2. GENERAL OVERVIEW OF THE SPECTRUM

This chapter is essentially in two parts. The first part aims to introduce the basic concept of binary numbers. The second part explains each of the main sections of the Spectrum in a general way.

SECTION A

Computers are essentially two state devices. They rely upon logic to operate. This logic can be in one of two states only. For convenience, these two states are usually represented by a 0 and a 1. At its simplest level, the computer manipulates 1's and 0's to produce the answers. Consider for example the simple operation of addition. This could be represented by a black box with two inputs, A, B and two outputs C, D. A and B could be added together to produce their sum represented by C and D. The addition would be carried out by several simple transistors inside the black box. The addition function would be defined in binary as follows:

A + B = D (sum)	& C(carry)
0 + 0 = 0	0
0 + 1 = 1	0
1 + 0 = 1	0
1 + 1 = 0	1

Note that 1 + 1 cannot equal 2 because in the binary number system, only 0 and 1 can be used. The carry bit here is similar to that in the decimal number system. If we added 8 + 9, this would give 7 carry 1 in the decimal number system. The difference is that a carry in binary represents 2, whereas it represents 10 in decimal.

SECTION B

In the following descriptions of the various sections within the Spectrum, you will find it helpful to refer to the overall block diagram in fig 1.

The central processing unit (CPU) is, as its name implies, central to the operation of the Spectrum. It is connected to other parts of the computer by data, control and address buses (more about these later). The CPU in the Spectrum is a Z80A and lives inside the large chip IC2 at the centre of your Spectrum board. This processor is an 8 bit device which means that there are 8 separate connections in its data bus. The CPU can send information to other devices in the Spectrum along this data bus and the other devices can send data back to the CPU via the same bus. Because there are eight connections, each one of which can be either a logic 0 or a logic 1, any number between 0 (all 0's) and 255 (all 1's) can be sent via the data bus ($255 = 2^8 - 1$).

You may wonder how the CPU can understand very large decimal numbers or words typed at the keyboard when it is using BASIC. After all, if you type HELLO at the Spectrum keyboard it might be difficult to see how this can be represented by a number between 0 and 255. In fact, the answer is not very difficult. The CPU only deals with a small part of the operation at a time. To understand HELLO, the CPU first deals with the H (seen by the CPU as 72 decimal), then it deals with E

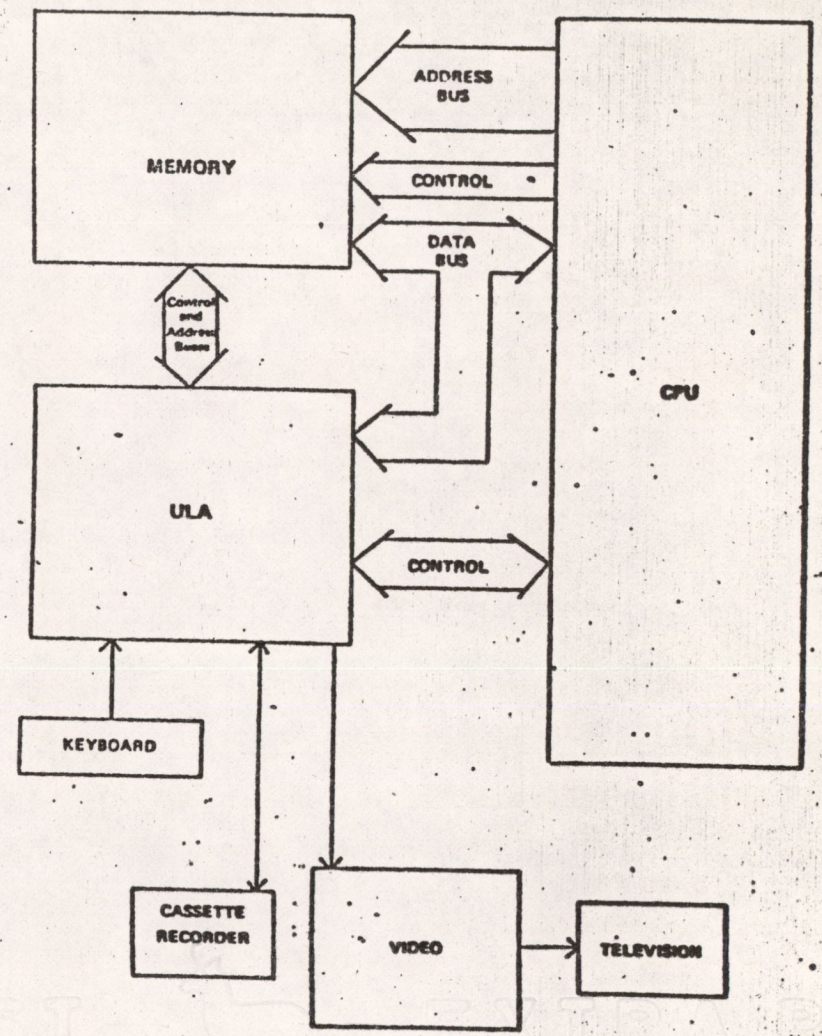

FIG 1 – SPECTRUM BLOCK DIAGRAM