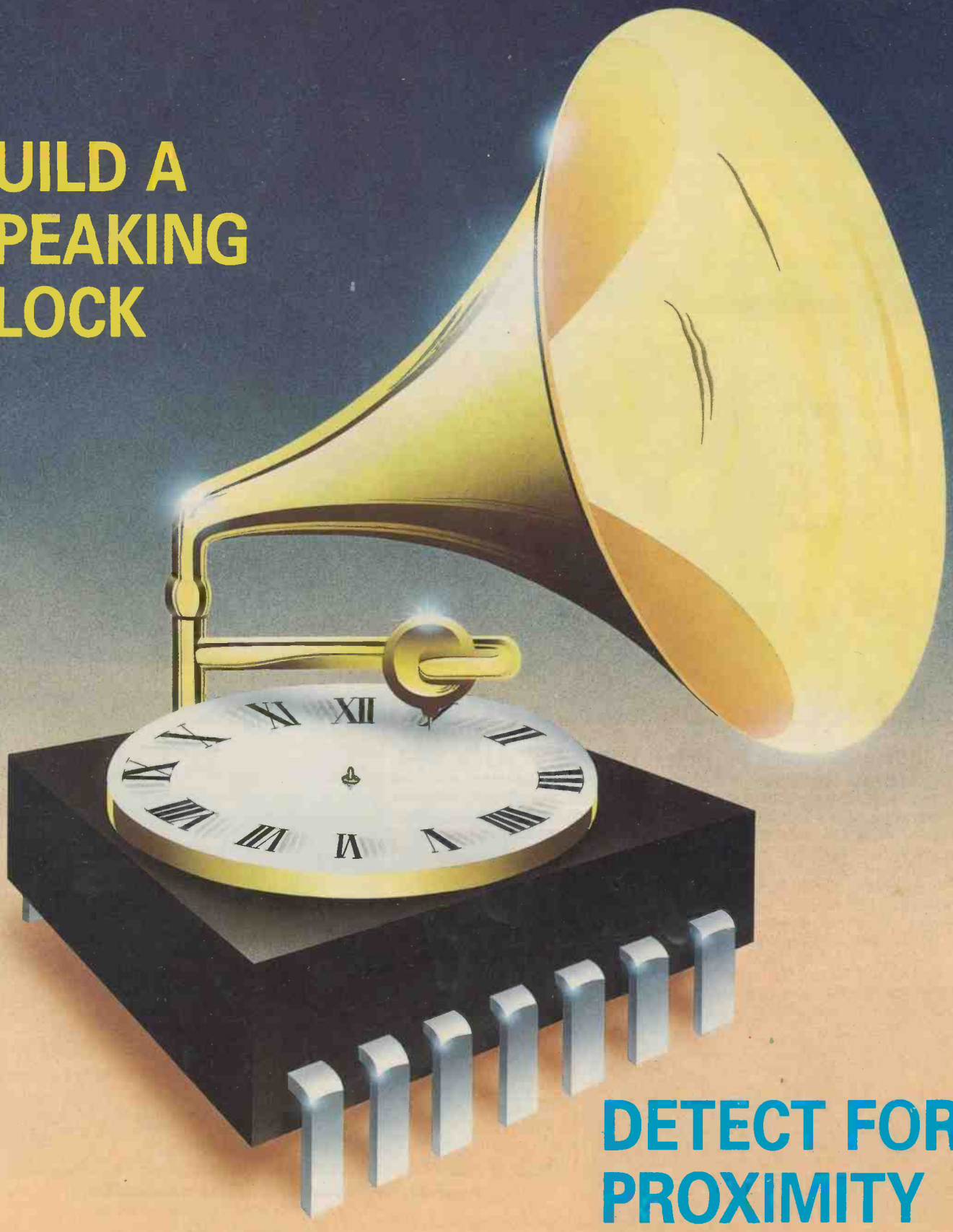
PRACTICAL
AUGUST 1988 · £1.25

ELECTRONICS

SCIENCE & TECHNOLOGY

BUILD A SPEAKING CLOCK



DETECT FOR PROXIMITY

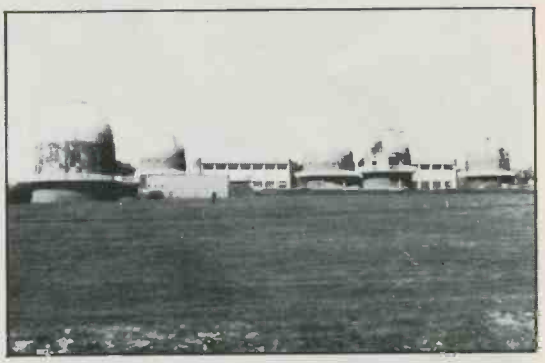
THE SCIENCE MAGAZINE FOR SERIOUS ELECTRONICS AND COMPUTER ENTHUSIASTS

PRACTICAL ELECTRONICS

PE VOL 24 NO 8 AUGUST 1988

CONSTRUCTIONAL PROJECTS

SPECIAL FEATURES



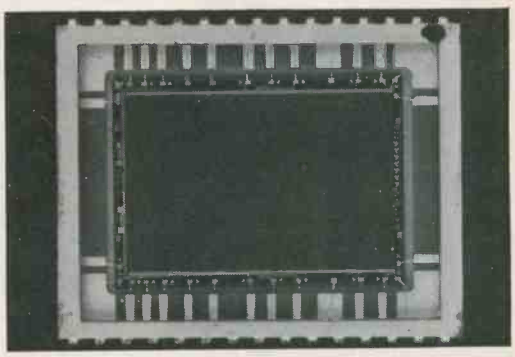
REGULAR FEATURES

PRODUCT FEATURES

NEXT MONTH . . .

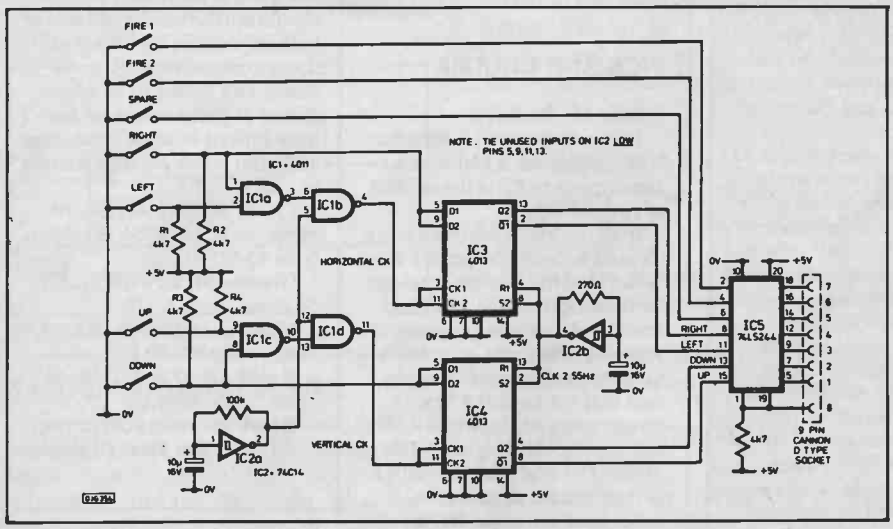
WE'VE BEEN PRESSURED TO TAKE A STAR-SPANGLED LOOK AT – A COMPUTERISED BAROMETER • ASTRONOMICAL ELECTRONICS • MUXXING-OUT THE BEEB • MICROPROCESSOR DEVELOPMENT • CD ROM • AND IT'S BACK TO THE CHALK-TALK FOR THE START OF ANOTHER CLASSIC GCSE TUTORIAL SERIES •

MASTER YOUR SCHOLARLY NEWSAGENT AND ORDER YOUR FIRST-CLASS COPY OF OUR SEPTEMBER ISSUE ON SALE FROM FRIDAY AUGUST 5TH



"INGENUITY UNLIMITED"

AN AMSTRAD MOUSE SIMULATOR



THE joystick port on the Amstrad CPC computers is scanned as part of the keyboard system at a default speed of 50 times every second. Anyone who has tried to use the joystick for accurate control of the screen cursor position will have found it difficult. This is due to the rate at which the computer scans the joy-stick port, requiring only a small movement on the joystick to produce a large movement of the screen position. The following circuit was designed to get around this problem by converting the joystick switch positions into a stream of pulses on the joystick lines, similar to those generated by moving a mouse.

The circuit consists of only five dil ic's these go to make up the input gating from the direction switched, two astable clocks, direction latches and a tri-state buffer to drive the joystick port on the computer. The Schmitt invertor gate IC2a and it's associated components R5 and C1 form the first astable clock which is fed to one of the inputs on gates IC1b and IC1d whose outputs generate the vertical and horizontal clocks and which are used to clock the position latches IC3 and IC4. The outputs of IC1b and IC1d are normally held high by the low outputs from IC1a and IC1c, only when one of the four position switches are pressed does the relevant clock get gated through to the latch. This signal now clocks the selected position through to the Q2 and NOT-Q1 outputs. This will appear at the joystick port each time the port is scanned. However the second astable clock made from IC2b and R6,C2 clocks the set and reset inputs on the position latches every 18ms, changing the state of the latch outputs to a high state. As the computer only scans the port for a low condition to signify joystick movement it no longer sees the port as active and will cease to increment the screen position.

If the position switches are kept depressed then the position clock will remain active and will re-clock the position into the latch, appearing once again on the joystick port before being cleared by the second astable clock. The overall effect of this action is to only allow the position switches to be available for reading by the port for a short period of time before being cleared, hence only updating the screen position by small amounts each time. The circuit uses cmos ic's to keep power consumption to a minimum the four pullup resistors R1 to R4 are to prevent the inputs of IC1 from floating when no position buttons are pressed.

A small plastic box used to house the circuit. Seven normally open push-buttons were mounted on the lid, three in a row at the top providing fire1 fire2 and spare fire button. The four position switches were arranged below, similar to the cursor key layout. Wires were taken down from these switches to the main circuit board mounted in the base of the box. Two leads were brought out of the rear of the box, one having a 9 pin socket to connect into the joystick port, the other carried the power to the circuit board. The +5v required may be picked up from the expansion connector pin 27. The usual static precautions should be observed when handling the ic's and the use of dil sockets recommended. When the unit is completed and plugged into the computer, enter the following program:

```
10 cls:mode1
20 locate 20,12:print joy(0)
30 if joy(0)<0 then 20 else 40
```

The following numbers should be displayed on depressing a key on the unit: TOP LEFT=32TOP CENTER=64TOP RIGHT=16 UP=1 DOWN=2 LEFT=4 RIGHT=8. If the right numbers appear, clear oput the program and enter the next program. This allows simple line drawings to be created on the screen. The top right key cancels the line, top centre allows cursor movement without drawing, pressing it again returns you back to drawing mode, top left makes the line permanent.

R. Hewertson, Leigh

Main program for the Amstrad mouse simulator

```
10 CLS:INPUT"sensitivity=";s
20 MODE 1:INK 3,24
30 x=320:y=200:h=320:v=200:lin=1:GOSUB 140
40 WHILE-1
50 IF INKEY(72)=0 THEN GOSUB 190:v=v+s:GOSUB 190
60 IF INKEY(73)=0 THEN GOSUB 190:v=v-s:GOSUB 190
70 IF INKEY(75)=0 THEN GOSUB 190:h=h+s:GOSUB 190
80 IF INKEY(74)=0 THEN GOSUB 190:h=h-s:GOSUB 190
90 IF INKEY(77)=0 AND LIN=1 THEN GOSUB 240
100 IF INKEY(76)=0 AND LIN=1 THEN GOSUB 270
110 IF INKEY(78)=0 AND LIN=0 THEN GOSUB 270
120 IF INKEY(79)=0 THEN GOSUB 310
130 WEND
140 PRINT CHR$(23);CHR$(1);
150 LOCATE 1,1:PRINT "x=";h;" y=";v;"sensitivity=";s
160 MOVE h,v
170 DRAWR 0,5,2:DRAWR 5,0:DRAWR 0,-5:DRAWR -5,0
180 RETURN
190 IF LIN=0 THEN GOTO 220
200 PRINT CHR$(23);CHR$(1);
210 MOVE x,y:DRAW h,v
220 GOSUB 140
230 RETURN
240 PRINT CHR$(23);CHR$(0);
250 MOVE x,y:DRAW h,v,1:x=h:y=v
260 RETURN
270 GOSUB 190
280 IF LIN=1 THEN LIN=0 ELSE LIN=1:x=h:y=v
290 GOSUB 190
300 RETURN
310 PRINT CHR$(23);CHR$(0);
320 MOVE h,v:DRAW x,y,0:x=h:y=v
330 RETURN
```