ROM. The purpose of this routine is to scan the keyboard to see if a key has been pressed and to increment a 3 byte counter in memory. By connecting $\overline{INT}$ to +5 volts, you stop the ULA interrupt signal reaching the CPU. The CPU doesn't ever scan the keyboard or increment its counter unless it receives an interrupt. Therefore none of the keys on the keyboard operate and the counter stops.

Now enter this one line program.

```
10  FOR t = 1 TO 30: BEEP 0.01,t: NEXT t: GO TO 10
```

If you run it you will hear a BEEP of rising pitch from the buzzer. Try operating the INT switch. It has no effect and the tone continues. INT doesn't affect output from the CPU to the ULA. It only affects inputs from the ULA to the CPU. Switch INT off again. Now try operating the IORQGE switch. The BEEP suddenly stops. Yet again the keyboard has become totally unuseable, but for a totally different reason. Can you see why?

All I/O which is done by the ULA relies upon the ULA being able to detect whenever the CPU wants to send it information or get information from it. The ULA uses the address lines, $\overline{RD}$, $\overline{WR}$ and $\overline{IORQGE}$ to test if the CPU is trying to communicate. By connecting IORQGE to +5 volts, you are preventing the Z80A $\overline{IORQ}$ signal from reaching the ULA. No communication between the two devices can therefore occur. The 3 byte ON time clock does continue to run because the ULA is still generating interrupts for the CPU 50 times per second.

ADDING 128 I/O PORTS

In the Spectrum a very simple form of decoding I/O port addresses is used. A0 is used to select the ULA, A1 the ZX printer and A3, A4, A5 the microdrives and RS232 interface. No more than one of these address lines should ever be at logic 0 during an I/O operation. If they are then several devices will try to use the data bus simultaneously. Homebrew circuits therefore only have address lines A5, A6 and A7 available to select them. These address lines would all be used up addressing one PIO chip (as in chapter 15). Some method of using A0 — A4 would therefore be useful. This is what this circuit could do, but there are some problems.

Consider the circuit shown in fig 10g. The 74LS01 chip has 'open collector' outputs which can sink current but cannot source it. When A7 is high $\overline{IORQ}$ is fed through to $\overline{IORQGE}$ as normal. However, when A7 is low, $\overline{IORQGE}$ remains high irrespective of $\overline{IORQ}$.

In the case of the ULA it will only be selected as an I/O device when A7 = 1 and A0 = 0. It will not now be selected when A7 = 0 as it would have been without this extra circuit. If you want to use the ZX printer and other peripherals which use the $\overline{IORQ}$ instead of the $\overline{IORQGE}$ signal then the circuit in fig 10g must be modified. The input $\overline{IORQ}$ must come from the edge connector. The $\overline{IORQ}$ output should then be connected to the $\overline{IORQ}$ input on all extra device connectors in place of the original Z80A $\overline{IORQ}$ signal.

To show that the circuit works with the ULA try this short program.

```
100  LET a = IN 63486
110  LET b = IN (63486—128)
120  CLS: PRINT a,b
130  PAUSE 5
140  GO TO 100
```

This program reads from the keyboard keys 1 — 5. If you press any of the keys 1 to 5 you will observe that the 'a' value changes from 255 but that 'b' doesn't. If you try this same program without the extra circuit connected then both a and b will change when you press any key from 1 — 5. 'b' is not read in from the ULA with the circuit connected because A7 is at 0, thereby disabling I/O with the ULA.

NOTE: This circuit cannot be used with Issue 2 Spectrums with a transistor soldered across the Z80A chip.