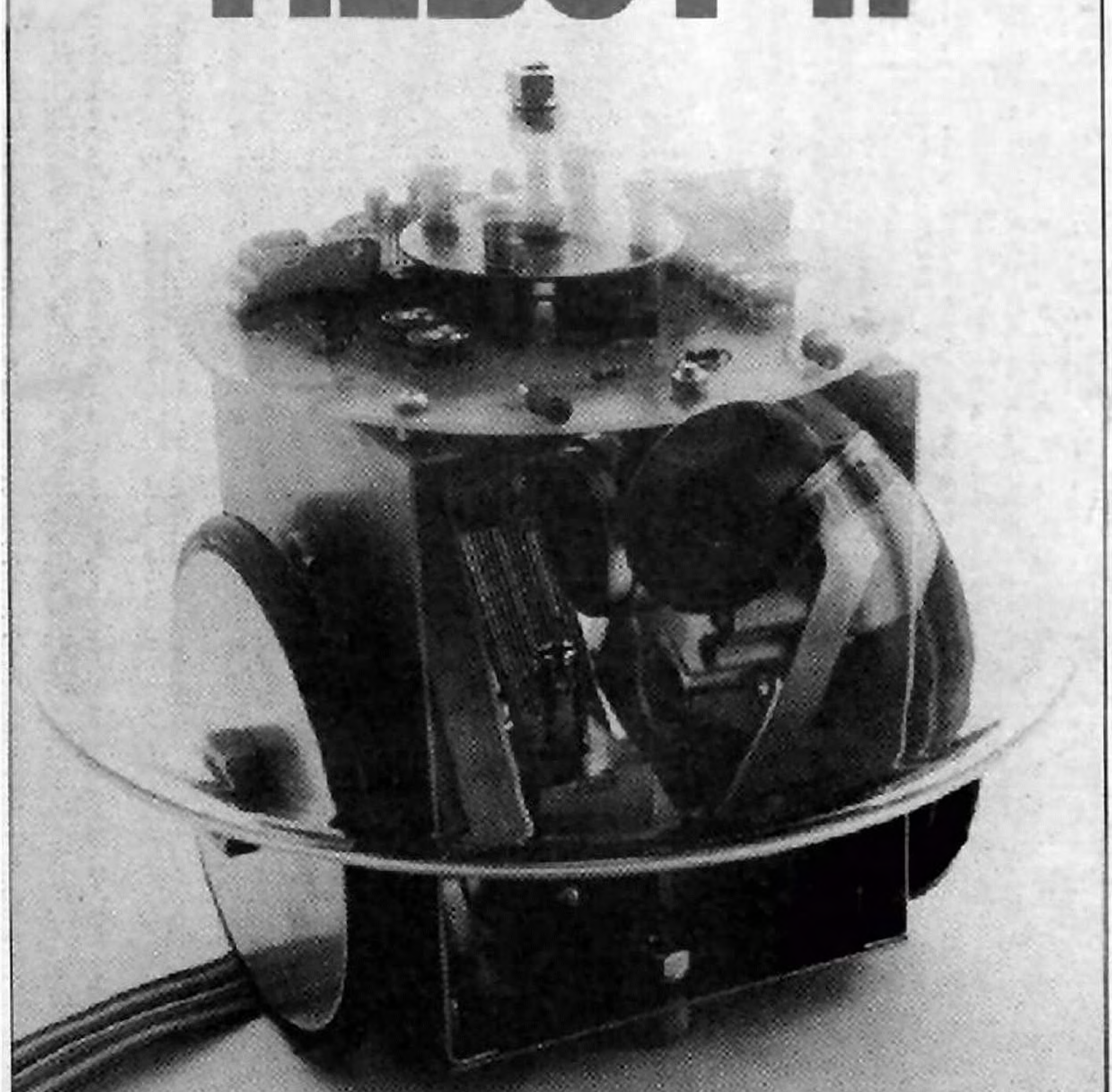
HEBOT II



Most people have hobby horses –
now here's the Hobby Robot!

THE IDEA of a mechanical man, or "robot", is an ancient one, dating back to the time of the Egyptian Pharaohs. The word itself was coined by the writer Karel Kapek; it comes from the word *robota* in his native Czech, meaning "labourer" or "worker". Kapek's fictional invention undoubtedly owed a debt to the legend of the Golem, a walking figure formed from the clay of the earth by occult powers; the story of Frankenstein, although written by an English gentlewoman, is also derived from the legends of Bohemia and Transylvannia and offers another point of departure for the invention of the robot. Perhaps these shadowy origins account for some, at least, of the fear and loathing with which the robot is sometimes regarded!

This was not always so; in the past, clockwork mechanisms delighted the crowds at fairs and carnivals, while more elaborate toys and models were built for the pleasure of the well-to-do.

Apart from these, however, the robot has been a total figment of science fiction, particularly in the works of Issac Asimov, who created a virtual science of Robotics in his many novels on the theme. Another writer, whose fictional approach came closer to the modern realisation of the robot, was Robert Heinlein. His "Waldoes" were giant manipulators with the strength of an hydraulic forge, linked to special controls in the form of arm-length gauntlets; the Waldo mechanisms simply duplicated the movements of the operator, transmitted via the gauntlets.

Like most good science fiction, the works of Asimov and Heinlein were soundly based on real science. At about the time the authors were formulating their ideas, Norbert Weiner was defining the science of Cybernetics as "The field of control and communications, whether in the machine or in the animal", and the British mathemetician Turing was laying down the standards for assessing artificial intelligence, ie computer or robot intelligence. Then, too, a man named Thomas Ross built the first "robot mouse", the prototype of all maze-solving turtle-type robots.

All this took place in the 1930s, but subsequently robotics developed at a very slow pace until the late 70s, when useful industrial robots became a practical proposition. In the past ten years, of course, robots have become an exceedingly hot topic in the national press, to the point where they are proclaimed the keystone of the Second Industrial Revolution and Britain's economic revival.

It was at the start of the revival of interest in robotics that HEBOT, Hobby Electronics' first robot, appeared. The "Amiable Automaton" was, by today's standards, a simple device, entirely controlled by the on-board 'hardware'. In fact HEBOT represented no great *operational* improvement on Ross' original micro-mouse, though of course the control system was both simpler and more versatile due to the use of integrated circuit technology not available to Ross in 1938!

Son of HEBOT

Three years on and HEBOT II, while in many ways a very similar beastie, is considerably improved both in its control system and in operation, because it is designed to be used in conjunction with a microcomputer. Specifically, it is designed to be run by a ZX81, but the control system is quite simple and any microcomputer

How It Works

HEBOT II is controlled by eight data lines derived from the ZX81 computer's data bus via an interface board.

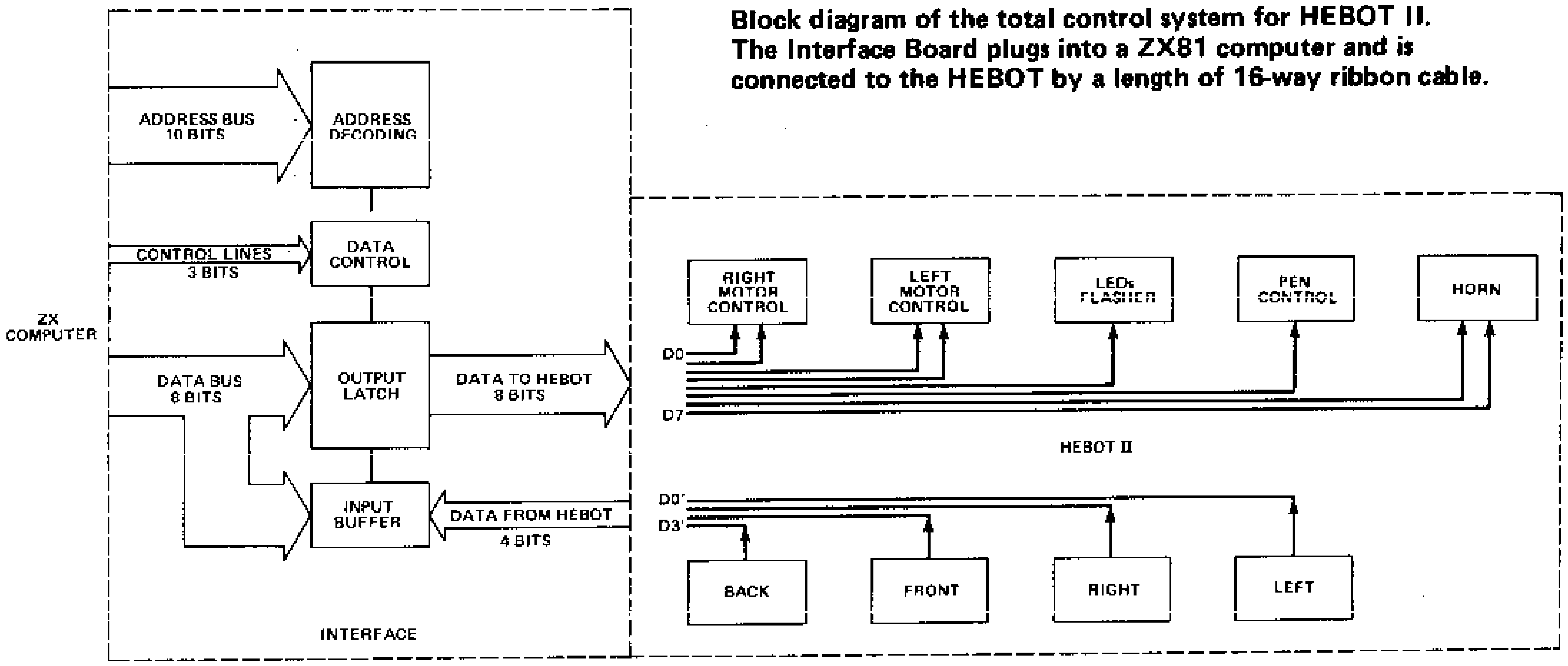
Bits 0 and 1 control the right motor drive circuits. When bit 0 is high and bit 1 is low, the motor will drive in the forward direction; when the bit pattern is reversed (D0 low, D1 high), the motor drives in reverse. When both bits are low OR both are high, the motor is stopped. The left motor is controlled by bits 2 and 3, in identical fashion, so that by turning on, or off, different combinations of bits 0-4, HEBOT can execute quite a variety of movements.

The LEDs on the robot are switched on when bit 4 goes high, and bit 5 turns on the solenoid, forcing down the centrally-mounted pen. Finally, bits 6 and 7 control HEBOT's horn; a low tone sounds when bit 6 is high, increasing in pitch when bit 7 is taken high.

The action of the control bits, especially movement controls, is summarised in **Table 1.**

Four microswitches are mounted on the robot's PCB, mechanically connected to the 'shell' so that they operate as collision detectors. They are directly linked to four output data lines, D0'-D3', and thence to the computer data bus via the interface board.

The interface board itself consists of four circuit blocks; the address decoder operates on the top ten address lines to produce an output only for a certain bit-pattern, corresponding to a particular address — the address of HEBOT II. The control circuitry determines, in conjunction with the output from the address decoder, whether the computer is writing data to HEBOT or reading data from it. The output latch accepts data from the computer at specific times and stores it until the next time data is sent to the machine. The input buffer transmits data from the robot when the computer is ready to accept it.



Block diagram of the total control system for HEBOT II. The Interface Board plugs into a ZX81 computer and is connected to the HEBOT by a length of 16-way ribbon cable.

Table 1. HEBOT II data input bus.

RM1 = Right motor forward
RM2 = Right motor reverse
LM1 = Left motor forward
LM2 = Left motor reverse

L = Lights on
P = Pen down
H = Horn on
T = High tone

D7	D6	D5	D4	D3	D2	D1	D0	DATA BITS
T	H	P	L	LM2	LM1	RM2	RM1	HEBOT CONTROL
128	64	32	16	8	4	2	1	DECIMAL VALUE

LM2	LM1	RM2	RM1	DECIMAL	LM	RM	MOVEMENT
0	0	0	0	0	off	off	stop
0	0	0	1	1	off	fwd	turn left about left side
0	0	1	0	2	off	rev	turn right about left side
0	0	1	1	3	off	off	stop
0	1	0	0	4	fwd	off	turn right about right side
0	1	0	1	5	fwd	fwd	forward
0	1	1	0	6	fwd	rev	turn right about centre
0	1	1	1	7	fwd	off	turn right about right side
1	0	0	0	8	rev	off	turn left about right side
1	0	0	1	9	rev	fwd	turn left about centre
1	0	1	0	10	rev	rev	backward
1	0	1	1	11	rev	off	turn left about right side
1	1	0	0	12	off	off	stop
1	1	0	1	13	off	fwd	turn left about left side
1	1	1	0	14	off	rev	turn right about left side
1	1	1	1	15	off	off	stop

Table 1 (above) shows the data control lines to HEBOT II. Data bits D0 — D3 control the direction of movement, and various bit-patterns permit a great variety of movements, particularly in turning motions. The remaining bits, D4 — D7, control the lights (LEDs), the horn and its pitch, and the position of the solenoid (up or down). In practice, HEBOT is controlled by POKEing the decimal value corresponding to the required action or combination of actions; eg, POKE (address), 5+64 commands forward movement, with the horn sounding. See Tables 3 and 4 for further examples.

Table 2 (below) shows the sensor data from HEBOT. PEEKing the robot's address will return a decimal value corresponding to a specific sensor, if a single-point collision has occurred. A 'touch' activating two sensors will return a decimal value corresponding to the combination of sensors.

Table 2. HEBOT II data output bus.

D7'	D6'	D5'	D4'	D3'	D2'	D1'	D0'
—	—	—	—	Back Touch	Front Touch	Right Touch	Left Touch

TOUCH SENSORS ACTIVATED	BINARY	DECIMAL
None	00000000	0
Left only	00000001	1
Left and front	00000101	5
Left and back	00001001	9
Right only	00000010	2
Right and front	00000110	6
Right and back	00001010	10
Front only	00000100	4
Back only	00001000	8

capable of input/output operations, whether via a dedicated I/O port or via an expansion port (as used by the ZX) can be used to control the machine. A memory-mapped interface board for the ZX81 computer is presented as part of this project, and further issues of Hobby Electronics will explore the possibilities of other interface boards and computers. *(We would welcome readers' submissions on this subject! — Ed.)*

The significance of computer control is that, whereas the opriginal HEBOT had only a small 'library' of hard-wired routines, HEBOT II has almost unlimited capabilities within the restraints of available memory and computer speed.

Programming In The Real World

Not only does the use of computer-control greatly increase the capabilities of HEBOT II but the machine itself takes programming out of the two dimensional world of the VDU into the real, three-dimensional world. It can perform a bewildering number of moves under program control — forward, backward, left and right — with each wheel independantly controlled. Programs can be developed so that HEBOT can sense its environment via the obstacle sensing switches coupled to its 'shell', allowing the most devious "aviodance routines" to be devised to solve a maze or map the shape of a room. It can even report directly on its environment, via the blinking LEDs and the two tone horn,and one of the most interesting possibilities is the use of the built-in pen, which can be forced down onto a sheet of paper or artboard, to draw graphs or outline sketches.

Two simple programs are listed, later, to illustrate the tremendous potential of this machine. Both routines are given as 'starting points', because they are very, very basic; this will quickly become obvious and at this point, it is left to the reader to develop more useful routines! However, HEBOT II will be fully functioning on improved programs at this year's Breadboard Exhibition, at the Royal Horticultural Society's New Hall, Greycoat Street, Westminster, London SW1, so come along and see it for yourself!

The first program is a simple 'walk and avoid' routine; when it is LOADed and RUN, HEBOT II will "proceed in a forward direction" (m'lud), flashing its 'eyes' until it encounters an obstacle. Then it will back off, sounding a note, turn left and continue forward until the next obstacle is encountered.

The second routine is a basic "learning" program which will allow up to five movement commands to be stored and repeated indefinitely. This program could form the basis of a routine for drawing patterns or graphs on a large piece of paper or board,

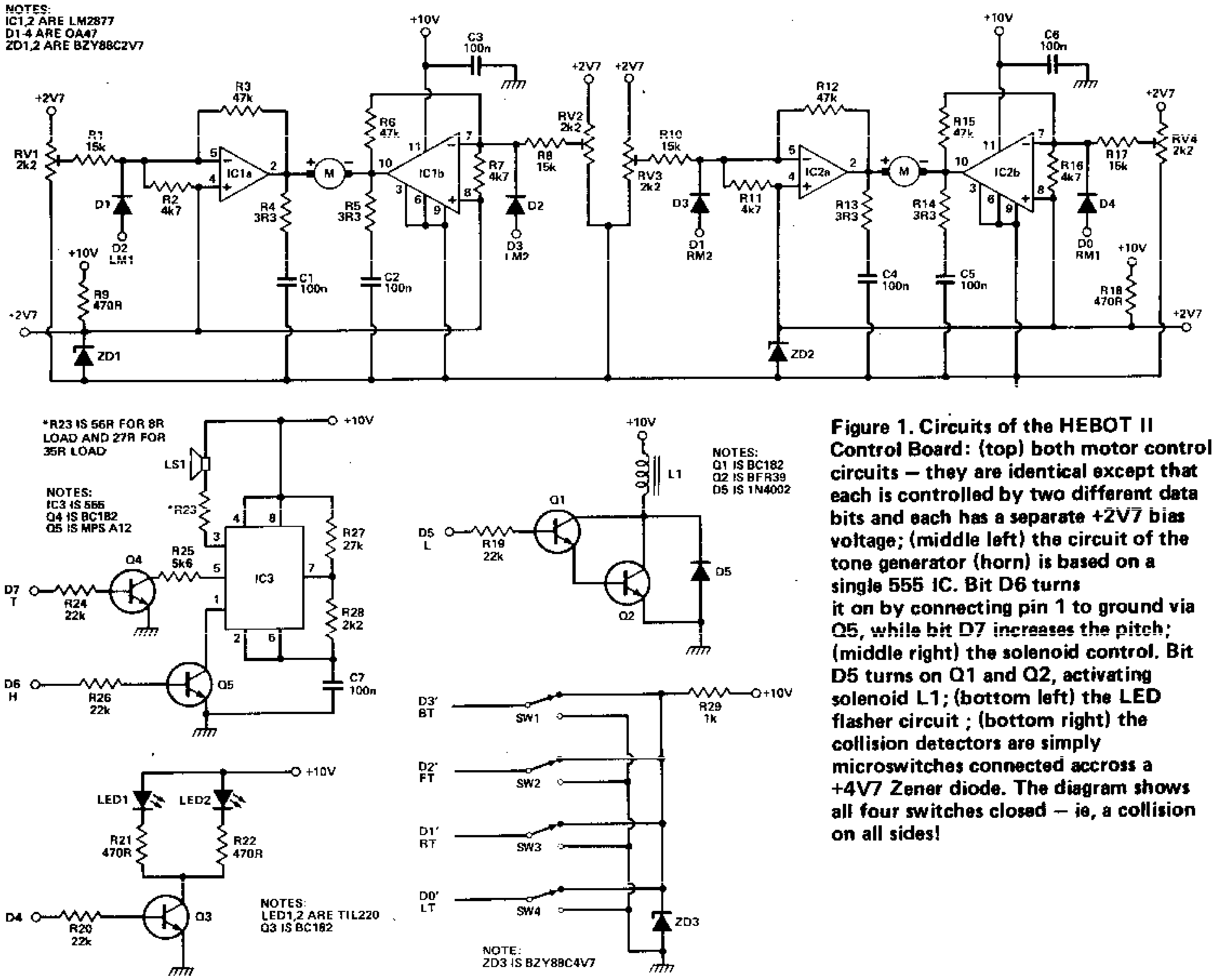


Figure 1. Circuits of the HEBOT II Control Board: (top) both motor control circuits — they are identical except that each is controlled by two different data bits and each has a separate +2V7 bias voltage; (middle left) the circuit of the tone generator (horn) is based on a single 555 IC. Bit D6 turns it on by connecting pin 1 to ground via Q5, while bit D7 increases the pitch; (middle right) the solenoid control. Bit D5 turns on Q1 and Q2, activating solenoid L1; (bottom left) the LED flasher circuit ; (bottom right) the collision detectors are simply microswitches connected accross a +4V7 Zener diode. The diagram shows all four switches closed — ie, a collision on all sides!

using the pen control. The scope of this routine is severely limited by available memory — five commands are the most that can be squeezed into what's left of the 1k of a basic ZX81!

HEBOT — The Circuit

The circular PCB mounted atop HEBOT contains all the circuitry for the motor drives, the LED drivers and the tone generator, plus the collision detectors — which are simply microswitches mechanically coupled to the shell. All circuits are shown in **Figure 1**.

Each motor is driven by a pair of high gain operational amplifiers contained in a single package. The ICs chosen for the design are LM2877s, dual four watt devices with internal current limiting, short circuit and thermal shutdown protection. The amplifiers are internally compensated to maintain stability for gains greater than ten however, because a gain of about three is all that is required, input shunt resistors are used to provide stability with the gain at this level.

The drive circuits are identical for each motor. Considering IC2a, the gain is set by R10, R12; R11, which is connected across the inputs, limits the gain at high frequencies to give improved stability.

The voltage on the non-inverting input, pin 4, is set precisely at 2V7 by R18 and the Zener diode, ZD2. This stabilised voltage is also fed to the inverting input at pin 5, via the trimpot RV3, and when data input D1 is low (0V), this voltage is amplified to about 6-7V by the op-amp and applied to the motor. If, at the same time, the D0 input is taken high, pin 7 of IC2b is taken above the voltage set by RV4, to about +5V, and the output at pin 10 goes hard towards 0V (actually about 1V6), allowing the motor to drive.

Similarly, if D1 is taken high and D0 low, pin 10 of IC2b goes to about 6V and pin 2 of IC2a goes to nearly 0V — and the motor drives in the opposite direction. However, if both data inputs are the same (both high or both low), then the op-amp outputs will be the same and the motor will not turn.

Thus these simple op-amp circuits convert logic levels into reversible and independently asjustable voltages; further, since the motor drive voltages are derived from a Zener stabilised supply, they will remain independent of variations in the supply line voltage. The RC networks on the outputs of the op-amps are 'Zobel networks', to further ensure high frequency stability.

The remaining circuits are very simple; data bit four (D4) turns on Q3 when high, thus turning on LEDs 1 and 2. The solenoid is controlled by D5; Q2 is a high current transistor which receives its basic drive from Q1 when D5 is high. The two-tone horn consists of a 555 IC operating as an astable and driving a loudspeaker. Transistor Q5 will switch on whenever D6 is high, thus connecting pin 1 to ground allowing it to oscillate at about 500 Hz; the frequency is increased to 1000 Hz by taking D7 high under program control.

On the output side, the microswitches simply switch from 0V to +4V7 whenever a collision occurs.

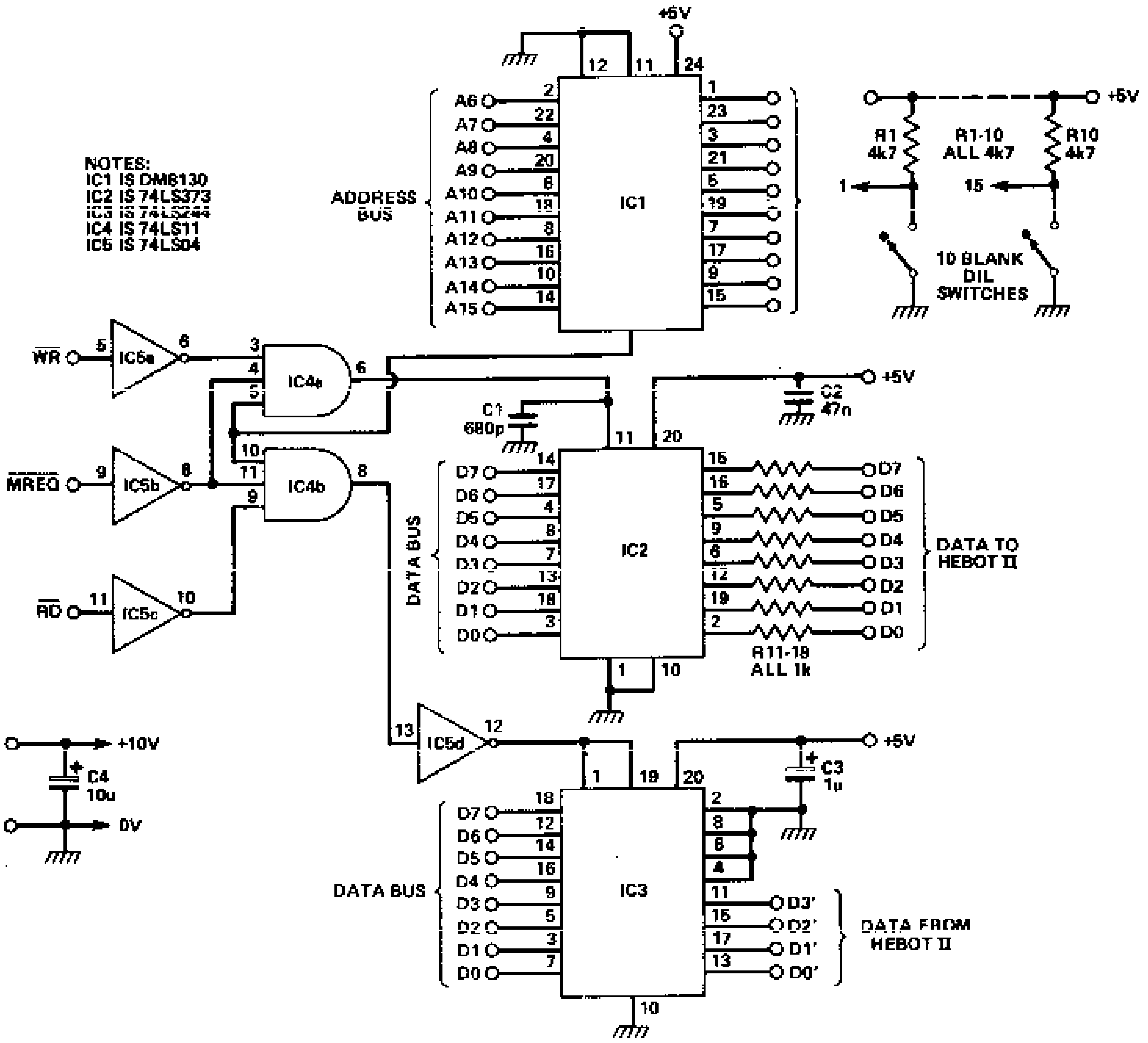


Figure 2. The circuit of the ZX81 Interface Board. Resistors R2 — R9 are contained in a nine-pin SIL (Single-In -Line) package.

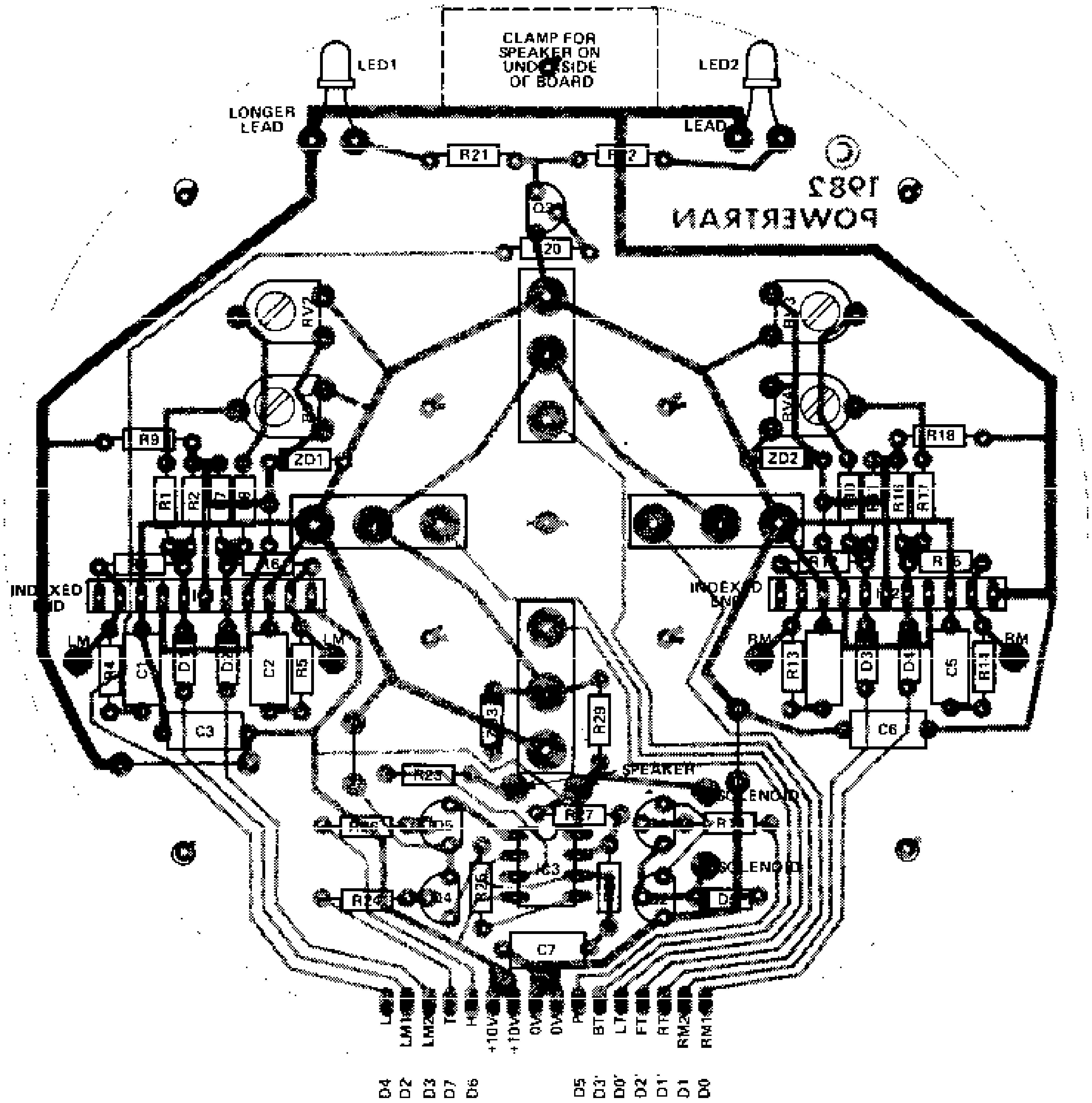


Figure 3. The component overlay board for the HEBOT Control Board; the LEDs indicate the "forward" direction.

The Interface Circuit

This circuit (**Figure 2**) enables HEBOT II to be treated as a memory-mapped I/O device; in other words, data can be sent to and received from the robot as if it were another memory location in RAM or ROM. It uses the computer's address and data busses, together with the control lines $\overline{WR}$ (active low WRite command), $\overline{RD}$ (ReaD) and $\overline{MREQ}$ (Memory REQuest).

As briefly explained in How It Works, the interface board consists of four circuit elements. The address decoder is a comparator, IC1, which compares the top ten address lines with a ten-bit code set up on the ten-bank DIL switch and resistor network. Thus by setting these switches, any one of 1024 64-byte wide memory sectors can be selected as HEBOT's location in memory. The most convenient address is 65535, right at the top of the memory space, corresponding to all DIL switches open (ie, all address bits high).

Whatever address is selected, IC1's output at pin 13 will go high only when the top ten address bits correspond to the code set up on the switches. When there is a match, and when both the $\overline{WR}$ and $\overline{MREQ}$ lines are low, data latch IC2 will be enabled and the bit-pattern on the computer's data base will be tranferred to HEBOT's data input lines. After this, the data will remain latched in IC2 until a new WRite instruction to HEBOT is issued by the computer.

Simillarly, IC3, which is an eight-bit buffer IC, will transfer data to the computer when both $\overline{RD}$ and $\overline{MREQ}$ are low and when there is an address match from IC1.

Construction

Start by assembling the HEBOT PCB (**Figure 3**): the only points to watch here are that the microswitches are fitted square and firmly mounted, and that the ICs are fitted the right way around. To help, the makers have put an index mark on the package at the pin 1 end and, in case you miss that, they have also cut the corner off the cooling fin! Next, wire in the 16-way ribbon cable which connects HEBOT to the interface band. The remaining components can then be fitted and the PCB completed.

Now for the collision detector —see **Figure 4**. First insert the central mounting screw, which will eventually hold the shell in place. Next assemble the mounting plate mechanism and screw it loosely to the PCB; the small ball-bearing must be slipped under the mounting plate, where it will be held between the depression in the PCB and the bottom of the central mounting screw — this is probably the trickiest part of the entire assembly! Now tighten down the four mounting screws, but then unscrew them about half a turn, to permit the plate to rock slightly. Pressing down on one side should operate one or two of the microswitches, and they should

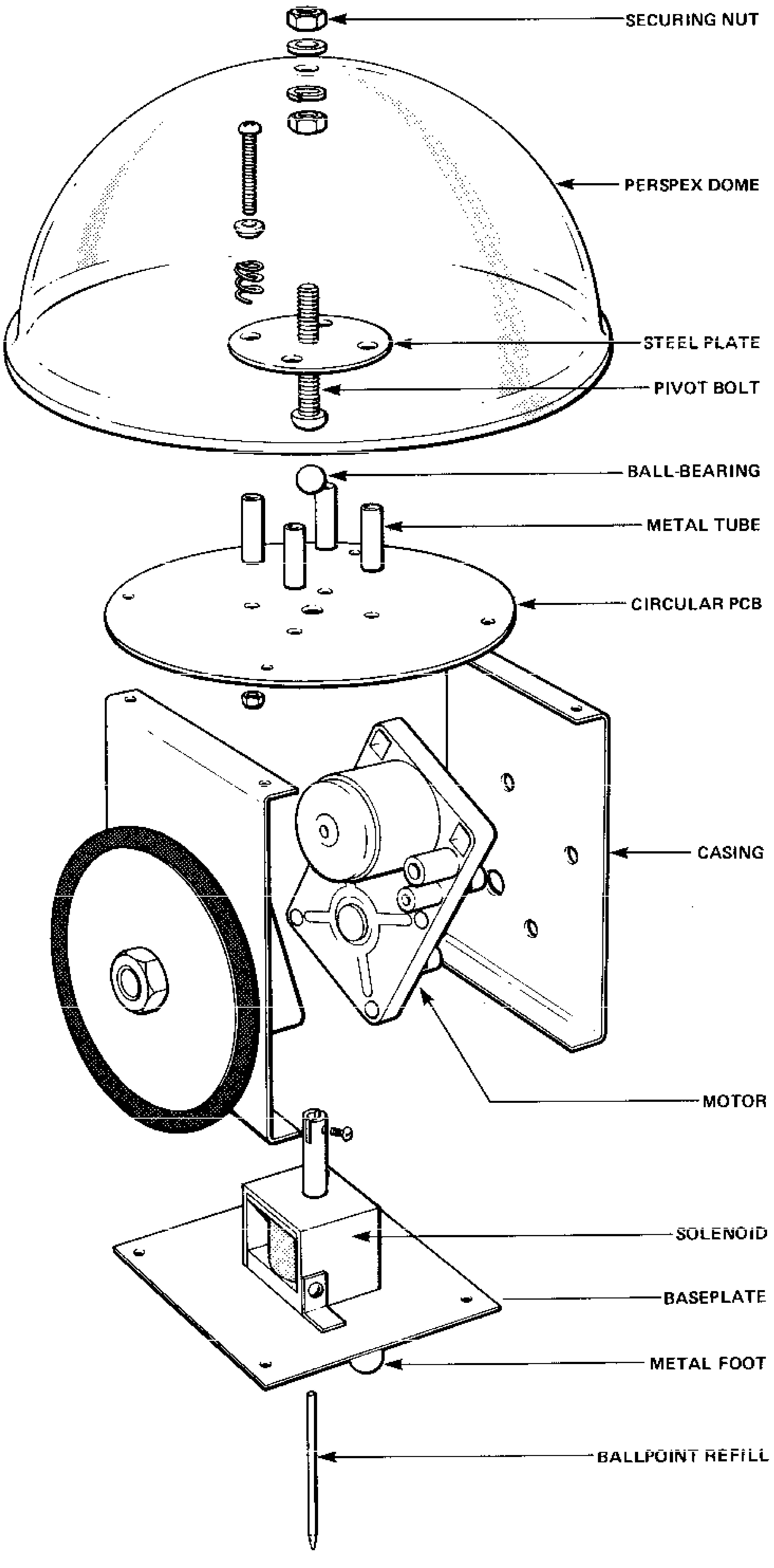


Figure 5. The HEBOT mechanical assembly diagram.

release when the pressure is removed from the plate. After this, you can fit the 'speaker to the underside of the PCB using the special clip, and wire it in.

The next stage is to fit the motors to the side plates (see **Figure 5**) and to attach these to the base plate. Mount the PCB to the side plates using stand-off spacers, and wire up the motors; the LEDs and the speaker are regarded as being the front of the robot. Secure the pen in the solenoid — **Figure 6** — and fit it to the base plate; fix the wheels to the motor shafts with a nylon washer between the wheels and the plates and then adjust the pen so

Parts

A complete kit of parts for HEBOT ii, consisting of the components and hardware listed below, is available from **Powertran Cybernetics** — see **Buylines** for details. The components for the Interface Board (including the double-sided PCB), the 23-way connector and the three-way RAM pack adaptor are all available separately — see page 33.

I/O Board

RESISTORS

(all ¼ watt 5% carbon, unless stated)

R1,10	4k7
R2-8	4k7 SIL package
R11-18	1k

CAPACITORS

C1	680p ceramic disc
C2	47n ceramic disc
C3	1u 16V tantalum bead
C4	10u 16V tantalum bead

SEMICONDUCTORS

IC1	DM8130 10-bit address decoder
IC2	74LS373 octal latch
IC3	74LS244 tri-state octal buffer
IC4	74LS11 triple 3-input AND
IC5	74LS04 hex inverter

MISCELLANEOUS

SW1-10	DIL switch 10 way rocker type

PCB; cable clip; 14 pin (2 off), 20 pin (2 off), 24 pin DIL sockets; 23 + 23 way edge connector, polarising key (posn 3), PCB mounting (2 off); 16 way ribbon cable.

Main Board and Mechanics

RESISTORS

(All ¼ watt 5% carbon)

R1,8,10,17	15k

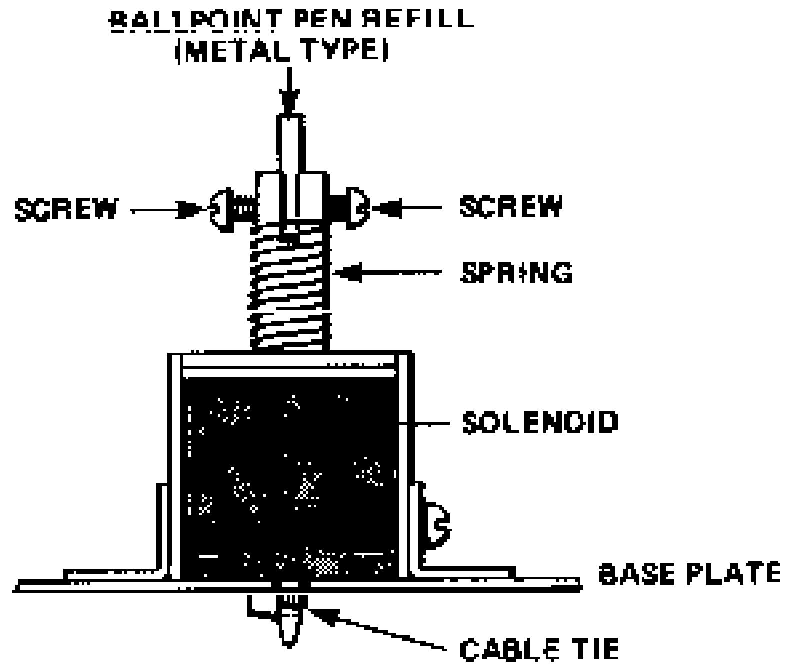


Figure 6. Mechanical detail of the solenoid. Note the cable tie, which holds the pen in place.

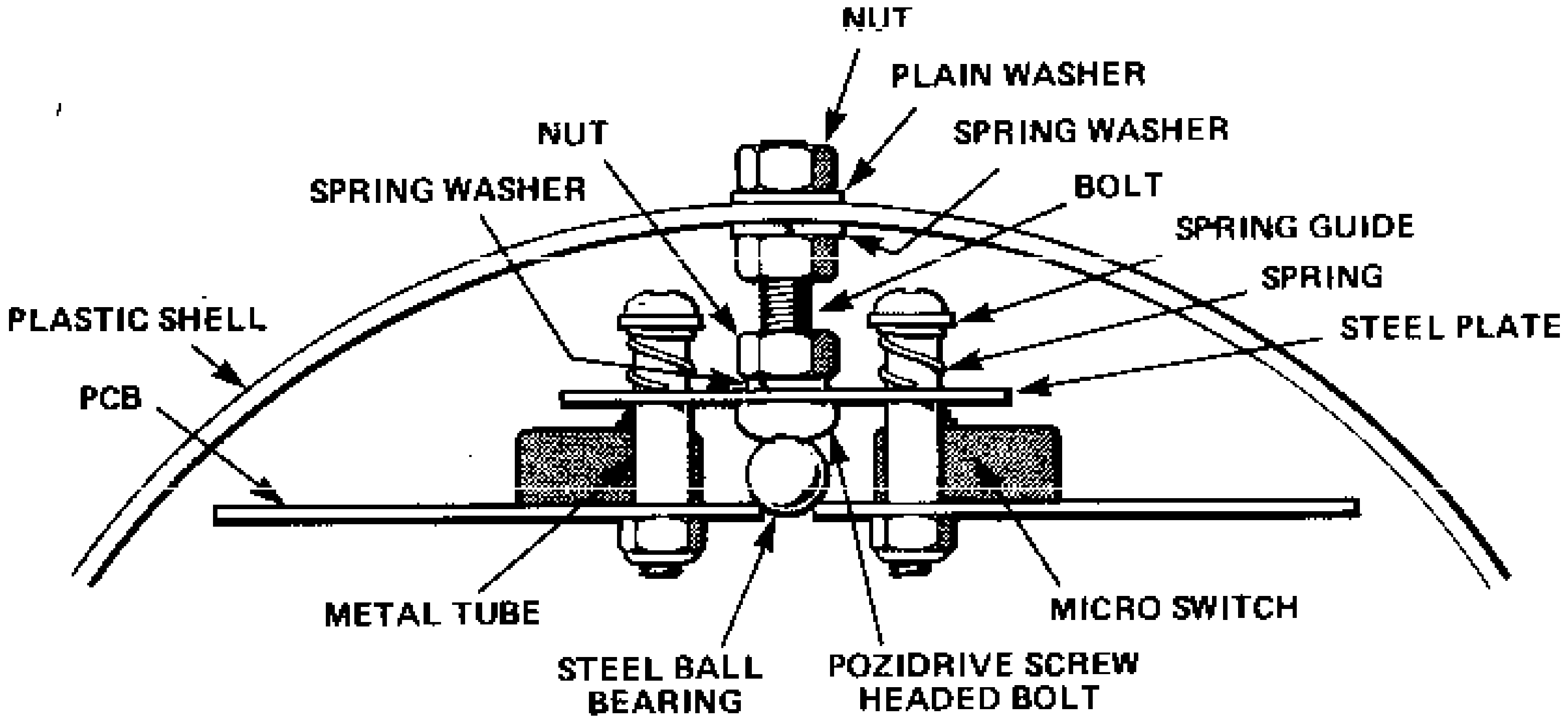


Figure 4. How to assemble the collision detector mechanism.

List

R2,7,11,16	4k7
R3,6,12,15	47k
R4,5,13,14	3R3
R9,18,21,22	470R
R19,20,24,26	22k
R23	(see text)
R25	5k6
R27	27k
R28	2k2

POTENTIOMETERS
(All miniature carbon presets)

RV1-4	2k2

CAPACITORS
(All polyester C280)

C1-7	100n

SEMICONDUCTORS

IC1,2	LM2877 dual 4 watt power amp
IC3	NE555 timer
Q1,3,4	BC182 silicon NPN transistor
Q2	BFR39 silicon NPN transistor
Q5	MPSA12 silicon NPN transistor
D1-D4	OA47 signal diode
D5	IN4002 rectifier diode
LED1,2	TIL20 red 0.2" LED
ZD1,2	BZY88C2V7 400mW zener diode
ZD3	BZY88C4V7 400mW zener diode

MISCELLANEOUS

L1	solenoid (see text)
LS1	speaker 8-35R 1½"

PCB; micro-switch (4 off); 8 pin DIL socket; motor with integral gear box — 2 off; aluminium sheet; wheels; toes; sprints; steel ball-bearing; clear plastic shell; nuts; bolts; wire; solder etc.

BUYLINES .... page 33

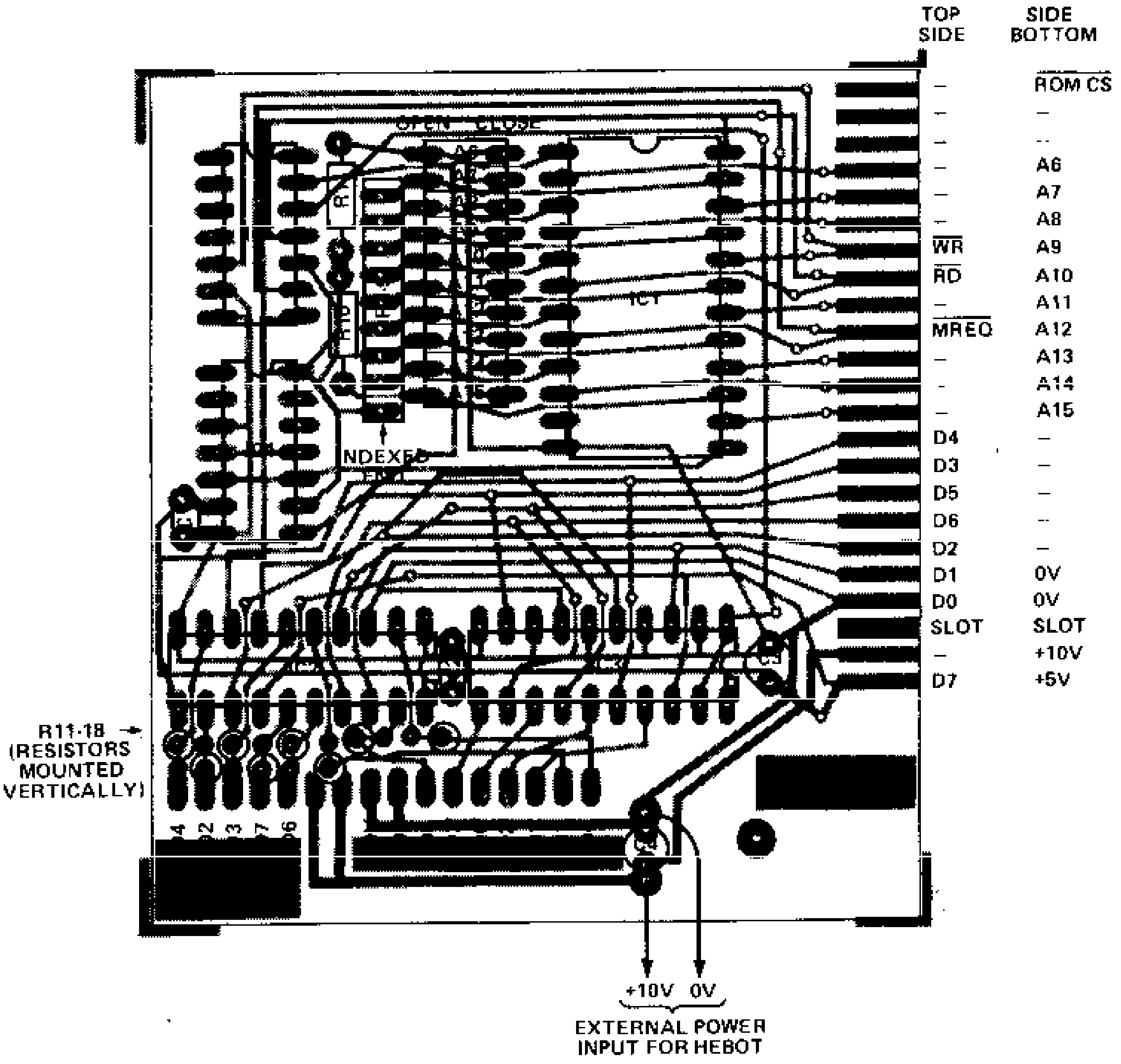


Figure 7. The component layout of the Interface Board. Note that both sides of the double-sided PCB are shown.

that, when it is fully down, it is about 2mm below the level of the tyres. To prevent the spring from ejecting the pen when the solenoid is deactivated, fit a cable tie around the pen, below the solenoid. Lastly, wire up the solenoid, screw on the 'toes' at the front and back of the base plate, attach the shell — and HEBOT is ready to roll! One word of caution however, try not to pick it up by the base of the shell, as this will release the ball-bearing from its mounting. It remains trapped by the microswitches and the plate bolts, but the collision detection will not operate unless the bearing is in the correct position.

The Interface

The interface board has been specially designed for use with HEBOT and should present no difficulties in assembly — just make sure that the ribbon cable wires go to the correct places! As mentioned earlier, any I/O system capable of controlling eight bits of output data and accepting at least four bits of input can be used to control the robot; the ZX I/O board which appeared in the September issue of Hobby Electronics, for example, would be adequate. In any case make sure you know the robot's address. On the HEBOT I/O board, this is

Table 4.

```
10    REM "RECORD MOVES"                     ; comments
20    LET A = 65535
30    DIM Z(5)                               ; set up move array
40    DIM T(5)                               ; set up move time array
50    FOR D = 1 TO 5                         ; move counter
60    FOR N = 1 TO 100                       ; move timer
70    IF INKEY$ <> " " THEN GOTO 100         ; jump if move
80    PAUSE 10                               ; move time increment
90    NEXT N
100   IF INKEY$ = "S" THEN LET M = 0         ; stop
110   IF INKEY$ = "F" THEN LET M = 5         ; forward
120   IF INKEY$ = "B" THEN LET M = 10        ; 'back' (reverse)
130   IF INKEY$ = "L" THEN LET M = 9         ; left
140   IF INKEY$ = "R" THEN LET M = 6         ; right
150   POKE A, M                              ; move
160   LET Z(D) = M                           ; store move
170   LET T(D) = N                           ; store move time
180   NEXT D                                 ; next move
190   PAUSE 20
200   REM "PLAYBACK"
210   FOR D = 1 TO 5                         ; set up move counter
220   LET M = Z(D)                           ; recall first move
230   LET W = T(D) + 1                       ; first move time period
240   POKE A, M                              ; playback the move
250   FOR N = 1 TO 100                       ; set up move timer
260   PAUSE 10                               ; move time increment
270   IF N = W THEN GOTO 290                 ; at end of move period
280   NEXT N
290   NEXT D
300   POKE A, 0                              ; stop
310   STOP
```

Table 3.

```
10    REM "HEBOT'S MOVE"                     ; comments in this column
20    FAST
30    LET A = 65535                          ; Hebot's address
40    LET X = 0                              ; clear collision flag
50    LET M = 5                              ; movement command
60    POKE A, M                              ; move
70    GOSUB 260                              ; short pause to move
80    POKE A, M+16                           ; flash lights
90    GOSUB 260
100   LET K = PEEK A
110   IF K <> 0 THEN GOTO 150                ; collision!
120   IF X = 1 THEN GOTO 190                 ; previous collision
130   IF INKEY$ = "S" THEN GOTO 290          ; emergency stop
140   GOTO 60                                ; continue forward
150   IF K > 6 THEN GOTO 200                 ; rear-end collision
160   LET M = 10 + 64                        ; reverse and sound horn
170   LET X = 1                              ; set collision flag
180   GOTO 60                                ; move in reverse
190   LET X = 0                              ; clear collision flag
200   LET M = 8                              ; turn left
210   POKE A, M
220   GOSUB 240                              ; long pause
230   GOTO 50                                ; continue forward
240   PAUSE 50                               ; variable length pauses
250   PAUSE 30
260   PAUSE 10
270   POKE 16437, 255
280   RETURN
290   POKE A, 0                              ; emergency stop
300   STOP
```

set up on the ten-bank DIL switch, as explained. The most convenient address is right at the top of memory — a ReaD instruction to HEBOT will clash with the ZX81's unbuffered memory, however no problems have been experienced using this high address, as the interface board overrides the unbuffered memory.

If the ZX81 is to be used with a RAM pack, then an address between 8192 and 16383 should be used and the 'echo' of the computers ROM should be disabled by pulling $\overline{\text{ROMCS}}$ high with a diode from IC4 pin 8, to that line. This connector will be included on PCBs supplied in Powertran kits. To fit the RAM Pack as well as the Interface board, a 3-way adaptor is required and is available from Powertran. With this, the RAM Pack lies on top of the computer.

Testing

Plug the I/O board into your ZX81 computer and power on. HEBOT will (probably) immediately begin to move in the direction of the nearest exit or table'top, due to some random bit-pattern on the control lines! Quickly send the following command: POKE 65535,0. This will stop the robot in its tracks.

Now turn each motor drive preset fully clockwise; then, using a voltmeter, turn each preset back till the voltage has dropped by 1V; this will allow the supply voltage to fall by up to this amount without affecting the robot's speed.

Next, POKE 65535,5 and HEBOT will move forward; RV2 and RV3 must be adjusted to ensure that it travels in a straight line (make sure the rubber wheels are on straight before you do this). Now POKE 65535,10 to move it in reverse, and adjust RV1,4 to match the reverse speed to the forward speed.

To test the remaining functions, POKE 65535,16; HEBOT should stop with the LEDs glowing balefully red; POKE 65535,32 should drop the pen, and POKE 65535,0 should retract it again. POKE 65535,64 will sound the horn in the lower frequency, while sending 192 should increase the pitch.

To test the collision detectors, write a short routine to repeatedly PEEK 65535 and look for 1, 2, 4, and 8 from the respective sensors.

Two Programs

To really give your new pet a workout, try the two simple programs listed in **Tables 3 and 4** — but please remember that these are presented only as starting points for further development. Come along to Breadboard '82 and show us what your HEBOT can do — or send in your program on cassette. The best routines will be presented in future issues as an inspiration to all HEBOT trainers. Watch out, Barbara Woodhouse!

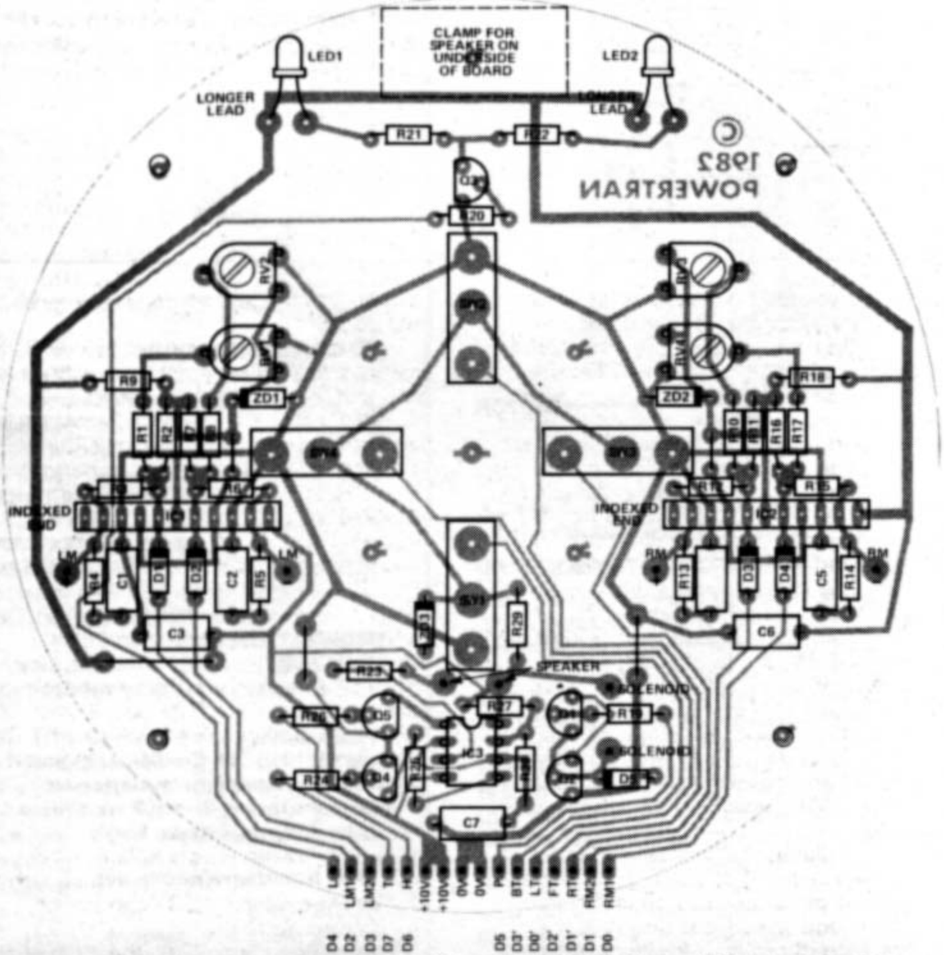


Figure 3. The component overlay board for the HEBOT Control Board; the LEDs indicate the "forward" direction.

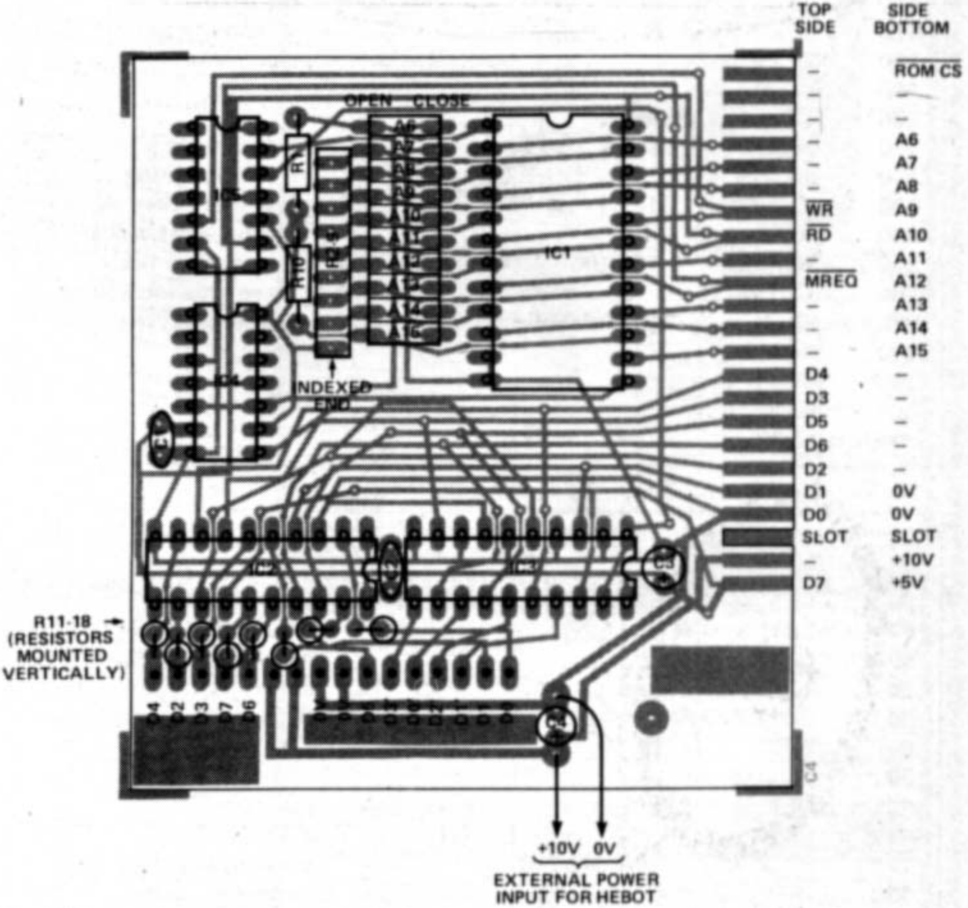


Figure 7. The component layout of the Interface Board. Note that both sides of the double-sided PCB are shown.