GRIPPER

▽Construct the main gripper axle unit ensuring that the two 14-cog gears are facing opposite directions as shown.

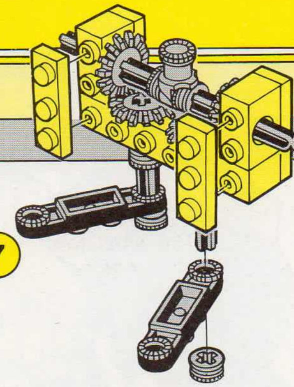
5



6

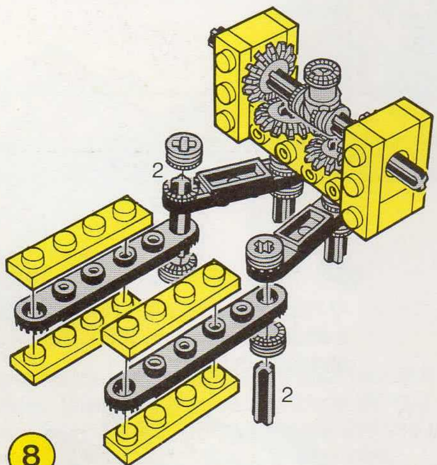


△ Add the 6-length beam, short axles, and two more 14-cog gears. Slide half-bushes part of the way up each axle.



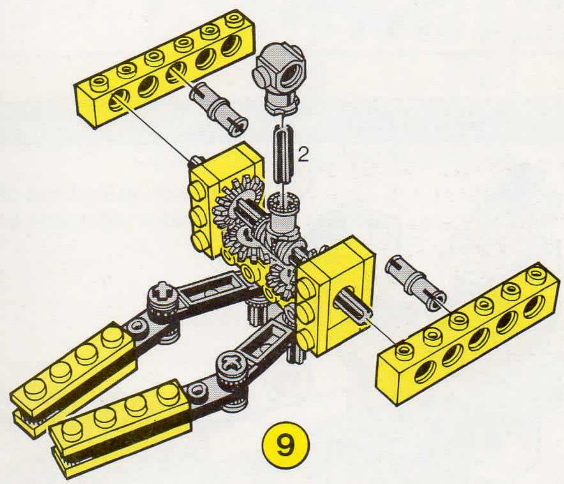
7

△On the 4-length axles, mount the 4-length steering plates, setting them in a V-shape. Lock them onto the axles using half-bushes. When the steering plates are moved in and out, the half-toggle on the top should move back and forth.



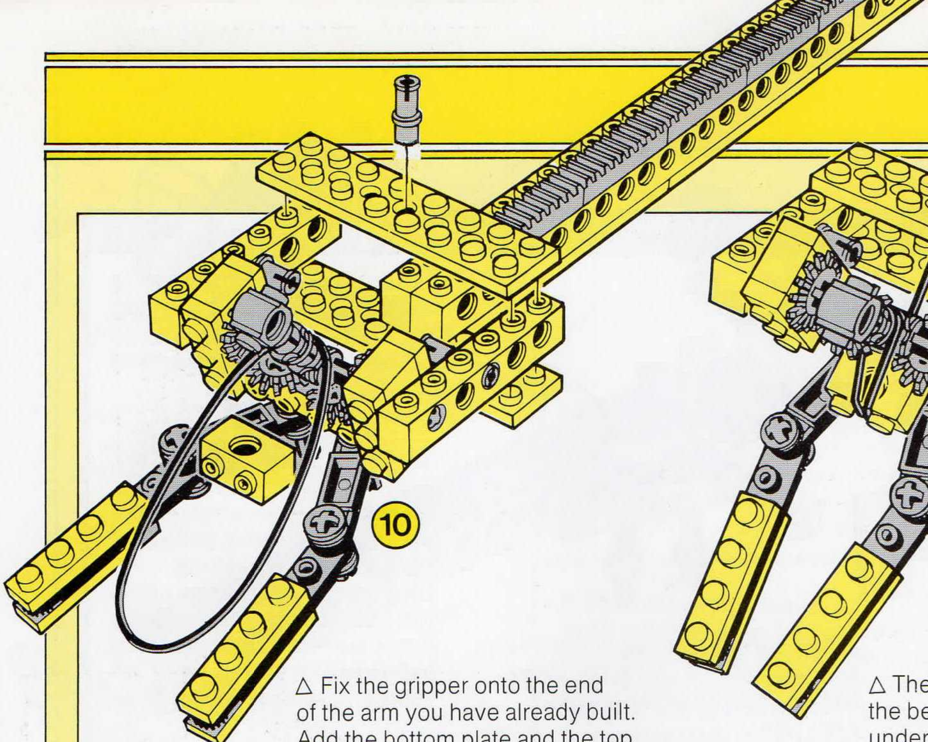
8

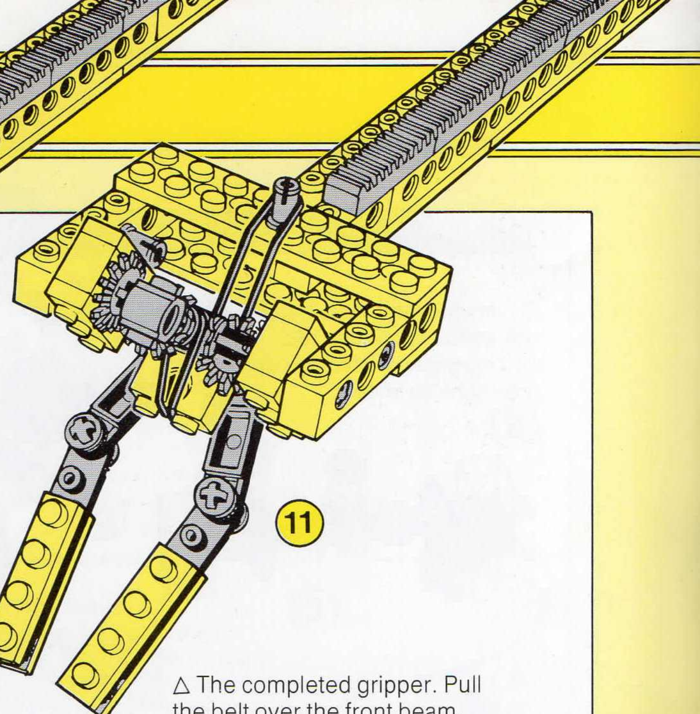
△Push 2-length axles through the ends of the short steering plates then complete the 'fingers' with 6-length steering plates and 4×1 plates. Use half-bushes on the ends of the axles to lock the steering plates together.



9

△Add the 6×1 side beams bearing pins and fit a short axle with a toggle in the top, central, half-toggle.





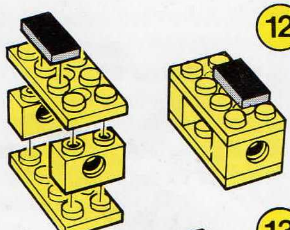
10

△ Fix the gripper onto the end of the arm you have already built. Add the bottom plate and the top plate bearing a half-pin. At the front, add the 2×1 beam. Hook a medium-size belt over the toggle unit.

11

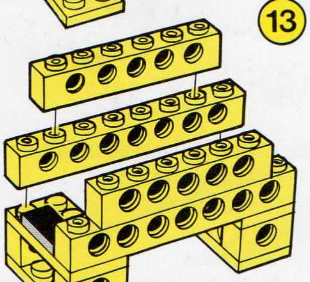
△ The completed gripper. Pull the belt over the front beam, underneath, then up onto the half-bush. The gripper should spring open in a downward position.

THE SHOULDER



12

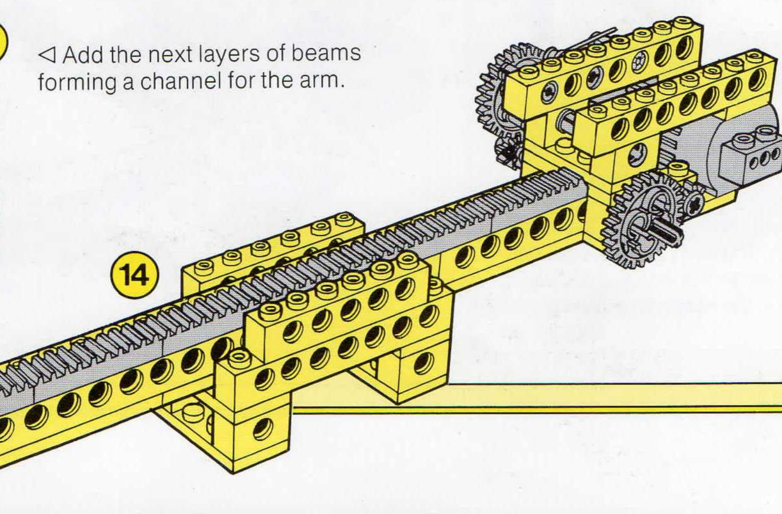
◁ Make up the base of the shoulder with plates and beams.



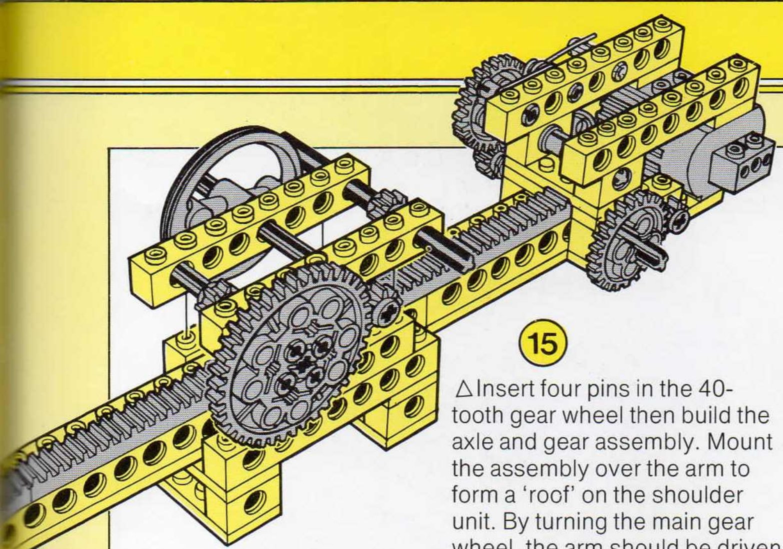
13

◁ Add the next layers of beams forming a channel for the arm.

▽ Place the arm in the shoulder. It should slide easily back and forth in the channel.

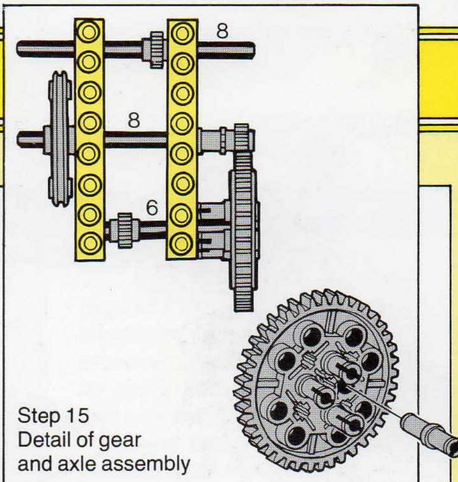


14

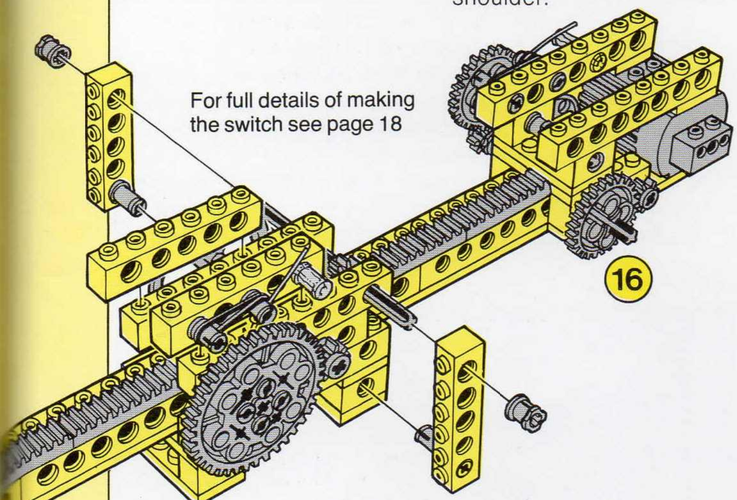


15

△ Insert four pins in the 40-tooth gear wheel then build the axle and gear assembly. Mount the assembly over the arm to form a 'roof' on the shoulder unit. By turning the main gear wheel, the arm should be driven smoothly in and out of the shoulder.



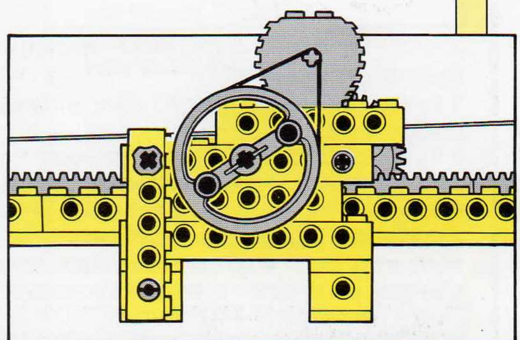
Step 15
Detail of gear
and axle assembly



For full details of making the switch see page 18

16

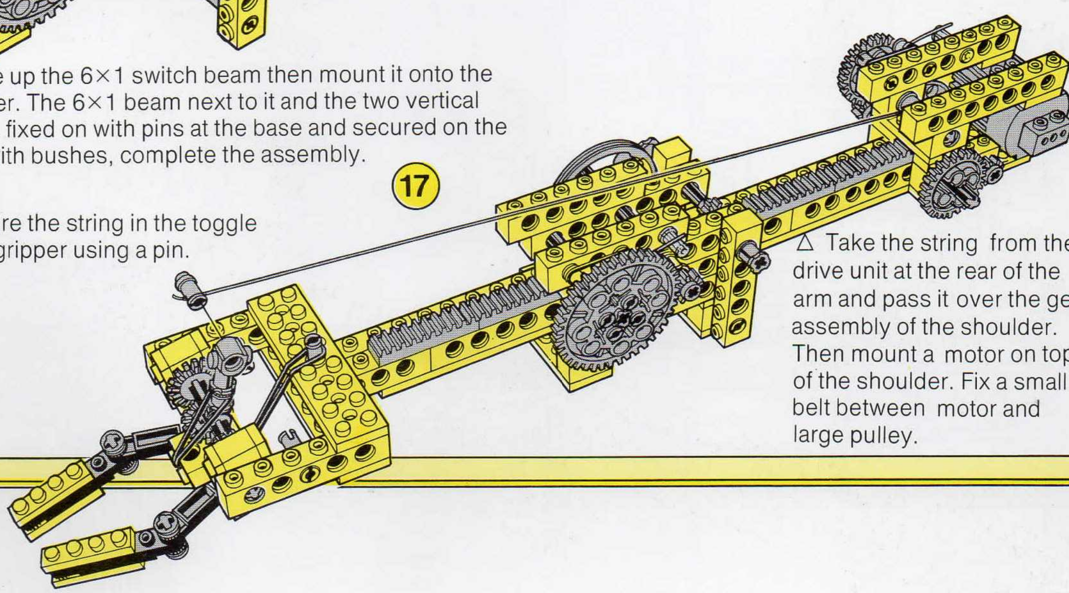
△ Make up the 6×1 switch beam then mount it onto the shoulder. The 6×1 beam next to it and the two vertical beams, fixed on with pins at the base and secured on the axles with bushes, complete the assembly.



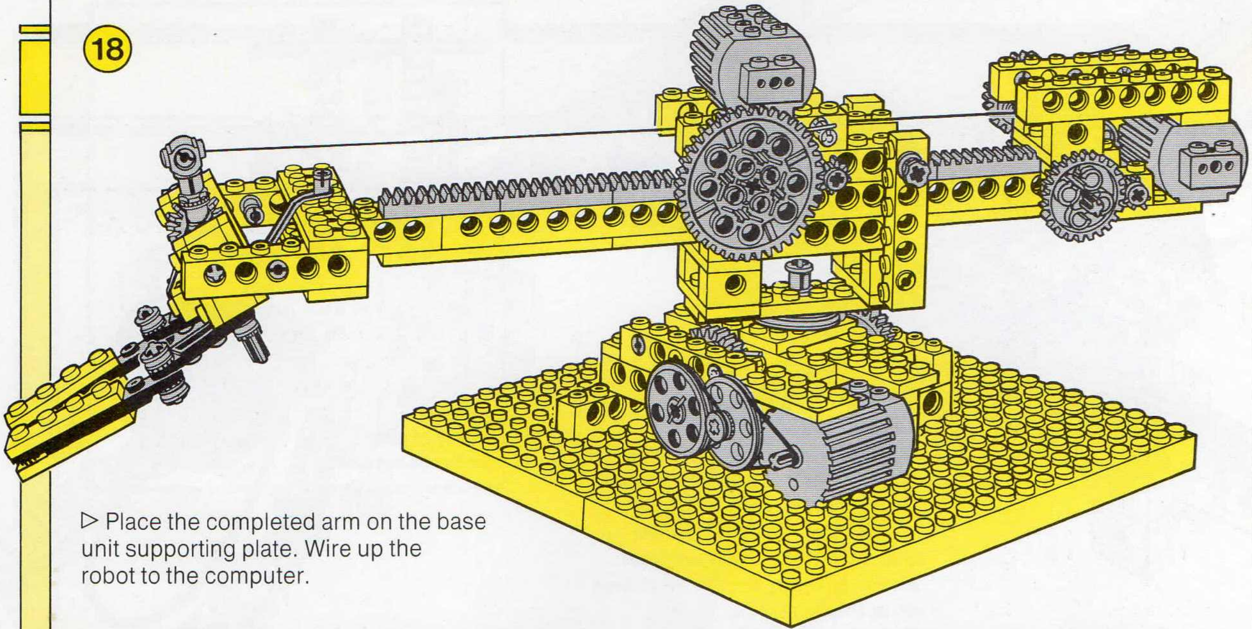
△ View of the shoulder from the side.

17

▽ Secure the string in the toggle on the gripper using a pin.



△ Take the string from the drive unit at the rear of the arm and pass it over the gear assembly of the shoulder. Then mount a motor on top of the shoulder. Fix a small belt between motor and large pulley.



▷ Place the completed arm on the base unit supporting plate. Wire up the robot to the computer.

Operating hints and tips

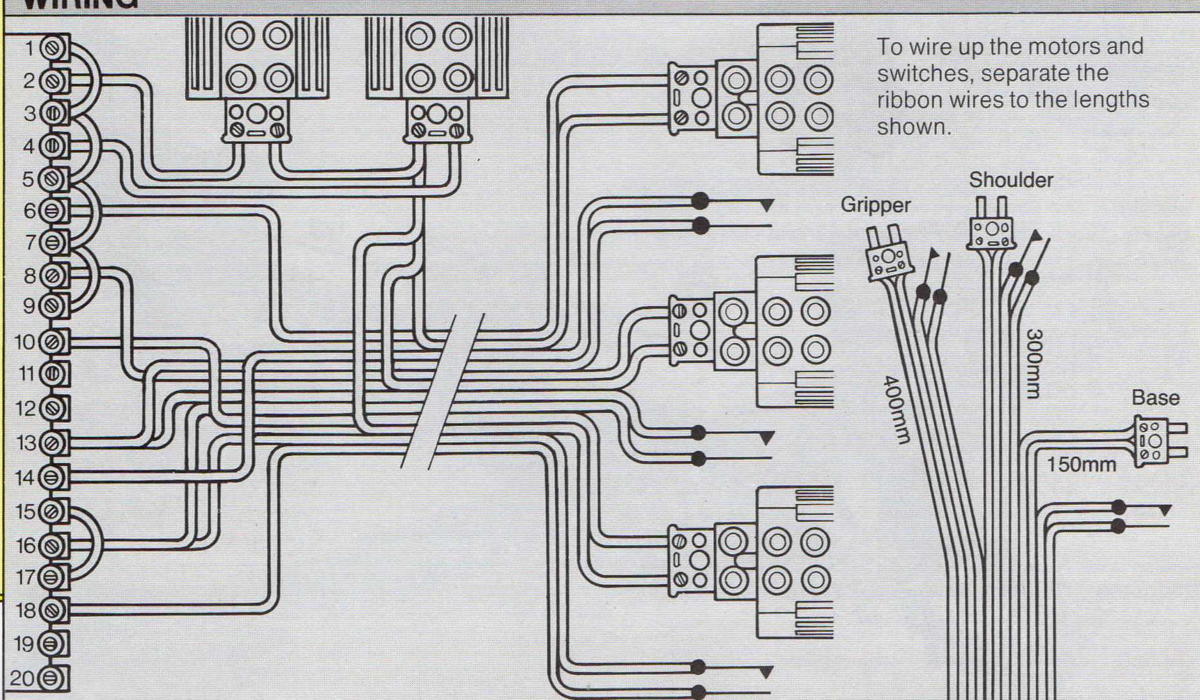
1 To allow the gripper to grasp objects more easily, use draught-excluder tape on the fingers.

2 If you interface the arm directly to your computer, you will need two battery boxes. However, if you are good with electronics, you could use the interface to switch a larger 'double-pole change-over relay' (obtainable from an electronics shop) which will act as a pole reverser, changing the direction of drive of the motors. (You will then need only one battery unit.)

3 If you use pole reversers on the two battery boxes to avoid over-running the arm, you may find that the motors will only drive in one direction. Simply change the position of the reverser on one battery box.

4 With wires going to all parts of the model, be sure that you keep the wiring neat and tidy so as not to foul any of the mechanisms. (We discuss wiring up the base unit on page 30.)

WIRING



To wire up the motors and switches, separate the ribbon wires to the lengths shown.

```
10 REM ARM CONTROLLER
20 PRT=56577:POKE56579,63
30 MB=16:MS=8:MG=4:D(0)=1:D(2)=2:SW=128
40 DIMB(50),S(50),G(50)
50 D$="█████":C$="          █████████"
60 PRINT"██POS NO.    BASE     SHOULDER  GRIPPER"
70 PRINT"███MODE:"
100 PRINTD$"█SELECT":RT=0:GOSUB900
110 IFA$="T"THENM$="█TEACH  ":RT=1
120 IFA$="H"THENM$="█HOME   ":RT=2
130 IFA$="R"THENM$="█REPEAT":RT=3
140 PRINTD$M$:ONRTGOSUB200,500,600
150 GOTO100
200 PN=1:GOSUB800
210 GOSUB900:IFA$=CHR$(13)THENRETURN
220 IFA$=CHR$(29)THENGOSUB280
230 IFA$=CHR$(157)THENGOSUB290
240 IFA$=CHR$(148)THENGOSUB300
250 IFA$=CHR$(20)THENGOSUB330
260 IFA$="S"THENGOSUB360
270 GOSUB700:GOSUB800:GOTO210
280 PN=PN-(PN<50):GOSUB380:RETURN
290 PN=PN+(PN>0):GOSUB380:RETURN
300 FORI=49TOPNSTEP-1
310 B(I+1)=B(I):S(I+1)=S(I):G(I+1)=G(I)
320 NEXTI:GOSUB380:GOSUB1000:RETURN
330 FORI=PNTO49
340 B(I)=B(I+1):S(I)=S(I+1):G(I)=G(I+1)
350 NEXTI:GOSUB380:GOSUB1000:RETURN
360 B(PN)=BC:S(PN)=SC:G(PN)=GC
370 PL=PN:PN=PN-(PN<50):RETURN
380 BN=B(PN):SN=S(PN):GN=G(PN):GOSUB1000
390 RETURN
500 GOSUB900:IFA$=CHR$(13)THEN520
510 GOSUB700:GOTO500
520 BC=0:SC=0:GC=0:PN=0:GOSUB800:RETURN
600 FORI=0TOPL:BN=B(I):SN=S(I):GN=G(I)
610 GOSUB1000:GOSUB800:NEXTI
620 GOSUB900:RETURN
700 BN=BC-(A$="3")+(A$="E")
710 SN=SC-(A$="2")+(A$="W")
720 GN=GC-(A$="1")+(A$="Q")
730 GOSUB1000:RETURN
800 PRINT"██";C$;PN,C$;BC,C$;SC,C$;GC
810 RETURN
900 GETA$:IFA$=""THEN900
910 RETURN
1000 SP=BN-BC:IFSP=0THEN1020
1010 POKEPRT,MB+D(SGN(SP)+1):GOSUB1100
1020 SP=SN-SC:IFSP=0THEN1040
1030 POKEPRT,MS+D(SGN(SP)+1):GOSUB1100
1040 SP=GN-GC:IFSP=0THEN1060
1050 POKEPRT,MG+D(SGN(SP)+1):GOSUB1100
1060 BC=BN:SC=SN:GC=GN:RETURN
1100 FORCT=1TOABS(SP)
1110 IF(PEEK(PRT)ANDSW)=0THEN1110
1120 IF(PEEK(PRT)ANDSW)=SWTHEN1120
1130 NEXTCT:POKEPRT,0:RETURN
```

PROGRAMMING

This program allows you to teach the arm a series of moves which it can then repeat. It also allows you to alter the moves by going through them one by one and changing the positions that are wrong. You can also insert and delete positions. The program displays the positions of the base, shoulder and arm and the current operating mode. The modes include Select, Teach, Home and Repeat. Type in the program, SAVE it then RUN it.

How to use the program

When you run the program, the screen will clear and the following display appear:

POS. NO. BASE SHOULDER GRIPPER

MODE
SELECT

This means the computer is waiting for you to select one of the modes by pressing:

Teaching mode–

Use	1 Q	to drive the base
	2 W	to drive the shoulder
	3 E	to drive the gripper
CRSR »		to move to next position
CRSR «		to move to previous position
INST		to insert a position
DEL		to delete a position
RETURN		to go back to the Select mode

Homing mode –
Use the keys used in the teaching mode to drive the arm. When you have set the home position, press Return.

Repeat mode –
The arm will repeat the moves automatically. When it has finished, the computer will wait for you to press a key to get back to the Select mode.

USEFUL ADDRESSES & INFORMATION

Major electronic component suppliers
Maplin Electronic Supplies Ltd,
PO Box 3,
Rayleigh, Essex,
SS6 8LR
(you will need to apply first
for an order form)
Maplin also have shops at:
Lynton Square, Perry Barr,
Birmingham
8 Oxford Road, Manchester
159-161 King Street, Hammersmith,
London W6
282-284 London Road, Westcliff-
on-Sea, Essex
46-48 Bevois Valley Road,
Southampton
The shops are open 0900 to
1730 Tuesdays to Saturdays.

Watford Electronics,
33 Cardiff Road, Watford,
Herts WD1 8ED

Also look at the adverts
in major computer magazines.

LEGO parts
For spares:
Spares Service
LEGO UK LTD,
Ruthin Road,
Wrexham,
Clwyd, LL13 7TQ

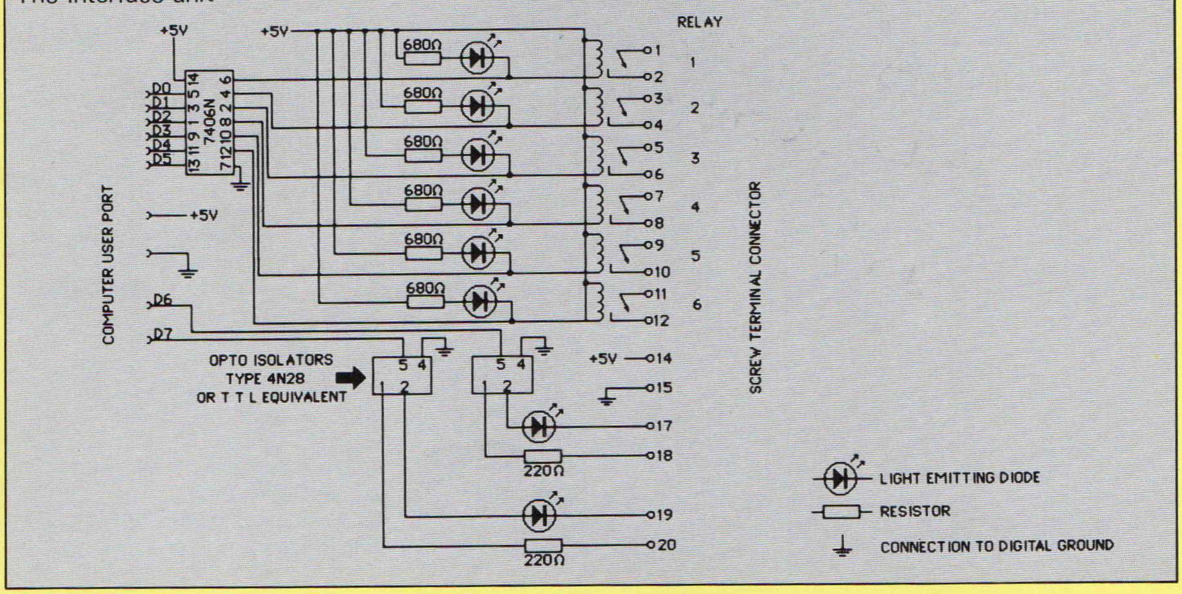
If your school is
applying for parts:
LEGO UK LTD,
Educational Department,
Ruthin Road,
Wrexham,
Clwyd, LL13 7TQ

VicRel
Write and mention this book to:
Handic Software,
5 Albert Road,
Crowthorne,
Berkshire,
RG11 7TL
Tel: 0344 778800

or from major
computer shops

The author would like to thank the following for their help in preparing this book: Ben Matthews, Malcolm Crocker, CSE Computers (Pangbourne), John Grey, Helen Seymour-Jones, Robert Solly.

The Interface unit



A Beaver Original

Would you like to make a robot? A robot that can walk about, draw pictures or pick things up? Using LEGO® building sets and your home computer, you can easily construct and control these fascinating working models in just a few hours.

You can make
- **an android**
- **a whirly turtle**
- **a lift**
- **a pen plotter**

and many more robots!

In *Make and Program Your Own Robots*, there are step-by-step instructions for building each model and simple programs to make your own robots perform. You don't need any electrical or computer skills and once you've made the models in this book, you'll be able to develop fantastic robots of your own.

UK £2.95
Australia $8.95*
Canada $5.95*
*recommended price

ISBN 0-09-942660-9

00295

9 780099 426608