8

COMPANY CONFIDENTIAL

PROJECT BETA

February 14, 1956

FILE MEMO #17

SUBJECT: Logical Connectives

BY: W. A. Hunt

The following simple explanation was obtained from Dr. B. Dunham and is reproduced here for information purposes.

If we have two binary variables, p and q, (binary in that they are either 1 or 0, respectively) the following "truth table" of logical connectives can be written.

p	q	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

If the columns are inspected, it can be seen that the columns are the usual binary representative of zero through 15.

If the logical connectives are thought of as an operator which operates on p, q, then the "truth table" can be defined as the results which will be obtained when p, q is operated on by the particular operator involved. To illustrate this statement, we can consider an operator O_8 which when it operates on p, q, generates the results shown under column 8 of the "truth table". This can be written symbolically as,

$$O_8(p, q) \equiv \text{Column \#8 or } C_8$$

If the columns are examined, the following is apparent:

1. C_9 through C_{16} are the ones complement of C_0 through C_8

2. C_4, C_{13} and C_6, C_{11} are functions of only q and p, respectively.

This leaves only five of the first eight. (The second half can be considered as redundant cases of the first eight.) Of the remaining five operators, O_2 is inclusive OR, O_7 is identity, and O_8 is AND. O_3 and O_5 are different from O_2, O_7, and O_8 in that if the values of p and q are interchanged in the table, the values of C_3 and C_5 are interchanged in the table also. This may be stated symbolically as:

$$O_3\ (p, q) \equiv O_5\ (q, p) \equiv C_3$$

or

$$O_5\ (p, q) \equiv O_3\ (q, p) \equiv C_5$$

O_3 and O_5 illustrate the two cases of the five where p and q do not commute, however, p, q are "interchange equivalent" in these two cases.

Dr. Dunham also points out in his paper* that if one examines a full adder where Ad is the addend, Au is the augend, Ci is the carry in, and S is the sum, and Co is the carry out, the following table results:

Ad	Au	Ci	S	Co
1	1	1	1	1
0	1	1	0	1
1	0	1	0	1
0	0	1	1	0
1	1	0	0	1
0	1	0	1	0
1	0	0	1	0
0	0	0	0	0

From the table, it can be seen that by controlling the carry in (Ci) three of the five operations (O_2, O_7, and O_8) can be executed.

The following information was not obtained from Dr. Dunham.

If one examines a full subtracter where Su is the subtrahend, M is the minuend, Bi is the borrow from the preceding operation (analogous to Ci in the full adder), R is the remainder, and Bo is borrow out, the following table results:

M	Su	Bi	R	Bo
1	1	1	1	1
0	1	1	0	1
1	0	1	0	0
0	0	1	1	1
1	1	0	0	0
0	1	0	1	1
1	0	0	1	0
0	0	0	0	0

From this table it is apparent that if a full subtracter exists, the remaining two operations, O_3 and $\overline{O_5}$ (not O_5) can be executed by controlling

* IBM Technical Report, Code, 001.011.580, B. Dunham, "The Multipurpose Bias Device," August 8, 1955

the borrow in (Bi).

Noting that if a full adder, a full subtracter, and an inverter are available, all of the binary connectives for two binary variables can be generated. It seems obvious that if it is possible to generate all of the binary connectives and store the results in an orderly fashion, a full adder and/or a full subtracter exists.

Because of the ease of forming a ones complement in a binary system, a method presents itself for error checking the add or subtract operation. The method is, essentially, that an add and a subtract operation be carried out each time that add or subtract is executed, and the results be compared with each other. To explain this further if an add operation is desired, an add is performed, and, also, the ones complement of the addend or the augend is formed and the subtract operation is executed starting with an initial borrow. (The initial borrow changes the ones complement into a twos complement.)

The subtract operation proceeds in a like manner with the subtrahend being complemented and an initial carry propagated in the add part of the operation.

If the equipment is to be kept to a minimum, the same register can be used for obtaining the "complement" number on add or subtract by tieing the subtract circuit into the zero side of the register. This requires that on subtract the ones complement of the subtrahend be formed in the "complementing" register before subtraction is performed. An initial carry of one is started in the add part in this case.

The full subtracter works, also, for multiply where the borrow is propagated only one place to the left for each multiplier bit, until the multiplier is used up and then is propagated through the remaining places. Note that the extract operation is O_8 and is available without special equipment.

W A Hunt

W. A. Hunt

WAH:SG