HARVEST REPORT #2

Subject: Use of Data Bits to Generate Memory Addresses

By: S. W. Dunwell

Date: September 6, 1956

September 6, 1956
(Revision of July 10, 1956)

HARVEST REPORT #2

Subject: Use of Data Bits to Generate Memory Addresses

An extremely important feature of the computer being planned will be the ability to use tables held in memory as substitutes for specialized mechanism. As an example, it will be possible to use memory as counters to obtain distributions for statistical purposes. Similarly, multiple reference to tables in memory will allow second and third order indexing operations. For these, and many other applications, one or more characters of data must be associated with data of other kinds and processed as an argument to produce the memory address of the word desired from the table.

When the primary concern is to use the least amount of memory, the usual practice will be to perform a series of arithmetic operations on the data to obtain the memory address. However, when it is desired to operate at the highest possible speed, the time required to perform address arithmetic becomes an obstacle. To escape this, it becomes necessary to develop the memory address by direct assembly of data bits, avoiding arithmetic processes upon them.

Parts of the Address

When assembled in this manner, the address may have the following parts, not all of which are necessarily present in every instance.

1. Prefix

The prefix identifies the beginning point of the table in memory.

2. True Argument

The data which determines the word to be selected from the table is the true argument. In a typical case of counting under control of 3-digit numbers, it might

cnnsist of three 4-bit characters, or a total of 12 bits.

3. Suffix

If it is desired to store the table in its entirety in a single memory unit, rather than to distribute the table across the several memories in the usual manner, it is necessary to append as a suffix one or more bits to designate the memory unit to be used. This is because the low order digits of the address select among the memory units to obtain an automatic statistical distribution of the memory references for ordinary operations.

4. Split

If several words of the table are stored in a single word of memory, it is necessary to specify the portion of the memory word which is to be read out. When reading out in this way, the data appears on the low order positions of the bus system. This kind of operation may be quite common, since the data stored in the table will often be a single character or a 7 or 8 bit count.

Size of Tables

Several consequences of the use of data bits as addresses can be immediately perceived. One of these is that the table must occupy a rectangular space in memory, with the sides of the rectangle being powers of two. Each side of the table, whether the table has one, two or more dimensions must have a number of rows, columns, etc. which is a power of 2. A two dimensional table can be 16 x 16, 32 x 32, 64 x 64. It cannot be 29 x 29. If only 29 of the 32 binary possibilities are needed, the remainder of the possibilities in each dimension will be vacant and unused unless some special use can be found for them. There is no way of avoiding this without the use of arithmetic on the arguments of the table.

Starting Point of Tables

The starting point of a table, as identified by the prefix, must be zero or a power of two times the size of the table. Otherwise,

the prefix would differ with different true argument values in the table.

Partial Word Tables

Whenever several function values are stored in a single word of memory, the number of values in the word must be a power of two. If we allow for a 64-bit memory word, it is possible to store two values of 32 bits, four of 16 bits, eight of 8 bits, sixteen of 4 bits, thirty-two of 2 bits, or sixty-four of 1 bit.

Successive Table Lookup

There are a number of cases in which several table references must be made, with the value read from each succeeding table providing some of the data bits from which the address for the next following reference will be made. When this occurs, one wishes to insure that each reference is made to a different memory to guarantee against the occasional conflicting references and corresponding speed reduction which result from the standard method of addressing. The address suffix already discussed allows one to select memories on a basis which avoids conflicting references.

Special Characters

It will sometimes happen that one or more specific bit combinations will convey special meanings, and must be recognized by the logical unit and given special treatment. These characters will typically be record marks, garbles, spaces, and word separators. As an example of the special handling required, the presence of a garble in either of two streams generating a third stream might interrupt normal processing and substitute a special process suitable for garbles.

Alternating Areas in Memory

When counting in memory, the process of evaluating the count may be of such length as to require that it take place simultaneously with the assembly of the next count. In this event, alternating areas in memory may be required with, at any time, one being used for counting and the other for evaluation. To accomplish this, the prefix portion of the address will be varied between each count and the one which follows it.

Assembling Addresses

Since the number of parts and the size of parts from which the address is assembled is highly variable, the mechanism which specifies them must be similarly variable. Plugboards are ideally suited for this purpose, and it may be that an elementary plugboard with a small number of short coaxial lines will serve best. However the operation is performed, it will be classed as a logical operation, to be performed by the logical mechanism.

Example

An example will serve to illustrate the principles involved in assembling an address. We will assume that from two continuous streams of random digits it is desired to perform an index operation on every pair of corresponding digits. Such 2-digit numbers will be obtained at a rate of one each 0.2 microseconds from memory through a pair of continuous stream registers. The index operation requires that the 2-digit numbers identify a particular segment of memory to be operated on.

1. The table must be 16 x 16, since 16 is the lowest power of 2 which is greater than 10.

2. The input characters must be expressed as 4-bit characters, and as 8-bit binary-coded-decimal numbers.

3. There will be no suffix, since it will be desired to distribute the 100 memory references statistically.

4. Each memory word will contain four 16-bit segments containing information to correspond to four different 2-digit numbers. Two low order bits of one of the two digits will identify the split of the memory word.

5. The remaining two bits of the first digit and the four bits of the second digit will be assembled as the low order six bits of the memory address.

6. The remaining 14 bits of the 20 bit address will define one of the permissible 2^{14} starting points for the table in the memory.

Revision: The above is a revision of the original Report #2 of July 10, 1956 prepared as a result of the joint conference of July 13, 1956.

S. W. Dunwell

SWD/jh

S. W. Dunwell