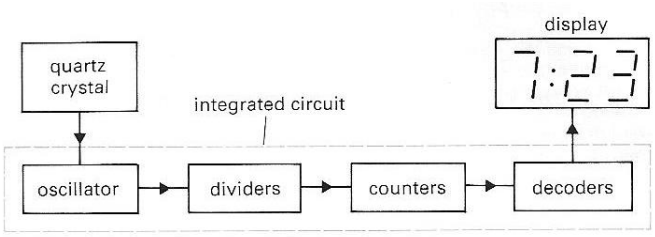
Digital watch

The electronics of a digital watch may be contained in a single integrated circuit. However, we can think of the chip as a number of separate units, each performing a different function. This can be shown in a block diagram.



The oscillator generates pulses at a fixed frequency 32768 Hz. This frequency is determined by the natural oscillation of the quartz crystal. The divider circuits perform binary division on the pulses to reduce their frequency to one pulse per second. A binary count of these pulses is made by the counter circuits and the decoders convert the binary output into signals which light up the appropriate segments of the digital display.

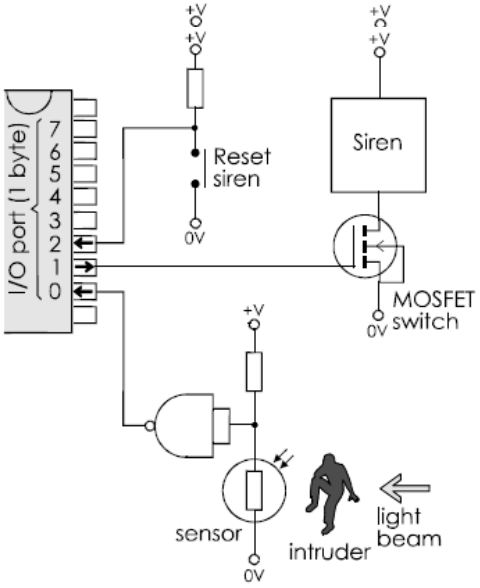
Dividers form one of the main components of this system. They consist of bistables.

Microcontrollers

A microcontroller is a single integrated circuit on which are combined most of the circuit blocks that we find as separate units within a computer.

- Microcontrollers are "**embedded**" inside some other device (often a consumer product) so that they can control the features or actions of the product.
- Microcontrollers are **dedicated** to one task and run one specific program. The program is stored in ROM (read-only memory) and generally does not change.
- Microcontrollers are often **low-power devices**. A battery-operated microcontroller might consume 50 milliwatts.
- A microcontroller is often **small and low cost**. The components are chosen to minimize size and to be as inexpensive as possible.

Microcontrollers are hidden inside a surprising number of products these days. All modern automobiles contain at least one microcontroller, and can have as many as six or seven: the engine is controlled by a microcontroller, as are the anti-lock brakes, the cruise control and so on. Any device that has a remote control almost certainly contains a Microcontroller. Basically, any product or device that interacts with its user has a microcontroller buried inside.



Choose the correct word in the sensor description

In this very simple programming example, we need only three bits. Two are programmed to be inputs (from the sensor and from the reset button). One is programmed as an output (to switch on the siren). The sensor circuit includes a NAND gate with its inputs connected together. It acts as a **NOT/NOR** gate.

The output goes **high/low** when the beam is broken by the intruder, making bit 0 low. The output bit 1 is normally low. When it is made high it switches **on/off** the MOSFET and the siren sounds. Bit 2 is an input that reads the state of the reset button. Pressing the button makes bit 2 go **high/low**.

In fact, it is barely worthwhile to use a controller on such a simple system as our example. The same action can be obtained with a few logic gates.