INEQUALITY COMPARATOR*

SLAC-PUB-546
January 1969

by

M. Fishman and D. Horelick

Stanford Linear Accelerator Center
Stanford University
Stanford, California

One interesting method for comparing two binary numbers A and B involves the use of the carry logic of an adder summing A and the one's complement of B.

Let n = number of bits in A and B, so that the maximum count is $2^n - 1$. Then the one's complement of B is $2^n - 1 - B$, and the sum of this number and A is $2^n - 1 + (A-B)$. This sum has an overflow carry from the nth bit when A is greater than B, but has no carry when B is equal to or greater than A. Thus by implementing only the carry logic, one can construct a "greater than" comparator. The logic is described below.

Let C_1 be the carry generated when adding A_o and $\overline{B_o}$, the least significant digits. Then

$$C_1 = A_o \overline{B_o} .$$

Similarly, let C_2 be the carry generated when adding A_1, $\overline{B_1}$ and C_1. Then

$$C_2 = C_1 (A_1+\overline{B_1}) + A_1 \overline{B_1}$$

$$= C_1 A_1 + C_1 \overline{B_1} + A_1 \overline{B_1} .$$

In general, the j-th carry is given by

$$C_j = C_{j-1} A_{j-1} + C_{j-1} \overline{B_j} + A_j \overline{B_j} .$$

The carry generator may be easily realized by using DTL NAND gates to implement the "wired-or" as shown in Fig. 1 which illustrates a comparator of length n. Note that two gates for the least significant bit could be eliminated, but are included to demonstrate the completely iterative nature of the logic. When the complements of B are available, the comparator requires one chip per bit when using an integrated circuit such as the MC846P.

---

*Work supported by the U. S. Atomic Energy Commission

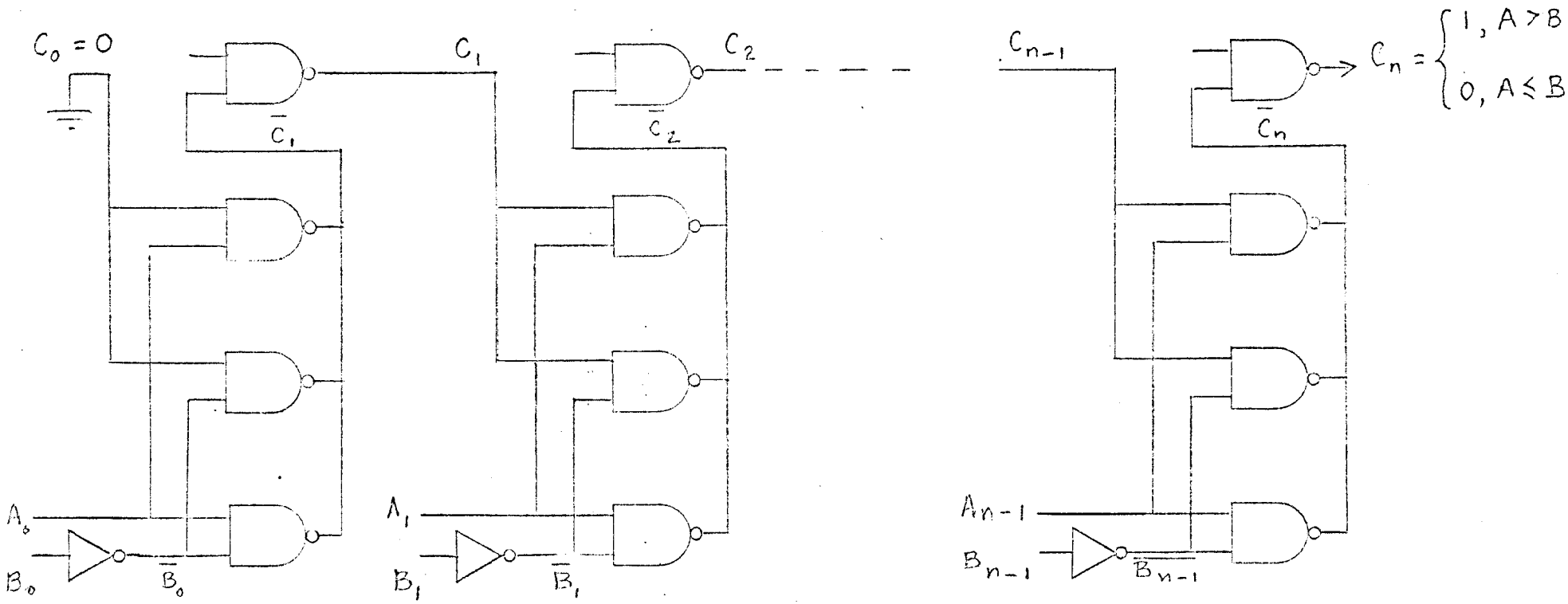

FIGURE 1. n-BIT "GREATER THAN" COMPARATOR

M. Fishman
D. Horelick
27 January 1969