Griffith

COMPANY CONFIDENTIAL

December 20, 1956

Exchange Memo No. 12

Information Packing by Use of a Byte Converter

By: H. K. Wild

In the Stretch computer system a large number of types of peripheral units is expected to be used. Incorporated within the system will be input and output units using bytes, ie: groups of bits, which are not exact submultiples of the number of bits used in a word in the main memory of the machine. In the Stretch Machine the word size in the main memory is 64 bits. Included in the complement of peripheral equipment for this machine there will be units in which the storage of information might be serial by 5, 6, 7, or 8 bit bytes. These bytes may or may not be individual characters. Regardless of this the organization of the machine is such that it is desirable that the information in the main memory of the machine be fully packed. By this it is meant that, for example, a stream of N bits stored in six bit groups on magnetic tape when transmitted to the working memory of the machine should be stored as N bits of information in 64 bit groups. There would be no gaps in the stored stream of bits.

The Exchange system provides a method whereby bytes which are submultiples, ie; 8 bits, of the 64 bit word are sequentially read into the Exchange from the peripheral storage media, assembled into 64 bit words which are then transmitted to the main memory of the machine. On output a full 64 bit word is broken up into 8 bit bytes which are sent sequentially to the output units.

A problem then exists for those peripheral units storing information in a size byte other than 8 bits. It is then necessary to convert the byte size from that used in the media to that desired for the Exchange input or output.

For example; if the storage media utilizes 6 bits as in the 727 tape then if N bits of information in 6 bit groups are presented to a black box the output of the black box should consist of the same sequence of N bits of information in 8 bit groups to be entered into the Exchange.

It has been determined that this can be done very effectively by the use of a core matrix which will now be described. A diagram of a 6 bit to 8 bit Byte Converter is shown in Figure 1. This diagram shows a core matrix of 24 cores into which information is read in 6 bit groups from a 727 Tape. The output of the 727 Read register is fed into the write drivers which causes a half write driver current to be present in each of the vertical write lines in the matrix corresponding to a 1 bit stored in the corresponding position in the Read register. Coincidentally a half write current is fed through a select diagonal so that the 6 bits of information are then stored along the diagonal which has been selected. The first 6 bit byte of each sequential group of 4 bytes is written into the diagonal selected by the write drive designated "1". These bits are shown as 1_1, 2_1, etc. The second 6 bit byte is written into the "2" diagonals selected by write current driver "2". These bits are designated as 1_2, 2_2, etc. Read Out driver number 1 will read out the first byte of 6 bits, 1_1, 2_1, etc., and also 1_2, and 2_2 of the 2nd 6 bit input byte to form the first 8 bit output byte. The second 8 bit output byte will be formed of 3_2, 4_2, 5_2, 6_2 and 1_3, 2_3, 3_3, and 4_3. The third 8 bit output byte will consist of 5_3, 6_3, and 1_4, 2_4, 3_4, 5_4, and 6_4. The process then repeats.

The timing of the output cycles relative to the input cycles is also shown in Figure 1. There are four input cycles taken for each three output cycles. Since coincidence selection is not necessary on read out full read out currents are used. The process repeats itself for every 4 characters

entered. The core matrix configuration and the orientation of write in and read out windings keeps the bits in the input sequence and remembers the remainder to provide a fully packed output byte.

In general, a byte converter will utilize a number of cores which is the least common multiple of the number of bits in the entry byte and the desired number of bits in the exit byte. For example, if it is desired to convert N bits of information in 7 bit groups into N bits of information in 8 bit groups the core matrix would contain 56 cores. There would be 8 input cycles for 7 output cycles. Seven half write drivers for the vertical write lines and eight half write drivers for the diagonals would be required. There would also be 7 full read drivers required.

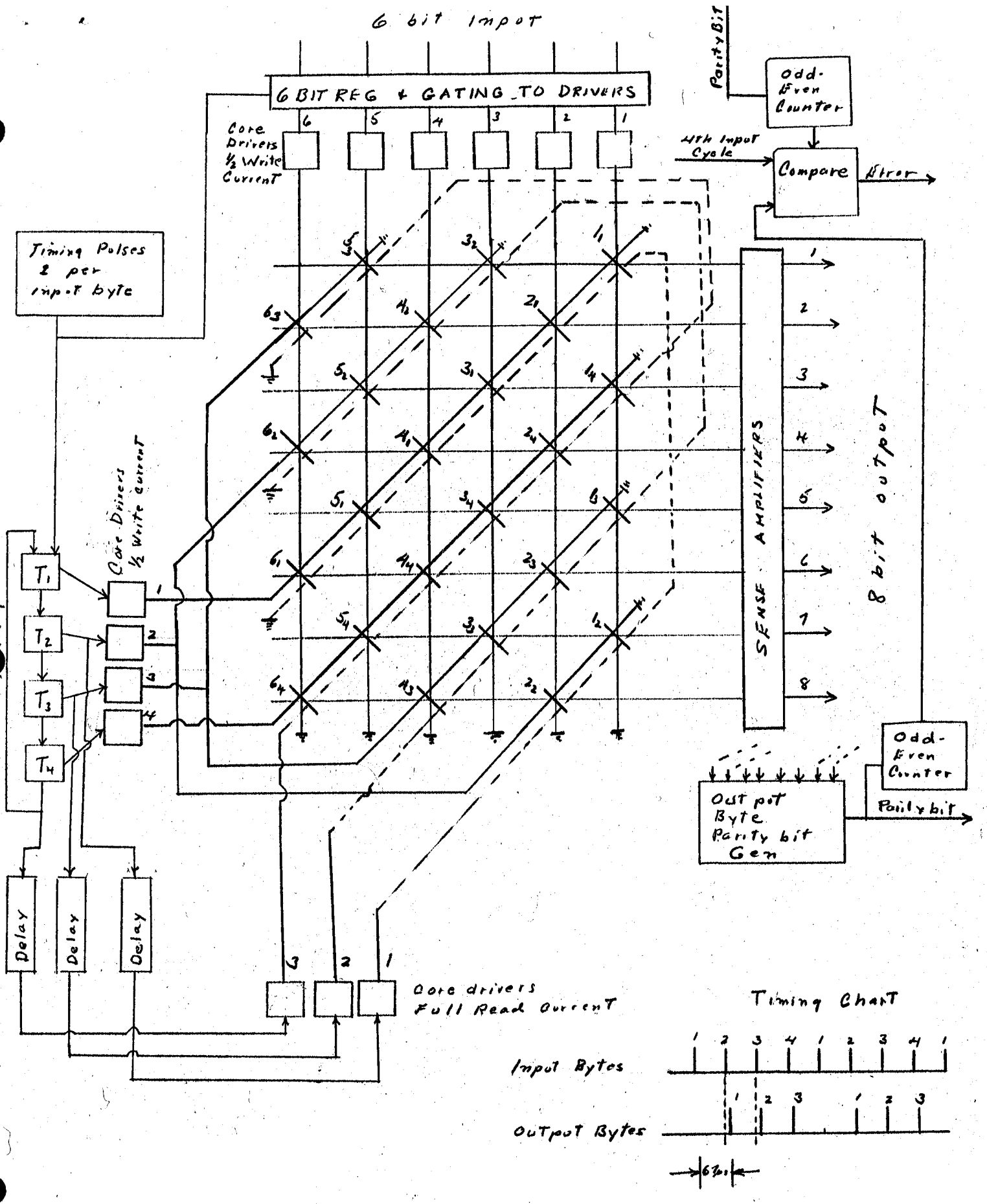
Another example would be that of converting N bits of information in 5 bit groups to N bits in 6 bit groups. This would require a matrix of 30 cores and the number of half write drivers for the diagonals in this case would be 6 and the number of read drivers would be 5. A diagram of the core matrix is shown in Figure 2. A particular matrix configuration is necessary to provide the proper continuity in the sequence of bits while storing the overflow, if any, for inclusion in the following output byte.

It is necessary when writing to an output unit to convert from the standard output byte to the bit group size which is utilized by the output media. In the case where the output byte size is 8 bits and the byte utilized by the media is 6 bits an 8 to 6 converter would be necessary. A diagram of the core matrix is shown in Figure 3. In this case it can be seen that for every 3 input cycles there are 4 output cycles.

If a parity bit is utilized by the media it is possible to carry the checking procedure around the converter to provide a check upon the operation of

the converter. For example; in the case of the 6 to 8 converter the parity bit would not be routed through the converter but instead would be summed in an odd-even counter over 4 input cycles. A parity bit would then be generated by each 8 bit group from the output of the converter and transmitted along with this information. In addition this parity bit would be summed in an odd-even counter over 3 output cycles and compared with the odd-even count of the parity bits from the input. Thus parity in this case can be considered as being taken over 24 bits. The same procedure can be followed for any N to M byte converter.

Essentially, this device provides a means of utilizing to the maximum efficiency parallel information channels for transmission or storage when there is a disparity in the size of bit groups stored or transmitted in "serial by byte" systems.

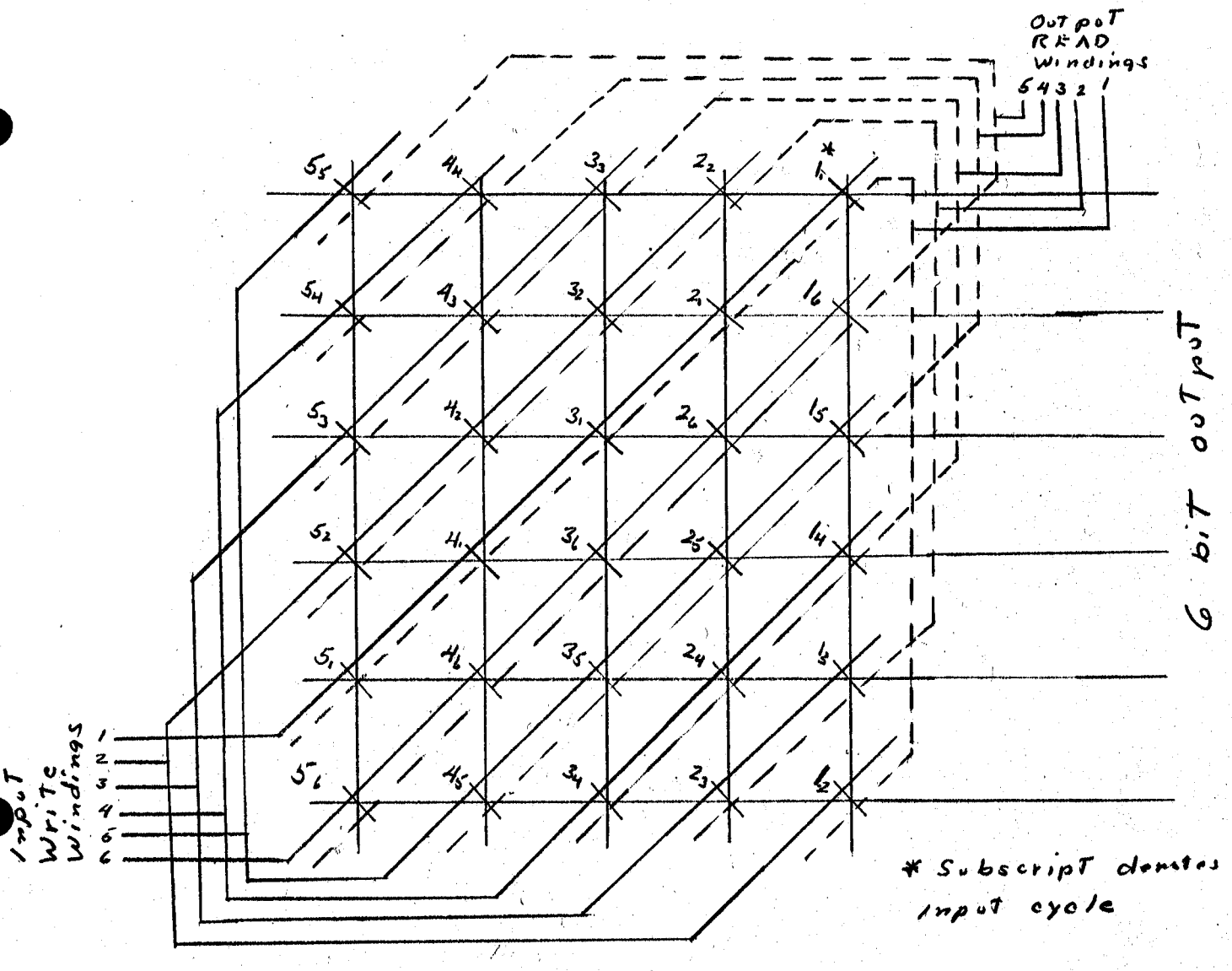


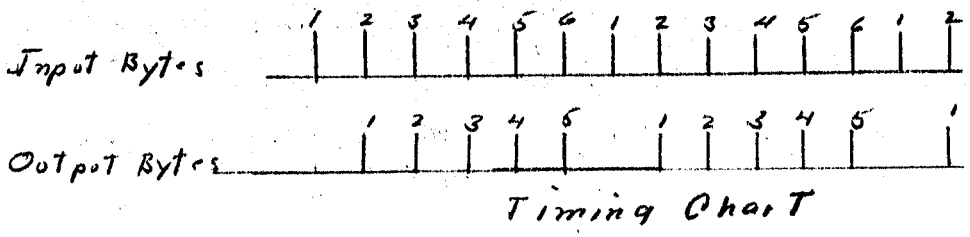
Byte Converter
6 bit to 8 bit

Fig 1

H.R. Wild
12-15-56

5 bit input

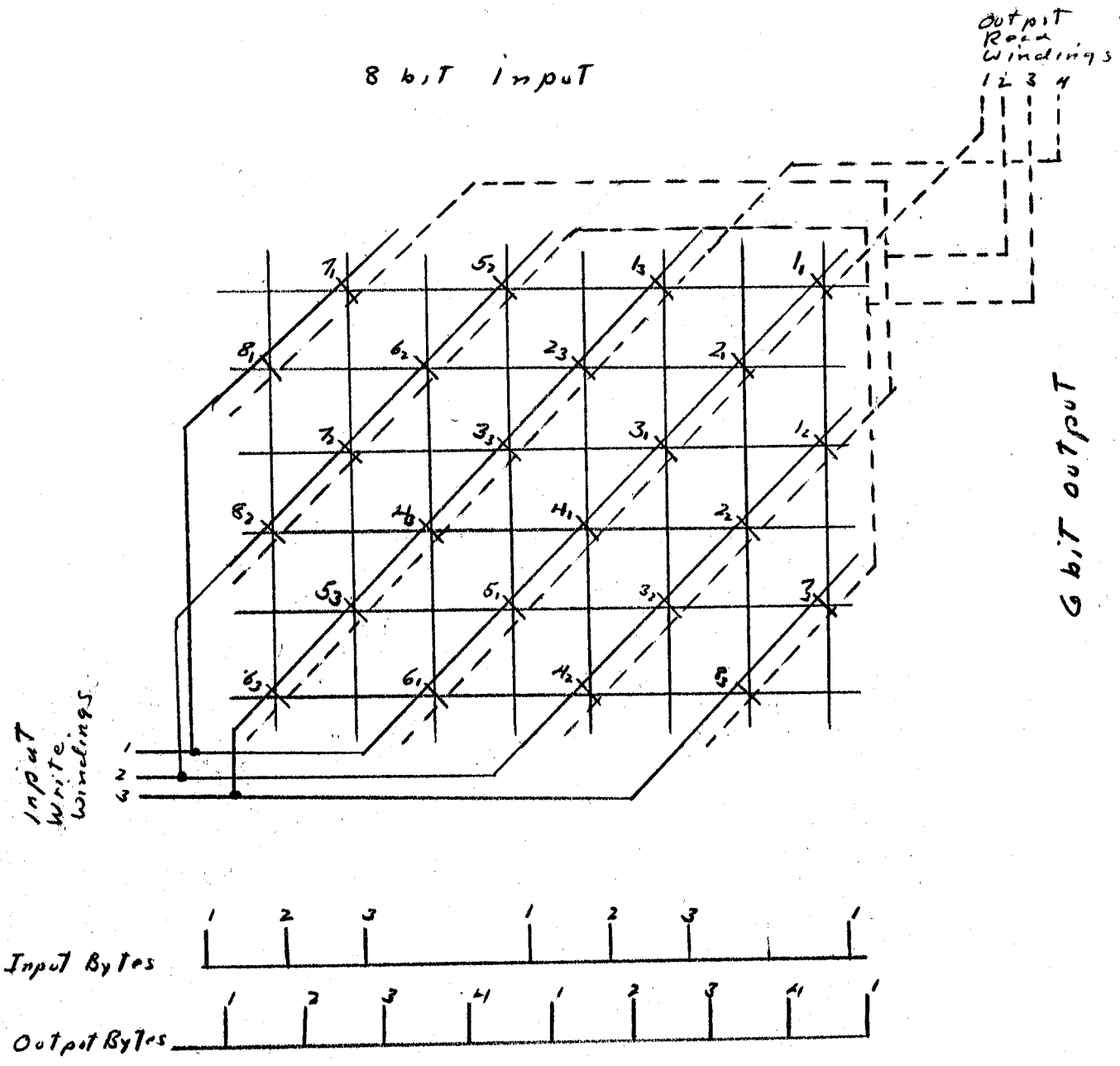




Core Matrix for Byte Converter
5 bit input
6 bit output.

Fig 2

H. K Wild
12-15-56



Core Matrix for Byte Converter
8 bit input
6 bit output.
Fig 3

H.K.Wild
12-15-56