CMOS LSI 4-Bit Arithmetic Logic Unit

The RCA-CD4057A is a low-power arithmetic logic unit (ALU) designed for use in LSI computers. An arithmetic system of virtually any size can be constructed by wiring together a number of CD4057A ALU's. The CD4057A provides 4-bit arithmetic operations, time sharing of data terminals, and full functional decoding for all control lines. The distributed control system of this device provides great flexibility in system designs by allowing hard-wired connection of N units in 4^N unique combinations. Four control lines provide 16 instructions which include Addition, Subtraction, Bidirectional and Cycle Shifts, Up-Down Counting, AND, OR, and Exclusive-OR logic operations.

Two mode control lines allow the CD4057A to function as any 4-bit section of a larger arithmetic unit by controlling the bidirectional serial transfer of data to adjacent arithmetic arrays. By means of three "Conditional Control" lines Overflow, All Zeros, and Negative State conditions may be detected and used to establish a conditional operation. Predetermined operation of the CD4057A on a conditional basis allows greater ALU flexibility. Although especially applicable as a parallel arithmetic unit, the CD4057A also finds use in virtually any application requiring one or more of its 16 basic instructions. The CD4057A is supplied in a hermetically sealed 28-lead dual-in-line ceramic package (CD4057AD), 28-lead ceramic flat package (CD4057AK), and in chip form (CD4057AH).

Applications:

- **Parallel Arithmetic Units**
- **Process Controllers**
- **Remote Data Sets**
- **Graphic Display Terminals**

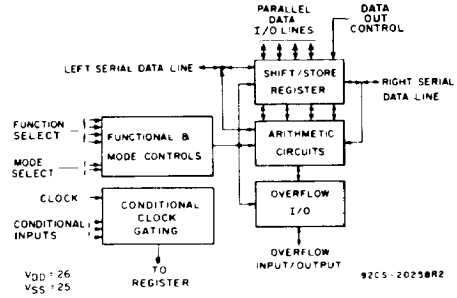


Fig. 1 — Block diagram — CD4057A.

Features:

- **LSI Complexity on a Single Chip**
- **16-Instruction Capability**
  - **-Add, Subtract, Count**
  - **-AND, OR, Exclusive-OR**
  - **-Right, Left, or Cyclic Shifts**
- **Bidirectional Data Busses**
- **Instruction Decoding on Chip**
- **Fully Static Operation**
- **Single-Phase Clocking**
- **Easily Expandable to 8, 12, 16,. . . . Bit Operation**
- **Low Quiescent Device Dissipation . . . . . 10 μW (typ.)**
- **Conditional-Operation Controls on Chip**
- **Add Time (Data In-To Sum Out) ≈375 ns (typ) at 10V**
- **Quiescent current specified to 15 V**
- **Maximum input leakage current of 1 μA at 15 V (full package-temperature range)**
- **1-V noise margin (full package-temperature range)**

MAXIMUM RATINGS, *Absolute-Maximum Values:*

STORAGE-TEMPERATURE RANGE (T_{stg})	−65 to +150°C
OPERATING-TEMPERATURE RANGE (T_A)	
PACKAGE TYPES D, K, H	−55 to +125°C
DC SUPPLY-VOLTAGE RANGE, (V_{DD})	
(Voltages referenced to V_{SS} Terminal)	−0.5 to +15 V
POWER DISSIPATION PER PACKAGE (P_D)	
For T_A = −55 to +100°C (PACKAGE TYPES D, K)	500 mW
For T_A = +100 to +125°C (PACKAGE TYPES D, K)	Derate Linearly at 12 mW/°C to 200 mW
DEVICE DISSIPATION PER OUTPUT TRANSISTOR	
FOR T_A = FULL PACKAGE-TEMPERATURE RANGE (ALL PACKAGE TYPES)	100 mW
INPUT VOLTAGE RANGE, ALL INPUTS	−0.5 to V_{DD} +0.5 V
LEAD TEMPERATURE (DURING SOLDERING):	
At distance 1/16 ± 1/32 inch (1.59 ± 0.79 mm) from case for 10 s max.	+265°C

RECOMMENDED OPERATING CONDITIONS

For maximum reliability, nominal operating conditions should be selected so that operation is always within the following ranges:

CHARACTERISTIC	V_{DD} (V)	LIMITS Min.	LIMITS Max.	UNITS
Supply-Voltage Range (For T_A = Full Package-Temperature Range)		3	12	V
Setup Time, t_S DATA	5	40	–	ns
	10	20	–	
OP CODE	5	4590	–	
	10	1320	–	
Clock Pulse Width, t_W	5	1200	–	ns
	10	375	–	
Clock Input Frequency, f_{CL} Count Mode	5	0.13	–	MHz
	10	0.46	–	
Shift Mode	5	0.33	–	
	10	1.4	–	
Clock Rise or Fall Time, t_rCL, t_fCL	5	–	15	μs
	10	–	15	

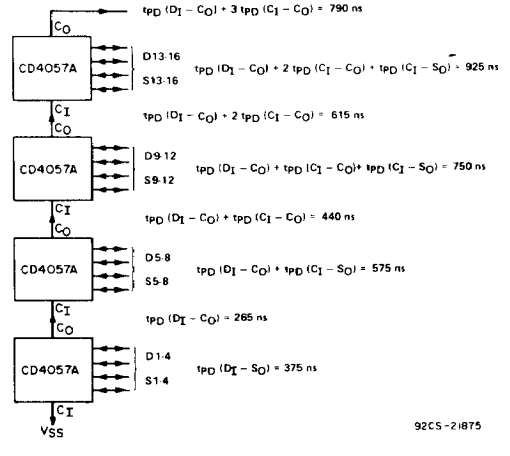


Fig. 2 — Typical speed characteristics of a 16-bit ALU at V_{DD} = 10 V.

STATIC ELECTRICAL CHARACTERISTICS

CHARACTERISTIC	CONDITIONS			Limits at Indicated Temperatures (°C)							UNITS
				CD4057AD, CD4057AK, CD4057AH							
	V_O (V)	V_{IN} (V)	V_{DD} (V)	−55°C		25°C			125°C		
				Min.	Max.	Min.	Typ.	Max.	Min.	Max.	
Quiescent Device Current I_L	–	–	5	–	5	–	0.5	5	–	150	μA
	–	–	10	–	10	–	1	10	–	300	
	–	–	15	–	50	–	1	50	–	2000	
Output Voltage; Low-Level, V_{OL}	–	5	5	0 Typ.; 0.05 Max.							V
	–	10	10	0 Typ.; 0.05 Max.							
High Level, V_{OH}	–	0	5	4.95 Min.; 5 Typ.							V
	–	0	10	9.95 Min.; 10 Typ.							
Noise Immunity (All Inputs) V_{NL}, V_{NH}	0.8	–	5	1.5 Min.; 2.25 Typ.							V
	1	–	10	3 Min.; 4.5 Typ.							
	4.2	–	5	1.5 Min.; 2.25 Typ.							
	9	–	10	3 Min.; 4.5 Typ.							
Noise Margin: Inputs Low, V_{NML}	4.5	–	5	1 Min.							V
	9	–	10	1 Min.							
Inputs High, V_{NMH}	0.5	–	5	1 Min.							
	1	–	10	1 Min.							
Output Drive Current: I_DN, I_DP Zero Indicator	0.5	–	5	0.11	–	0.09	0.16	–	0.06	–	mA
n-channel	0.5	–	10	0.12	–	0.10	0.16	–	0.07	–	
p-channel	3	–	5	0.04	–	0.03	0.06	–	0.02	–	
	7	–	10	0.08	–	0.07	0.13	–	0.05	–	
Negative Indicator	0.5	–	5	0.11	–	0.09	0.30	–	0.06	–	
n-channel	0.5	–	10	0.12	–	0.10	0.40	–	0.07	–	
p-channel	4.5	–	5	0.07	–	0.06	0.19	–	0.04	–	
	9.5	–	10	0.12	–	0.10	0.30	–	0.07	–	
Overflow Indicator	0.5	–	5	0.25	–	0.20	0.50	–	0.14	–	
n-channel	0.5	–	10	0.37	–	0.30	0.90	–	0.21	–	
p-channel	4.5	–	5	0.08	–	0.07	0.21	–	0.05	–	
	9.5	–	10	0.12	–	0.10	0.38	–	0.07	–	
All Other Outputs	0.5	–	5	0.11	–	0.09	0.10	–	0.06	–	
n-channel	0.5	–	10	0.06	–	0.05	0.12	–	0.03	–	
p-channel	4.5	–	5	0.02	–	0.02	0.05	–	0.01	–	
	9.5	–	10	0.06	–	0.05	0.08	–	0.03	–	
Input Leakage Current I_{IL}, I_{IH}	Any Input –	–	15	$\pm 10^{-5}$ Typ., ± 1 Max.							μA

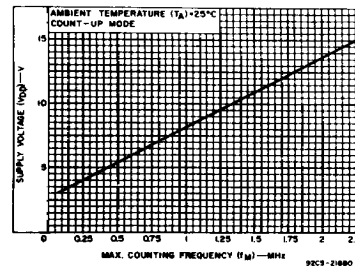


Fig. 3 — Maximum counting frequency vs. supply voltage for a typical CD4057A.

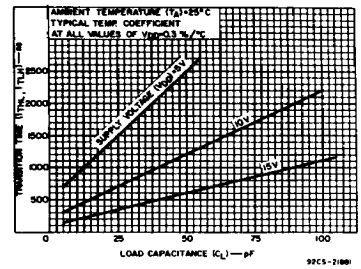


Fig. 4 — Transition time vs. load capacitance for data outputs (D1-D4).

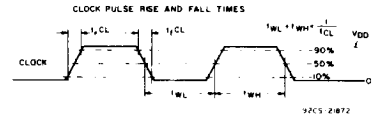


Fig. 5 — Clock pulse rise and fall times.

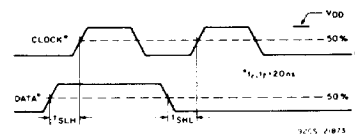


Fig. 6 — Data setup time.

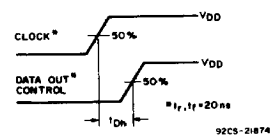


Fig. 7 — Data hold time.

DYNAMIC ELECTRICAL CHARACTERISTICS at T_A = 25°C and C_L = 15 pF, R_L = 200 kΩ, t_r,t_f = 20 ns

Typical Temperature Coefficient at all values of V_{DD} = 0.3%/°C

CHARACTERISTICS	TEST CONDITIONS	LIMITS CD4057AD, CD4057AK			UNITS
	V_{DD}	Min.	Typ.	Max.	
Propagation Delay Time: t_{PLH},t_{PHL} DATA IN-to-SUM OUT	5	–	1430	3900	ns
	10	–	375	720	
CARRY IN-to-SUM OUT	5	–	915	2550	
	10	–	310	840	
DATA IN-to-CARRY OUT	5	–	950	2580	
	10	–	265	720	
CARRY IN-to-CARRY OUT	5	–	485	1320	
	10	–	175	480	
ZI Input -to- ZI Output	5	–	1980	5400	
	10	–	750	2040	
	5	–	265	720	
	10	–	110	300	
Transition Time: t_{TLH},t_{THL} ZI Output	5	–	3700	10350	ns
	10	–	1650	4500	
	5	–	420	1140	
	10	–	220	600	
Negative Indicator and Overflow Indicator	5	–	300	825	
	10	–	165	450	
All Other Outputs	5	–	1000	2775	
	10	–	475	1275	
Minimum Clock Pulse Width, t_W	5	–	400	1200	ns
	10	–	125	375	
Clock Rise and Fall Time, t_rCL,t_fCL	5	–	–	15	μs
	10	–	–	15	
Minimum Set Up Time : t_{SLH},t_{SHL} DATA	5	–	20	40	ns
	10	–	10	20	
OP CODE	5	–	1675	4590	ns
	10	–	485	1320	
Minimum Data Hold Time, t_{HLH}, t_{HHL}	5	–	20	40	ns
	10	–	10	20	
Maximum Clock Frequency: f_{CL} Count Mode	5	0.13	0.36	–	MHz
	10	0.46	1.35	–	
Shift Mode	5	0.33	0.90	–	
	10	1.4	3.8	–	
Input Capacitance, C_I	ANY INPUT	–	5	–	pF

LOGIC DESCRIPTION

OPERATIONAL MODES

The CD4057A arithmetic logic unit operates in one of four possible modes. These modes control the transfer of information, either serial data or arithmetic operation carries, to and from the serial-data lines. Fig. 8 shows the manner in which the four modes control the data on the serial-data lines.

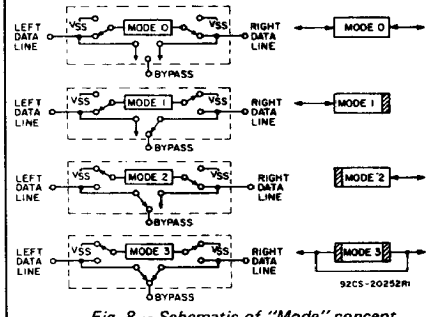


Fig. 8 – Schematic of "Mode" concept.

In MODE 0, data can enter or leave from either the left or the right serial-data line.

In MODE 1, data can enter or leave only on the left serial-data line.

In MODE 2, data can enter or leave only on the right serial-data line.

In MODE 3, serial data can neither enter nor leave the register,regardless of the nature of the operation. Furthermore, the register is by-passed electrically, i.e., there is an electrical bidirectional path between the right and left serial data terminals.

The two input lines labeled C1 and C2 in the terminal assignment diagram define one of four possible modes shown in Table I. Through the use of mode control, individual arithmetic arrays can be cascaded to form one large processor or many processors of various lengths.

TABLE I – MODE DEFINITION

C2	C1	MODE
0	0	0
0	1	1
1	0	2
1	1	3

Examples of how one " hard-wired " combination of three ALU's can form (a) a 12-bit parallel processor, (b) one 8-bit and one 4-bit parallel processor, or (c) three 4-bit parallel processors, merely by changes in the modes of each ALU are shown in Fig. 10.

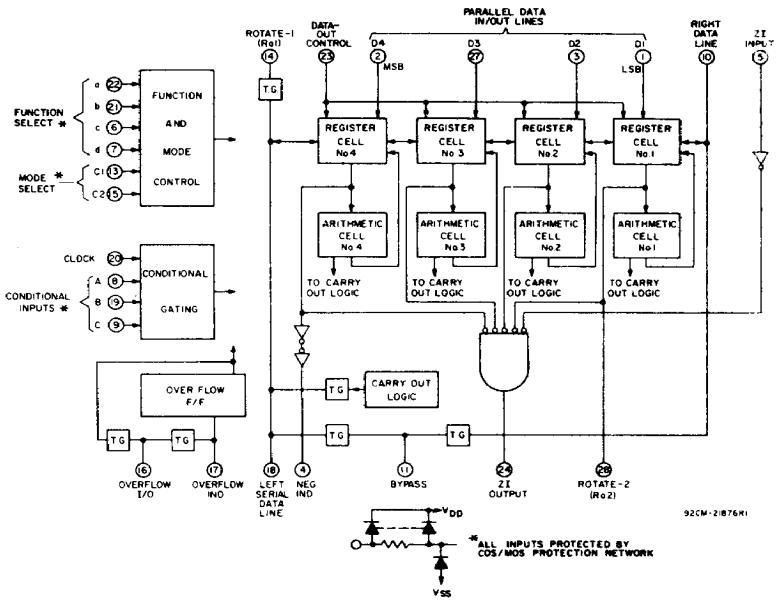


Fig. 9 – Simplified logic diagram.

Data-flow interruptions are shown by shaded areas. With these three ALU's and the four available modes, 61 more system combinations can be formed. If 4 ALU's are used, 4^4 combinations (256) are possible. *Fig.11* shows a diagram of 4 CD4057A's interconnected to form a 16-bit parallel processor.

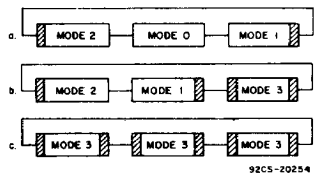


Fig. 10 – "Mode" connections for parallel processor:
(a) 12-bit unit,
(b) one 8-bit and one 4-bit unit
(c) three 4-bit units.

NOTE: The BYPASS terminal of the "most significant" CD4057A is connected to the bypass terminal of the "least significant" CD4057A. The bypass terminals on all other CD4057A's are left floating. This interconnection is performed whenever more than one CD4057A are used to form a processor.

INSTRUCTION REPERTOIRE

Four encoded lines are used to represent 16 instructions. Encoded instructions are as follows:

a	b	c	d	
0	0	0	0	NO-OP (Operational Inhibit)
0	0	0	1	AND
0	0	1	0	Count down
0	0	1	1	Count up
0	1	0	0	Subtract Stored number from zero (SMZ)
0	1	0	1	Subtract from parallel data lines (SM) (stored number from parallel data lines)
0	1	1	0	Add (AD)
0	1	1	1	Subtract (SUB) (Parallel data lines from stored number)
1	0	0	0	Set to all ones (SET)
1	0	0	1	Clear to all zeroes (CLEAR)
1	0	1	0	Exclusive-OR
1	0	1	1	OR
1	1	0	0	Input Data (From parallel data lines)
1	1	0	1	Left shift
1	1	1	0	Right shift
1	1	1	1	Rotate (cycle) right

All instructions are executed on the positive edge of the clock.

PARALLEL COMMANDS

a. **CLEAR** – sets register to zero.
b. **SET** –sets register to all ones.
c. **OR** –processes contents of register with value on parallel-data lines in a logical OR function.
d. **AND** –processes contents of register with value on parallel-data lines in a logical AND function.
e. **Exclusive-OR** – processes contents of register with data on parallel-data lines in a logical Exclusive-OR function.
f. **IN** –loads data on parallel-data lines into register.
g. **DATA OUT CONTROL** – unloads contents of register and overflow flip-flop onto parallel data lines and overflow I/O independent of all other controls.
h. **SUB:**
   In Mode 0, adds to the contents of the register the one's complement of the data on the parallel-data lines. Carries can enter on the right serial data line and can leave on the left serial data line. The overflow indicator does not change state.
   In Mode 1, adds to the contents of the register the two's complement of the data on the parallel-data lines. Generated carries can leave on the left serial line. The CARRY IN is set to zero. The overflow indicator does not change state.
   In Mode 2, same as Mode 0, except carries cannot leave on the right serial-data line.The absence or presence of an overflow is registered.
   In Mode 3, same as Mode 1, except carries cannot leave on the left serial-date line. The absence or presence of an overflow is registered.
i. **COUNT UP:**
   In Mode 0, adds to the contents of the register the data on the right serial-data line and permits any resulting carry to leave on the left serial-data line. No data enters the parallel-data lines.
   In Mode 1, internally adds a one to the contents of the register and permits any resulting carry to leave on the left serial-data line. No data enters or leaves the right serial-data line.
   In Mode 2, adds to the contents of the register the data on the right serial-data line. No data enters or leaves the left serial-data line.
   In Mode 3, internally adds a one to the contents of the register No data enters or leaves the register on any serial-data or parallel-data line.
   In all modes, with the DATA OUT control high

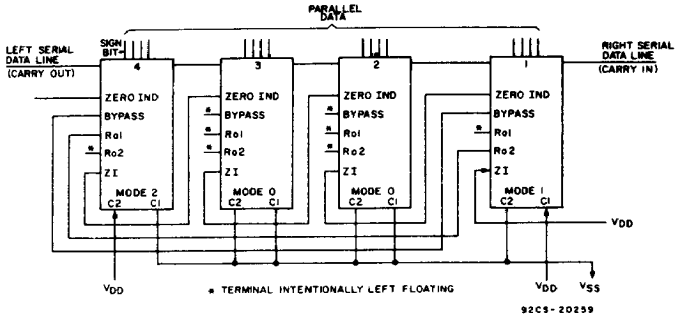


Fig. 11 — Connection for 16-bit arithmetic logic unit.

the count is presented on the parallel data lines (D1-D4).

j. COUNT DOWN:

In Mode 0, subtracts a one (2's complement form) from the contents of the register and adds to this result the data on the right serial-data line and permits any resulting carry to leave on the left serial-data line. No data enters on the parallel-data lines.

In Mode 1, internally subtracts a one from the contents of the register and permits any resulting carry to leave on the left serial-data line. No data enters or leaves the right serial-data line.

In Mode 2, subtracts a one from the contents of the register and adds to this result the data on the right serial-data line. No data enters or leaves on the left serial-data line.

In Mode 3, internally subtracts a one from the contents of the register. No data enters or leaves on the serial-data lines.

In all modes, with the DATA OUT control high the count is presented on the parallel data lines (D1-D4).

k. ADD(AD):

In Mode 0, adds the contents of the register to the data on the parallel-data lines and the right serial-data line. Any resulting carry can leave on the left serial-data line. The overflow indicator does not change state.

In Mode 1, adds the contents of the register to the data on the parallel-data lines and allows any resulting carry to leave on the left serial-data line. The right serial-data line is open-circuited. The overflow indicator does not change state. The CARRY-IN is set to zero.

In Mode 2, adds the contents of the register to the data on the parallel data lines and the right serial-data line. Any overflow sets the overflow indicator. The left serial-data line is open-circuited. The absence or presence of an overflow is registered.

In Mode 3, adds contents of the register to the data on the parallel-data lines. Any resulting carry sets the overflow indicator. The two serial-data lines are open circuited. The absence or presence of an overflow is registered. The CARRY-IN is set to zero.

l. SM — same operation as AD except the contents of the register are two's complemented during addition in Mode 1 and Mode 3. In Mode 0 or Mode 2, the contents of the register are one's complemented and added to the data on the right serial-data line and the parallel-data lines. Overflows occurring in Mode 1 or Mode 0 do not alter the overflow indicator. The presence or absence of overflows is registered on the overflow indicator in Mode 2 or Mode 3.

m. SMZ:

In Mode 0, one's complements the contents of the register and adds the data on the right serial-data line to the contents of the register. Any resulting carry can leave on the left serial-data line. The overflow indicator does not change state.

In Mode 1, two's complements the contents of the register and permits any carry to leave on the left serial-data line. No data can enter the right serial-data line. The overflow indicator does not change state. The CARRY-IN is set to zero.

In Mode 2, one's complements the contents of the register and adds the data on the right serial-data line to the contents of the register. Carries cannot leave the left serial data line. The absence or presence of an overflow alters the overflow indicator.

In Mode 3, two's complements the contents of the register. Serial data can neither enter the right serial-data line nor leave the left serial-data line. The overflow indicator is at zero. The CARRY-IN is set to zero.

n. NO-OP — no operation takes place, The clock input is inhibited and the state of all registers and indicators remains unchanged.

SERIAL-SHIFT OPERATIONS

a. ROTATE (cycle) RIGHT — This operation is internal. The contents of the register shift to the right, cyclic fashion with the leftmost stage accepting data from the rightmost stage regardless of the mode. Data can leave the register serially on the right data line only while the register is in Mode 1 or Mode 0. Data can enter the left data line serially while the register is in Mode 1 or Mode 0. The Ro1 terminal of the "Most Significant" CD4057A must be connected to the Ro2 terminal of the "Least Significant" CD4057A. All other Ro1 and Ro2 terminals must be left floating. When only one CD4057A is used, Ro1 must be connected to Ro2.

b. RIGHT SHIFT — The contents of the register shift to the right and serial operations are as follows:

In Mode 0, data can enter serially on the left data line, shift through the register, and leave on the right data line.

In Mode 1, data can enter serially on the left data line. The right data line effectively is open - circuited.

In Mode 2, data can leave serially on the right data line. The left data line effectively is open-circuited. Vacant spaces are filled with zeros.

In Mode 3, serial data can neither enter nor leave the register; however,the contents shift to the right and vacated places are filled with zeros.

In all modes, with the DATA OUT control high the data is presented on the parallel data lines (D1-D4).

c. **LEFT SHIFT** – The contents of the register shift to the left and serial operations are as follows;
   In Mode 0, data can enter the right data line, shift through the register, and leave on the left data line.
   In Mode 1, data can leave serially on the left data line. The right data line effectively is open-circuited. All vacant positions are filled with zeros.
   In Mode 2, data can enter serially on the right data line. The left data line effectively is open-circuited.
   In Mode 3, data can neither enter nor leave the register; however, the contents shift to the left, and vacated places are filled with zeros.
   In all modes, with the DATA OUT control high the data is presented on the parallel data lines (D1-D4).

Because the "DATA OUT" control instruction is independent of the other 16 instructions, care must be taken not to activate this control when data are to be loaded into the processor. This instruction should only be activated when the processor is executing a NO-OP, any SHIFT, SMZ, COUNT UP or DOWN, CLEAR, or SET.

If a data line, serial or parallel, is used as an input and the logic state of that line is not defined (i.e., the line is an open circuit),then the result of any operation using that line is undefined.

OPERATIONAL SEQUENCE FOR ARITHMETIC ADD CYCLE

1. Apply IN Instruction and Word A on Parallel Data Lines (D1–D4).
2. Apply CLOCK to load Word A into the register.
3. Apply OP CODE Instruction and Word B on Data Lines.
4. Apply CLOCK to load resulting function of A and B into the register.
5. Apply "DATA OUT" control to present result to Parallel Data Lines.

NOTE: Transitions of Step 2 and Step 3 may occur almost simultaneously: i.e. separated by only one data-hold time.

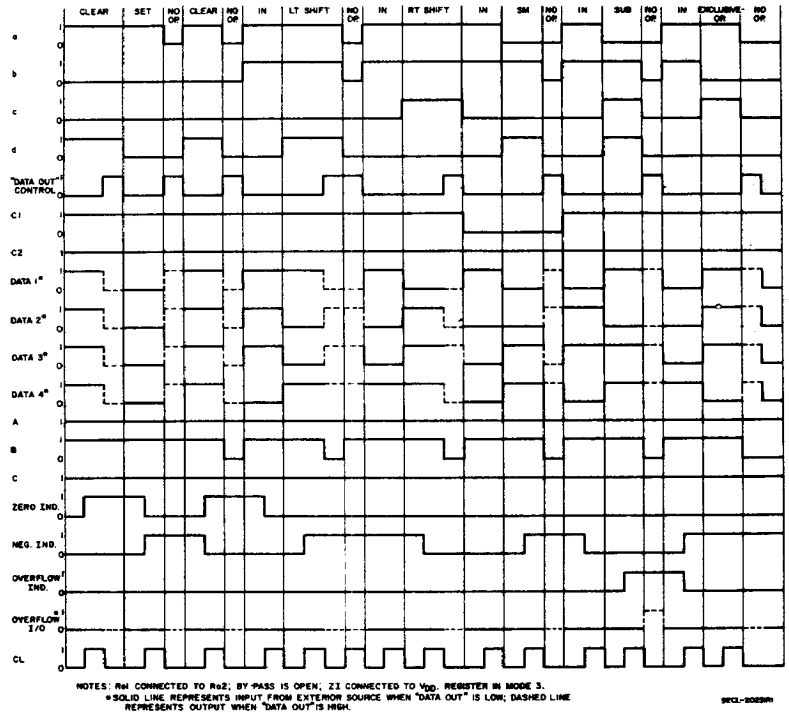


Fig. 12 – Timing diagram.

NEGATIVE-NUMBER DETECTION

The NEG IND terminal of the CD4057A is connected to the output of the flip-flop that is in the most significant bit position. A "1" on the NEG IND terminal indicates a negative number is in the register. This detection is also independent of modes.

ZERO DETECTION

The condition of "all zeros" is indicated by a "1" on the Zero Indicator terminal of the "Most Significant" CD4057A.
As shown in Fig. 11, terminal ZI of the CD4057A containing the least significant set of bits is connected to V_{DD}. Zero indication is independent of modes.

COMPLEMENTING NUMBERS

1. One's complement of number in ALU register.
   a) ALU must be in MODE 0 or MODE 2.
   b) Zero on Rt. Data Line.
   c) Execute an SMZ instruction.

(Continued)

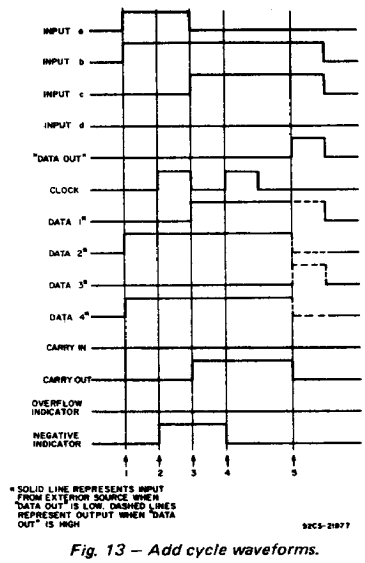
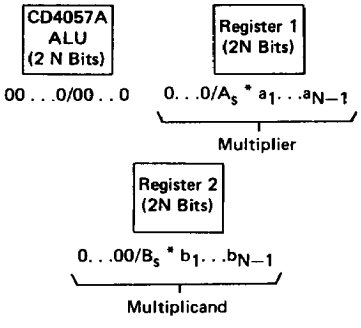


Fig. 13 – Add cycle waveforms.

2. One's complement of number to be loaded into ALU register.
   a) If zero indicator output is low, execute a CLEAR instruction, and make Rt. Data Line = 0.
   b) ALU must be in MODE 0 or MODE 2.
   c) Execute an SUB instruction.
3. Two's complement of number in ALU register.
   a) ALU must be in MODE 1 or MODE 3.
   b) Execute an SMZ instruction.
4. Two's complement of number to be loaded into ALU register.
   a) If zero indicator output is low, execute a CLEAR instruction, and make Rt. Data Line = 0.
   b) ALU must be in MODE 1 or MODE 3.
   c) Execute an SUB instruction.

The following algorithms are given as a general guideline to demonstrate some of the capabilities of the CD4057A.

MULTIPLICATION OF TWO N-BIT NUMBERS

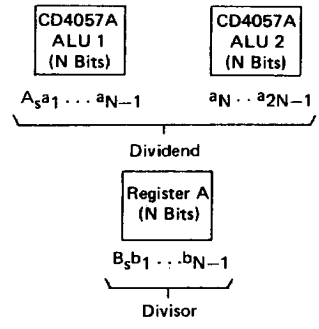


*A_s and B_s are sign bits

Multiplication Algorithm

1. Clear ALU to Zero
2. Store $A_s \oplus B_s$ in External Flip-Flop.
3. If A_s = 1, Complement Register 1.
4. If B_s = 1, Complement Register 2.
5. Load Register 2 into ALU.
6. Do shift Left on ALU N Times (N = number of bits).
7. Do N Times:
   (1)
   a) If MSB of ALU = 1 (Negative Indicator = High), Then shift ALU left 1 bit; add Register 1 to ALU.
   b) If MSB of ALU = 0 (Negative Indicator = Low) Then shift ALU left 1 bit.
8. If $A_s \oplus B_s$ = 1, then Complement ALU.
9. Answer in ALU.

Division Algorithm



1. Store $A_s \oplus B_s$ in External Flip-Flop.
2. If A_s = 1, complement ALU 1 and ALU 2.
3. If B_s = 1, complement Register A.
4. Check for Divisor = 0
   a) If Divisor = 0; stop, indicates division by 0.
   b) If Divisor $\neq$ 0; continue.
5. Apply SUB instruction to ALU 1 and the contents of Register A to ALU 1 data lines.
6. Put a zero on RT data line of ALU 2 and shift ALU 1 & ALU 2 left 1 bit.
7. Do "N" times.
   (1) Apply a sub instruction to ALU 1 and the contents of Register A to the ALU 1 data lines.
   a) If C_O = 1, then clock ALU 1, and put a 1 on right data line of ALU 2.
   b) If C_O = 0, then do not clock, and put a 0 on right data line of ALU 2.
   (2) Shift left 1 bit.
8. If sign Flip Flop = 1, complement ALU 2.
9. Answer in ALU 2.

CONDITIONAL OPERATION

Inhibition of the clock pulse can be accomplished with a programmed NO-OP instruction or through conditional input terminals A,B,and C.In a system of many CD4057A's, each CD4057A can be made to automatically control its own operation or the operation of any other CD4057A in the system in conjunction with the Overflow, Zero, or Negative (Number) indicators. Table II, the conditional inputs, truth table, defines the interactions among A,B, and C.

TABLE II – CONDITIONAL-INPUTS TRUTH TABLE

A	B	C	OPERATION PERMITTED
0	X	X	Yes
1	0	0	Yes
1	0	1	No
1	1	0	No
1	1	1	Yes

X = don't care

Two examples of how the conditional operation can be used are as follows:

1) For the Multiplication Algorithm
   A = 1, for step 7 (1)
   A = 0, for step 7 (2)
   B = 1
   C = negative Indicator
2) For the Division Algorithm
   A = 1, for step 7 (1)
   A = 0, for step 7 (2)
   B = 1
   C = C_O (left data line)

OVERFLOW DETECTION

The CD4057A is capable of detecting and indicating the presence or absence of an arithmetic two's-complement overflow. A two's-complement overflow is defined as having occurred if the signs of the two initial words are the same and the sign of the result is different while performing a carry-generating instruction.

For example:
$$\begin{array}{rr} & 0.011 \\ (+) & \underline{0.110} \\ & 1.001 \end{array}$$

Overflows can be detected and indicated only during operation in Mode 2 or Mode 3 and can occur for only four instructions (AD, SMZ, SM, and SUB). If an overflow is detected and stored in the overflow flip-flop, any one of the five instructions AD, SMZ, SM, SUB, or IN can change the overflow indicator.

When any of the three subtraction instructions is used, the sign bit of the data being subtracted is complemented and this value is used as one of the two initial signs to detect overflows. If an overflow has occurred, the final sign of the sum or difference is one's complemented and stored in the most-significant-bit position of the register.

The overflow flip-flop is updated at the same time the new result is stored in the CD4057A. Whenever data on the parallel-data lines are loaded into the CD4057A, whatever is on the Overflow I/O line is loaded into the overflow flip-flop. Also, whenever data are dumped on the parallel data lines from the CD4057A, the contents of the overflow flip-flop are dumped on the Overflow I/O line. Thus overflows may be stored elsewhere and then fed into the CD4057A at another time.

OPERATIONAL SEQUENCE AND WAVEFORMS FOR PROPAGATION-DELAY MEASUREMENTS

1. DATA-IN-to-CARRY OUT and DATA IN-to-SUM OUT
   - A. Apply Word A and IN instruction
   - B. Apply Clock to load word A into register
   - C. Apply AD instruction
   - D. Apply Word B (data in)
   - E. Apply Clock to load result (sum out)
   - F. Apply DATA OUT CONTROL to look at result

2. CARRY IN-to-CARRY OUT and CARRY IN-to-SUM OUT
   - A. Apply Word A and IN instruction
   - B. Apply Clock to load word A into register
   - C. Apply AD instruction
   - D. Apply Word B
   - E. Apply CARRY IN (carry in)
   - F. Apply Clock to load result (sum out)
   - G. Apply DATA OUT CONTROL to look at result

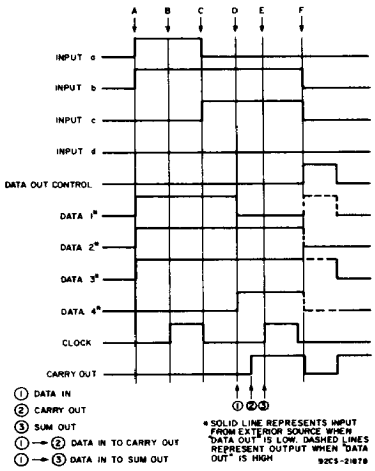


Fig. 14(a) – DATA IN-to-CARRY OUT and DATA IN-to-SUM OUT.

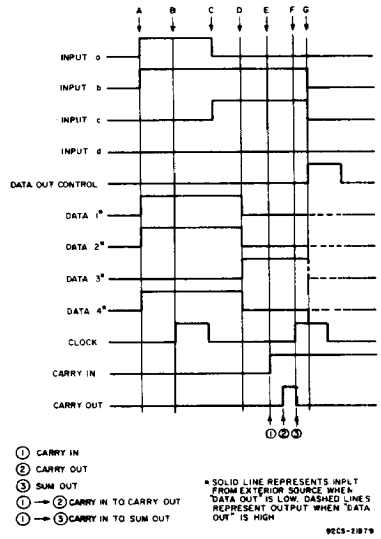


Fig. 14(b) – CARRY IN-to-CARRY OUT and CARRY IN-to-SUM OUT.

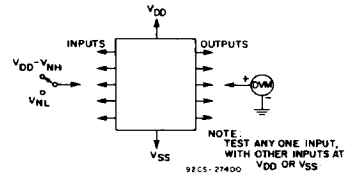


Fig. 15 – Noise-immunity test circuit.

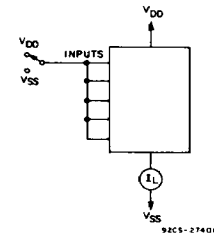


Fig. 16 – Quiescent-device-current test circuit

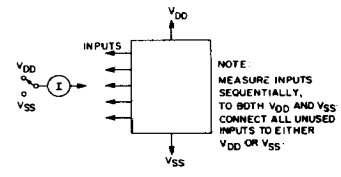


Fig. 17 – Input-leakage-current test circuit.

TYPICAL APPLICATION

The CD4057A has been designed for use as a parallel processor in flexible, programmable, easily expandable, special or general purpose computers, where minimization of external connections and data busing are primary design goals. The block diagram of Fig. 18 is an example of a computer that processes 8 bits in parallel.

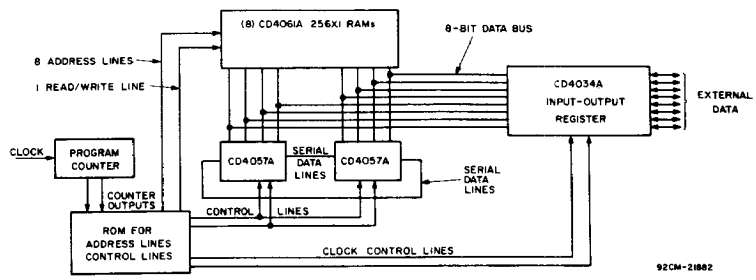


Fig. 18 – Example of computer organization using CD4057A.