T-46-13-47

Am29PL142

Field-Programmable Controller (FPC)

DISTINCTIVE CHARACTERISTICS

- Implements complex state machines
- 8 conditional inputs, 16 outputs
- 128-word by 34-bit PROM
- Up to 20-MHz clock rate, 28-pin DIP
- 28 instructions
  - Conditional branching
  - Conditional looping
  - Conditional subroutine call
  - Multiway branch
- Output instruction presents counter contents at the control outputs for implementing a larger class of state-machine designs
- A controller-expansion (EXP) fuse provides address to external registered PROMs for more than 16 outputs
- SSR fuse configures Serial Shadow Register (SSR™) diagnostics on chip

GENERAL DESCRIPTION

The Am29PL142 is a single-chip Field-Programmable Controller (FPC) that allows implementation of complex state machines and controllers by programming the appropriate sequence of instructions. Jumps, loops, and subroutine calls, conditionally executed based on the test inputs, provide the designer with powerful control flow primitives.

Intelligent control may be distributed throughout the system by using FPCs to control various self-contained functional units, such as register file/ALU, I/O, interrupt, diagnostic, and bus-control units.

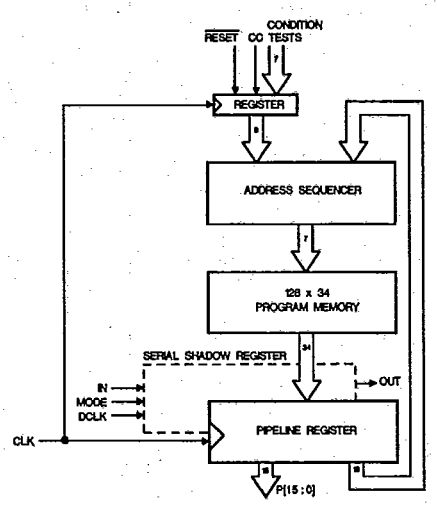
An address sequencer, the heart of the FPC, provides the address to an internal 128-word by 34-bit PROM. The fuse programming algorithm is almost identical to that used for AMD's Programmable Array Logic family.

The Am29PL142 can be expanded to address external registered PROMs by using the EXP fuse option to output the program-memory address through the control output pins P[14:8].

A counter register is provided and an instruction is available to present the counter-register contents at the control outputs P[14:8]. Using this, the control outputs can be dynamically modified for implementing a larger class of state machines.

As an option, the Am29PL142 may be programmed to have on-chip SSR diagnostics capability. Instructions can be serially shifted in, executed, and the results shifted out to facilitate system diagnostics.

SIMPLIFIED BLOCK DIAGRAM



BD007540

SSR is a trademark of Advanced Micro Devices, Inc.

Publication # 09390 | Rev. A | Amendment /0
Issue Date: March 1988

RELATED AMD PRODUCTS

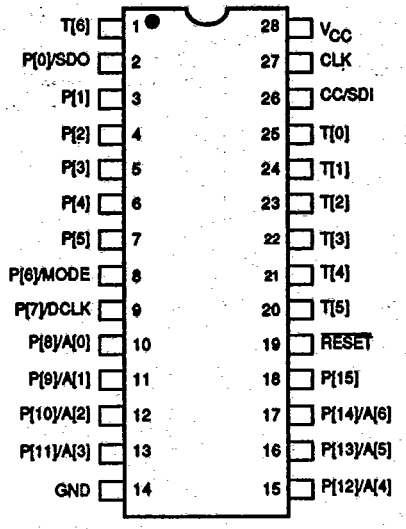
T-46-13-47

Part No.	Description
Am29C01-1	High-Speed CMOS 4-Bit Microprocessor Slice
Am29C10A-1	High-Speed CMOS 12-Bit Sequencer
Am29C101-1	High-Speed CMOS 16-Bit Microprocessor
Am29114	8-Level Real-Time Interrupt Controller (Expandable)
Am29116	16-Bit Bipolar Microprocessor (Supports 100 ns System Cycle Time)
Am29C116	CMOS Version of Am29116
Am29C116-1/-2	High-Performance Versions of Am29C116
Am29117	2-Port Version of Am29116
Am29C117	CMOS Version of Am29117
Am29C117-1/-2	High-Performance Versions of Am29C117
Am29118	8-Bit Am29116 I/O Support
Am29130	16-Bit Barrel Shifter (Expandable)
Am29PL131	24-Pin Slim DIP Version of the Am29PL141
Am29LPL131	Low Power Version of Am29PL131
Am2914	Vectored Priority Interrupt Controller
Am29LPL141	Low-Power Version of the Am29PL141
Am29PL141	Field-Programmable Controller
Am29CPL141	CMOS Version of the Am29PL141
Am29CPL144	CMOS High-Density Field-Programmable Controller
Am29C325	CMOS 32-Bit Floating Point Processor
Am29C327	CMOS Double-Precision Floating Point Processor
Am29337	16-Bit Bounds Checker
Am2940	DMA Address Generator
Am29C516A-1	35-ns CMOS 16 x 16 Multiplier
Am29818	SSR Diagnostics Pipeline Register
Am29C818	CMOS Version of the Am29818

T-46-13-47

CONNECTION DIAGRAM
Top View

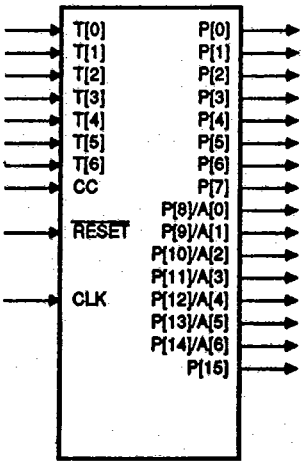
DIPs



CD011100

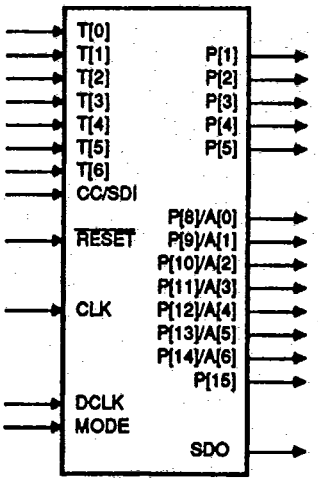
Note: Pin 1 is marked for orientation.

LOGIC SYMBOLS



CD011230

Normal Configuration

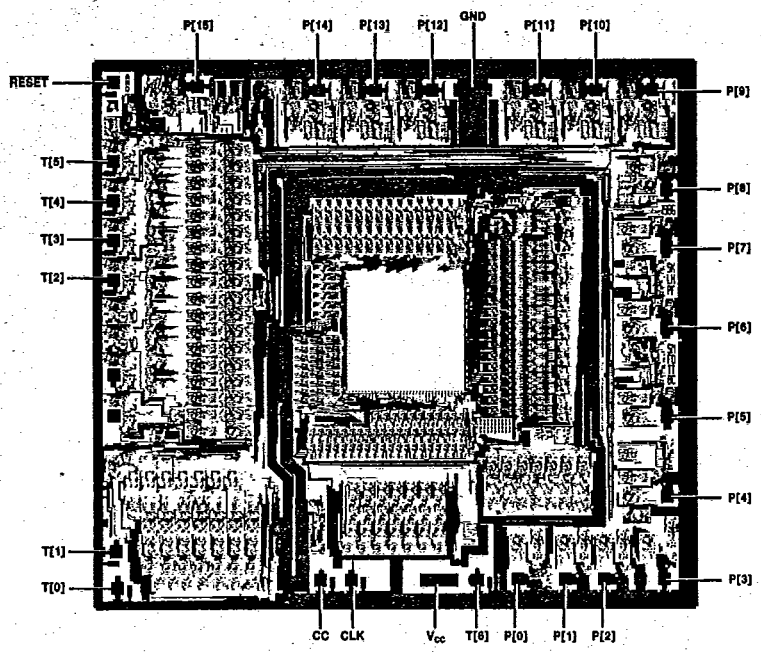


CD011241

SSR Diagnostics Configuration

T-46-13-47

METALLIZATION AND PAD LAYOUT



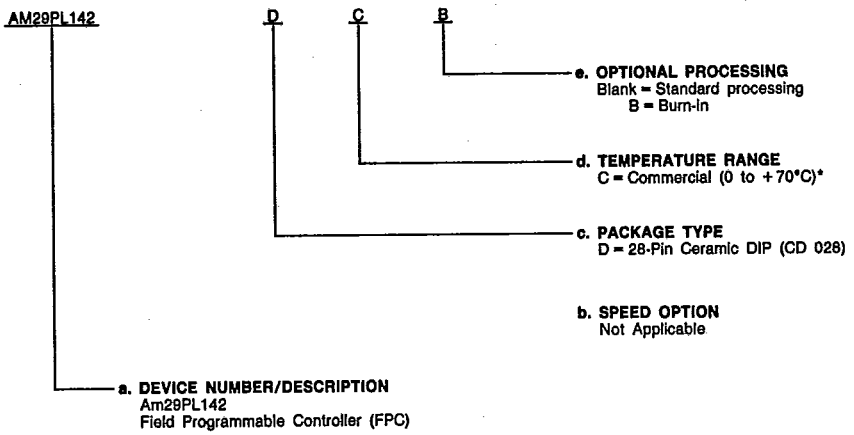
Die Size: 0.261″ x 0.238″
Gate Count: 600 Equivalent Gates and 4K of PROM

T-46-13-47

ORDERING INFORMATION
Standard Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of:

a. Device Number
b. Speed Option (if applicable)
c. Package Type
d. Temperature Range
e. Optional Processing



Valid Combinations	
AM29PL142	DC, DCB

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, to check on newly released valid combinations, and to obtain additional data on AMD's standard military grade products.

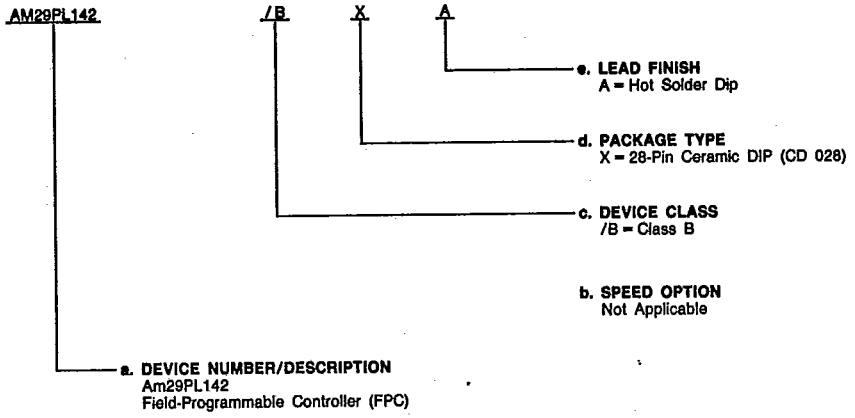
*The Commercial Operating Range is 0 to +85°C Case temperature for the Leadless Chip Carrier Package, due to the power constraints of that particular package.

MILITARY ORDERING INFORMATION
APL Products

T-46-13-47

AMD products for Aerospace and Defense applications are available in several packages and operating ranges. APL (Approved Products List) products are fully compliant with MIL-STD-883C requirements. The order number (Valid Combination) is formed by a combination of:

a. Device Number
b. Speed Option (if applicable)
c. Device Class
d. Package Type
e. Lead Finish



Valid Combinations	
AM29PL142	/BXA

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations or to check for newly released valid combinations.

Group A Tests

Group A tests consists of Subgroups 1, 2, 3, 7, 8, 9, 10, 11.

T-46-13-47

PIN DESCRIPTION

CC[SDI] Condition Code — TEST (Input)
Internally synchronized condition code test input is selected through the 4-bit test-condition select field. In the SSR mode, CC is also the Serial Data Input (SDI).

CLK Clock (Input)
The rising edge of the clock latches the PC register, count register, stack register, instruction-pipeline register, test-input register, CC register, reset register, and the EQ flag.

P[15], P[14:8]/A[6:0] Upper General-Purpose Control (Outputs)
The upper eight general-purpose control outputs are enabled by the OE bit from the instruction-pipeline register. When OE is HIGH, these outputs are enabled; when OE is LOW, they are three stated.

A controller Expansion (EXP) fuse can be programmed to set pins P[14:8] to output the program address A[6:0] from the PC MUX. These can be used to address external registered PROMs to provide more control outputs.

The contents of the internal count register (CREG) can also be routed to the control output pins P[14:8], using the OUTPUT instruction. Thus, the control outputs can be changed dynamically.

P[7:0] Lower General-Purpose Control (Outputs)
The lower eight general-purpose control outputs are permanently enabled. In the SSR diagnostics mode, P[7] becomes the diagnostic clock input (DCLK), P[6] becomes the diagnostic control input (MODE), and P[0] becomes the Serial Data Output (SDO). Both SDO and SDI are clocked by DCLK under control of the MODE bit.

$\overline{\text{RESET}}$ Internally Synchronized $\overline{\text{Reset}}$ (Input)
After it goes LOW, $\overline{\text{RESET}}$ is latched internally on the first rising edge of the clock. During the first clock cycle, the PC MUX is set to all ones (address 127). On the next rising edge of the clock, the program-memory contents at location 127 are loaded into the pipeline register. The EQ flag is also cleared at this time.

T[6:0] Internally Synchronized Test (Inputs)
In conditional instructions, these inputs are selected according to the 4-bit test condition select field. The inputs T[6:0] can also be used either as a branch address or as a value to be loaded into CREG depending on the instruction.

FUNCTIONAL DESCRIPTION

Figure 1, the detailed block diagram of the Am29PL142 FPC, shows logic blocks and interconnecting buses that permit parallel performance of different operations in a single instruction. The FPC consists of four main logic blocks: the program memory, address control logic, condition code selection logic, and instruction decode. A fifth optional block is the Serial Shadow Register (SSR).

The program memory contains the user-defined instruction flow and output sequence. The address control logic addresses the program memory. This control logic supports high-level instruction functions including conditional branches, subroutine calls and returns, loops, and multiway branches. The condition code selection logic selects the condition code input to be tested when a conditional instruction is executed. The polarity of the selected condition code input is controlled by the POL bit in the microword. The instruction decode generates the control signals necessary to perform the instruction specified by the instruction part (P[33 : 16]) of the microword. The SSR enables in-system testing to isolate problems down to the IC level.

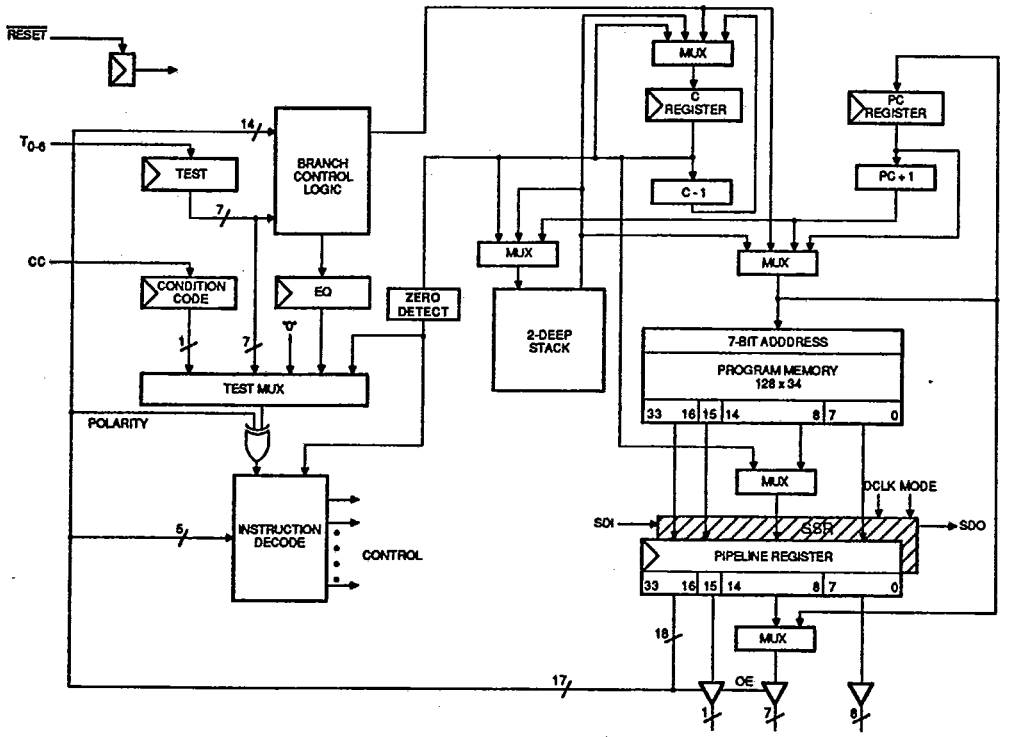
Program Memory

T-46-13-47

The FPC program memory is a 128-word by 34-bit PROM with a 34-bit pipeline register at its output. The upper 18 bits (P[33:16]) of the pipeline register are internal to the FPC and form the instruction to control address sequencing. The format for instructions is: a 1-bit synchronous Output Enable OE, a 5-bit OPCODE, a 1-bit test polarity select POL, a 4-bit TEST condition select field, and a 7-bit immediate DATA field. The DATA field is used to provide branch addresses, test input masks, and counter values.

The lower 16 bits (P[15:0]) of the pipeline register are brought out as user-defined, general purpose control outputs. The upper eight control outputs (P[15:8]) are three-stated when OE is programmed as a LOW. The lower eight control bits (P[7:0]) are always enabled.

Outputs P[14:8] will contain the next instruction address when the optional EXP fuse is programmed. The contents of the CREG can be sent to outputs P[14:8], regardless of whether the EXP fuse is programmed.



BD007691

Figure 1. Am29PL142 Detailed Block Diagram

Address Control Logic

The address control logic consists of four logic blocks:

PC GRP – the program counter multiplexer (PC MUX), program counter register (PC) and combinatorial incrementer (PC + 1)

STACK – subroutine multiplexer (S MUX) with a 2-word by 7-bit-wide stack

CNTR – count register (CREG) with counter multiplexer (C MUX), combinatorial decrementer (CREG – 1), and zero detect on count register

GOTO – multifunction branch control logic

PC GRP

The PC GRP consists of a 4:1 multiplexer, a program counter (PC) register, and a 7-bit combinatorial incrementer (PC + 1). It selects the PC, PC + 1, the branch address, or the top of stack as the next instruction address input to the program memory and the PC.

$\overline{\text{RESET}}$ is latched internally on the first rising edge of the clock after it goes LOW. During the first clock cycle, the PC MUX is set to all ones (address 127). On the next rising edge of the clock, the program memory contents at location 127 are loaded into the pipeline register.

STACK

The stack block consists of a 3:1 multiplexer (S MUX) that stores the data into the topmost location of the stack. The S MUX chooses from three sources: PC + 1, count register, and the top of the stack (for holding). PC + 1 is the input source when doing subroutine calls. PC MUX is the output destination when a return-from-subroutine instruction is performed.

CNTR

The CNTR block consists of a 4:1 multiplexer (C MUX) feeding a 7-bit count register (CREG) which, in turn, outputs to a combinatorial decrementer (CREG – 1) and a zero-detect circuit. The CNTR block is typically used for timing functions and iterative loop counting.

The C MUX has the following input sources: top of stack, the branch-logic output, CREG – 1, and the CREG (for holding).

GOTO

The GOTO (branch control) logic block performs three functions:

It provides a 7-bit value directly from the DATA field in the instruction pipeline register.

It provides a 7-bit branch address from the TEST inputs T[6:0] masked by the DATA field from the instruction pipeline register.

It compares the TEST inputs T[6:0] (masked by the DATA field in the instruction pipeline register) with the constant field comparison (see Instruction Format) in the pipeline register.

The EQ flip-flop is set according to the following expression:

EQ = [(TEST .AND. DATA) .XNOR. constant] .OR. EQ

The EQ flag can be tested by the condition-code selection logic. Multiple tests of any group of T inputs in a manner analogous to sum-of-products can be performed, since a no-match comparison does not reset the EQ flag. Any conditional branch on EQ will reset the EQ flag. Conditional returns on EQ will not change the EQ flag. $\overline{\text{RESET}}$ input LOW will reset the EQ flag.

NOTE: A zero in the DATA field blocks the corresponding bit in the TEST field; a one activates the corresponding bit.

The constant field bits that correspond to masked test field bits must be zero. A zero is substituted for masked test field bits. This masking operation is independent of the value of the POL bit.

Condition Code Selection Logic

The condition code selection logic consists of a 16:1 multiplexer. The 16 condition inputs are the seven test bits, the condition code input CC, the EQ flag, CREG ZERO status, and six UNCOND test conditions connected to zero for the unconditional mode. The TEST field in the pipeline register (P[26:23]) selects one of the 16 conditions. If one of the UNCOND is chosen, and the POL bit is a one, the instruction is executed with a "forced PASS" condition. If one of the UNCOND is chosen, and the POL bit is a zero, the instruction is executed with a "forced FAIL" condition. See opcode descriptions for more details.

The polarity bit POL in P[27] of the instruction pipeline register allows the user to test for either pass/true or fail/false condition. Refer to Table 2 for details.

Instruction Decode

The instruction decoder is a PLA that generates the control for 28 different instructions. The decoder inputs include the OPCODE field P[32:28], the zero detection flag from the CNTR, and the selected test condition code from the condition code select logic.

Operational Modes

The Am29PL142 operates as a 7-bit microcontroller in normal mode. By programming the EXP fuse, the output pins P[14:8] can be used as the program address for external registered PROMs. By using external registered PROMs in conjunction with the Am29PL142, control outputs wider than 16 bits can be obtained.

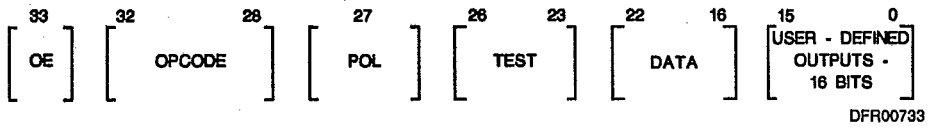
The SSR option provides on-chip diagnostic capabilities for in-system testing. This is accomplished by setting the appropriate conditions on the MODE control input. Note that this can only be done if the SSR diagnostics fuse has been programmed. See Table 1.

The SSR diagnostics configuration activates a 34-bit-wide, D-type register, called a "shadow" register, on the pipeline register inputs. The shadow register can be serially loaded from the SDI pin, parallel loaded from the pipeline register, or held. The pipeline register can be loaded from the program memory in normal operation or from the shadow register during diagnostics. A redefinition of four device pins is required to control the different diagnostics functions. CC also functions as the Serial Data Input (SDI), P[0] becomes the serial data output (SDO), P[7] becomes the diagnostic clock (DCLK), and P[6] becomes the diagnostic mode control (MODE). The various diagnostic and normal modes are shown in Table 1.

Serially loading a test instruction into the shadow register and parallel loading the shadow register contents into the pipeline register forces execution of the test instruction. The test result can then be clocked into the pipeline register, as in normal operation mode, parallel loaded into the shadow register, and serially shifted out for system diagnostics.

T-46-13-47

Am29PL142 General Instruction Format



DFR00733

WHERE:

OE = Synchronous Output Enable for P[15:8].

OPCODE = A 5-bit opcode field for selecting one of the 27 single data-field instructions.

POL = A 1-bit test condition polarity select.
0 = Test for true (HIGH) condition.
1 = Test for false (LOW) condition.

TEST = A 4-bit test condition select.

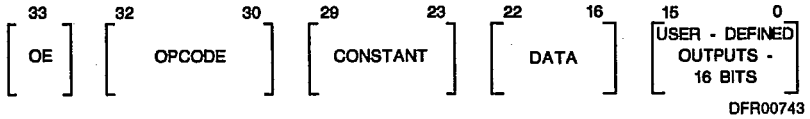
TEST[3:0]	UNDER TEST
0000	T[0]
0001	T[1]
0010	T[2]
0011	T[3]
0100	T[4]
0101	T[5]
0110	T[6]
0111	CC
1000	EQ
1001	CREG ZERO
1010-1111	UNCOND [0]

The polarity bit POL in an instruction allows the user to test for a pass/true or fail/false condition as shown in Table 2. An unconditional true is set by selecting UNCOND and POL = 1.

DATA = A 7-bit conditional branch address, test input mask, or counter value field designated as PL in instruction mnemonics.

The special two data field comparison instruction format is shown below:

Am29PL142 Comparison Instruction Format



DFR00743

WHERE:

OE = Synchronous Output Enable for P[15:8].

OPCODE = Compare instruction (binary 100).

CONSTANT = A 7-bit constant for equal to comparison with T*M.

DATA = A 7-bit mask field for masking the incoming T[6:0] inputs.

TABLE 1.

Inputs				Outputs			Operation
SDI	MODE	DCLK	CLK	SDO	Shadow Register	Pipeline Register	
D	L	↑	H,L,↓	S_o	$S_{i-1} \leftarrow S_i$ $S_{31} \leftarrow D$	Hold	Serial Right Shift Register
CC**	L	H,L,↓	↑	L	Hold	$P_i \leftarrow PROM_i$	Load Pipeline Register from PROM
L	H	↑	H,L,↓	L	$S_i \leftarrow P_i$	Hold	Load Shadow Register from Pipeline* Register
X	H	H,L,↓	↑	SDI	Hold	$P_i \leftarrow S_i$	Load Pipeline Register from Shadow Register
H	H	↑	H,L,↓	H	Hold	Hold	Hold Shadow Register

* S7, S6 are undefined. If P[33] in the microword is a one, S_{15}-S_8 are loaded from the pipeline register. If P[33] in the microword is a zero, $S_{15} - S_8$ are loaded from an external source.
** During normal operation, this pin behaves as the CC test input.

FUNCTION TABLE DEFINITIONS

INPUTS

H = HIGH
L = LOW
X = Don't Care
D = Serial Data
↑ = LOW-to-HIGH transition
↓ = HIGH-to-LOW transition

TABLE 2.

Input Condition Being Tested	POL	Condition
0	0	Fail
0	1	Pass
1	0	Pass
1	1	Fail

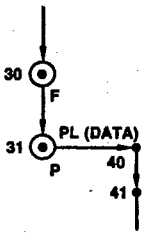
T-46-13-47

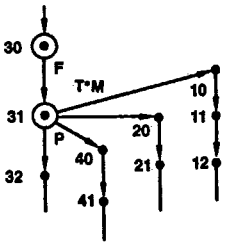
Am29PL142 INSTRUCTION SET DEFINITION

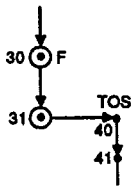
The stack consists of two registers; the top one is labeled Top of Stack (TOS), the bottom one is labeled Bottom of Stack (BOS).

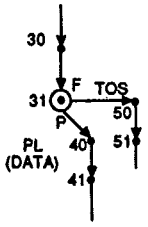
● = Other Instruction
⊙ = Instruction being described
O = Register in part
P = Test Pass
F = Test Fail
X, Y are arbitrary values in the CREG or STACK

Opcode	Mnemonic	Description	Execution Example	Register Transfer Description

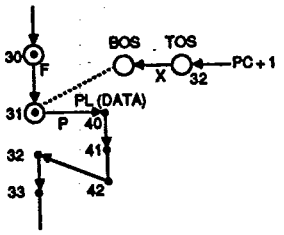
19	GOTOPL	**IF (cond) THEN GOTO PL (data)** Conditional branch to the address in the PL (DATA field). The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF001420	If (cond = true) Then
				PC ← PL(DATA)
				Else
				PC ← PC + 1

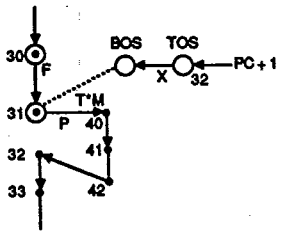
1F	GOTOTM	**IF (cond) THEN GOTO TM (data)** Conditional branch to the address defined by the T*M (T[6:0] under bitwise mask from the DATA field). This instruction is intended for multiway branches. The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF001441	If (cond = true) Then
				PC ← T*M
				Else
				PC ← PC + 1

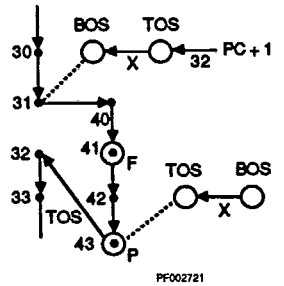
03	GOTOSTK	**IF (cond) THEN GOTO (STACK)** Conditional branch to the address at the top of the stack, or else continue. The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition if the test field is UNCOND and POL = 0.	 PF002680	If (cond = true) Then
				PC ← TOS
				Else
				PC ← PC + 1

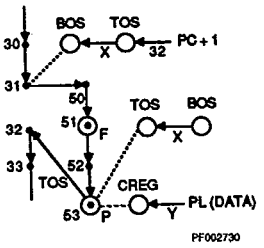
18	FORK	**IF (cond) THEN GOTO PL (data) ELSE GOTO (STACK)** Conditional branch to the address in the PL (DATA field) or the TOS. A branch to PL is taken if the condition is true and a branch to TOS if false. The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF002690	If (cond = true) Then
				PC ← PL(DATA)
				Else
				PC ← TOS

Opcode	Mnemonic	Description	Execution Example	Register Transfer Description

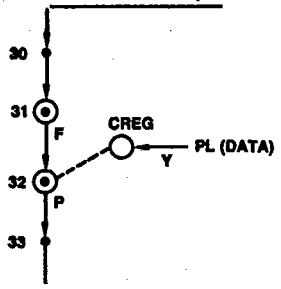
1C	CALPL	**IF (cond) THEN CALL PL (data)** Conditional jump to subroutine at the address in the PL (DATA field). The PC + 1 is pushed into the TOS as the return address. The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF002700	If (cond = true) Then
				BOS ← TOS
				TOS ← PC + 1
				PC ← PL(DATA)
				Else
				PC ← PC + 1
				T-46-13-47

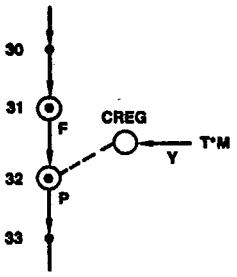
1E	CALTM	**IF (cond) THEN CALL TM (data)** Conditional jump to subroutine at the address specified by the T*M (T[6:0] under bitwise mask from the DATA field). The PC + 1 is pushed into the TOS as the return address. The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF002710	If (cond = true) Then
				BOS ← TOS
				TOS ← PC + 1
				PC ← T*M
				Else
				PC ← PC + 1

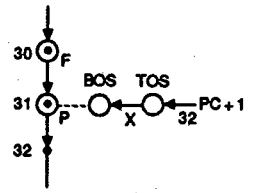
02	RET	**IF (cond) THEN RET** Conditional return from subroutine. The TOS provides the return from subroutine address and the stack is popped. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF002721	If (cond = true) Then
				PC ← TOS
				TOS ← BOS
				Else
				PC ← PC + 1

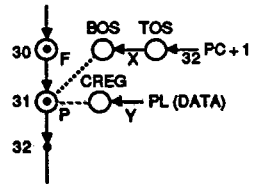
00	RETPL	**IF (cond) THEN RET, LOAD PL (data)** Conditional return from subroutine and load the CREG from the PL (DATA field). The TOS provides the return from subroutine address and the stack is popped. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF002730	If (cond = true) Then
				PC ← TOS
				TOS ← BOS
				CREG ← PL(data)
				Else
				PC ← PC + 1

Opcode	Mnemonic	Description	Execution Example	Register Transfer Description

04	LDPL	**IF (cond) THEN LOAD PL (data)** Conditional Load the CREG from the PL (DATA field). The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF001511	If (cond = true) Then
				CREG ← PL(DATA)
				PC ← PC + 1
				Else
				PC ← PC + 1

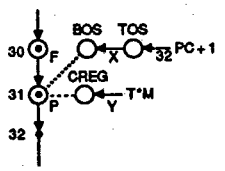
06	LDTM	**IF (cond) THEN LOAD TM (data)** Conditional load the CREG from the T*M (T[6:0] inputs under bitwise mask from the DATA field). The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF001521	If (cond = true) Then
				CREG ← T*M
				PC ← PC + 1
				Else
				PC ← PC + 1

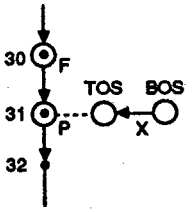
15	PSH	**IF (cond) THEN PUSH** Conditional push the PC + 1 into the TOS. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF0002740	If (cond = true) Then
				BOS ← TOS
				TOS ← PC + 1
				PC ← PC + 1
				Else
				PC ← PC + 1

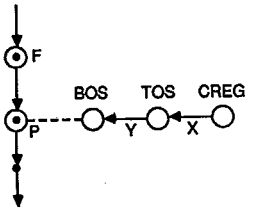
14	PSHPL	**IF (cond) THEN PUSH, LOAD PL (data)** Conditional push the PC + 1 into the TOS and load the CREG from the PL (DATA field). The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.	 PF002750	If (cond = true) Then
				BOS ← TOS
				CREG ← PL(DATA)
				TOS ← PC + 1
				PC ← PC + 1
				Else
				PC ← PC + 1

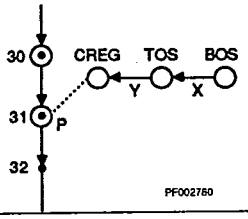
T-46-13-47

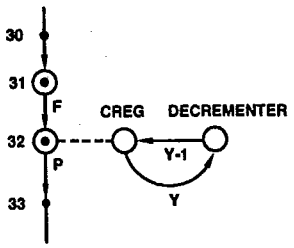
Opcode	Mnemonic	Description	Execution Example	Register Transfer Description

16	PSHTM	**IF (cond) THEN PUSH, LOAD TM (data)**
		Conditional push the PC + 1 into the TOS and load the CREG from the T*M (T[6:0] under bitwise mask from the DATA field). The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.
			
			PF002770
				If (cond = true) Then
				BOS ← TOS
				CREG ← T*M
				TOS ← PC + 1
				PC ← PC + 1
				Else
				PC ← PC + 1
				T-46-13-47

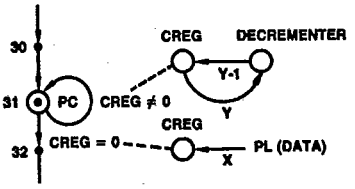
07	POP	**IF (cond) THEN POP**
		Conditional Pop the TOS. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.
			
			PF002760
				If (cond = true) Then
				TOS ← BOS
				PC ← PC + 1
				Else
				PC ← PC + 1

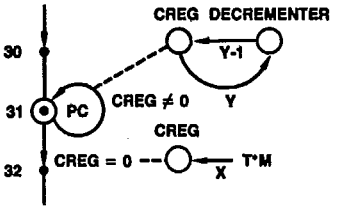
05	PSHCNTR	**IF (cond) THEN PUSH (CREG)**
		Conditional push CREG contents to top of stack. The instruction is executed with a forced PASS condition if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition if the test field is UNCOND and POL = 0.
			
			PF002790
				If (cond = true) Then
				BOS ← TOS
				TOS ← CREG
				PC ← PC + 1
				Else
				PC ← PC + 1

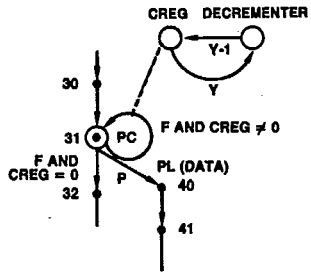
17	POPCNTR	**IF (cond) THEN POP TO (CREG)**
		Conditional pop TOS into CREG. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.
			
			PF002750
				If (cond = true) Then
				CREG ← TOS
				TOS ← BOS
				PC ← PC + 1
				Else
				PC ← PC + 1

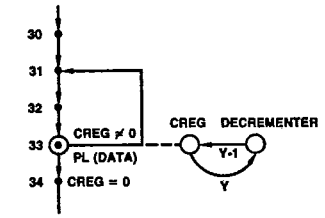
0B	DEC	**IF (cond) THEN DEC**
		Conditional decrement of the CREG. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.
			
			PF001621
				If (cond = true) Then
				CREG ← CREG - 1
				PC ← PC + 1
				Else
				PC ← PC + 1

Opcode	Mnemonic	Description	Execution Example	Register Transfer Description

0C	DECPL	**WHILE (CREG <> 0) WAIT ELSE LOAD PL (data)**
		Conditional Hold until the counter is equal to zero, then load CREG from the PL (DATA field). This instruction is intended for timing waveform generation. If the CREG is not equal to zero, the same instruction is refetched while CREG is decremented. Timing is complete when the CREG is equal to zero, causing the next instruction to be fetched and the CREG to be reloaded from PL. This instruction does not depend on the pass/fail condition.
			
			PF001632
				While (CREG ≠ 0)
				CREG ← CREG – 1
				PC ← PC
				End While
				CREG ← PL(DATA)
				PC ← PC + 1

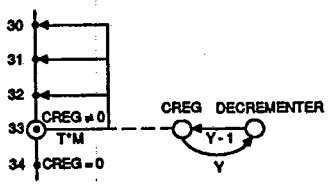
0E	DECTM	**WHILE (CREG <> 0) WAIT ELSE LOAD TM (data)**
		Conditional Hold until the counter is equal to zero, then load CREG from the T*M (T[6:0] under bitwise mask from the DATA field). This instruction is intended for timing waveform generation. If the CREG is not equal to zero, the same instruction is refetched while the CREG is decremented. Timing is complete when the CREG is equal to zero, causing the next instruction to be refetched and the CREG to be reloaded from T*M. This instruction does not depend on the pass/fail condition.
			
			PF001641
				While (CREG ≠ 0)
				CREG ← CREG – 1
				PC ← PC
				End While
				CREG ← T*M
				PC ← PC + 1

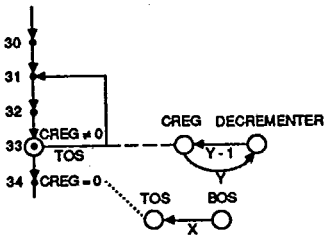
1D	DECGOPL	**IF (cond) THEN GOTO PL (data) ELSE WHILE (CREG <> 0) WAIT**
		Conditional Hold/Count. The current instruction will be refetched and the CREG decremented until the condition under test becomes true or the counter is equal to zero. If the condition becomes true, a branch to the address in the PL (DATA field) is executed. If the counter becomes zero without the condition becoming true, a CONTINUE is executed. The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition if the TEST field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition if the TEST field is UNCOND and POL = 0.
			
			PF001651
				While (cond = false)
				If (CREG ≠ 0)
				CREG ← CREG – 1
				PC ← PC
				Else
				PC ← PC + 1
				End While
				PC ← PL(DATA)

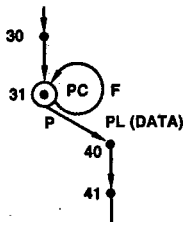
08	LPPL	**WHILE (CREG <> 0) LOOP TO PL (data)**
		Conditional loop to the address in the PL (DATA field). This instruction is intended to be placed at the bottom of an iterative loop. If the CREG is not equal to zero, it is decremented (signifying completion of an iteration), and a branch to the PL address (top of the loop) is executed. If the CREG is equal to zero, looping is complete and the next sequential instruction is executed. This instruction does not depend on the pass/fail condition. The EQ flag will be reset if the test field selects it and CREG is not equal to zero.
			
			PF001671
				While (CREG ≠ 0)
				CREG ← CREG – 1
				PC ← PL (DATA)
				End While
				PC ← PC + 1

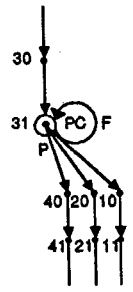
T-46-13-47

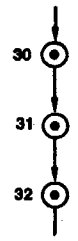
Opcode	Mnemonic	Description	Execution Example	Register Transfer Description

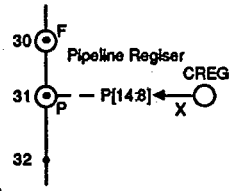
0A	LPTM	**WHILE (CREG <> 0) LOOP TO TM (data)**
		Conditional loop to the address T*M (T[6:0] under bitwise mask from the DATA field). This instruction should be placed at the bottom of an iterative loop. If CREG is not equal to zero, it is decremented (signifying completion of an iteration), and a branch to the T*M address (top of the loop) is executed. If CREG is equal to zero, looping is complete and the next sequential instruction is executed. This does not depend on the pass/fail condition. The EQ flag will be reset if the test field selects it and CREG is not equal to zero.
			
			PF002800
				While (CREG ≠ 0)
				CREG ← CREG – 1
				PC = T*M
				End While
				PC ← PC + 1
				
				T-46-13-47

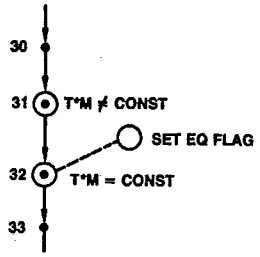
0F	LPSTK	**WHILE (CREG <> 0) LOOP TO (STACK)**
		Conditional loop to the address in the TOS. If CREG ≠ 0, the CREG is decremented and a branch to the TOS address is executed. If the CREG = 0, looping is complete, the stack is popped, and the next sequential instruction is executed. This instruction does not depend on the pass/fail condition. The EQ flag will be reset if the test field selects it and CREG is not equal to zero.
			
			PF002810
				While (CREG ≠ 0)
				CREG ← CREG – 1
				PC ← TOS
				End While
				TOS ← BOS
				PC ← PC + 1

1A	WAITPL	**IF (cond) THEN GOTO PL (data) ELSE WAIT**
		Conditional Hold. The current instruction will be refetched and executed until the condition under test becomes true. When true, a branch to the address in the PL (DATA field) is executed. The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.
			
			PF001660
				If (cond = true) Then
				PC ← PL(DATA)
				Else
				PC ← PC

1B	WAITTM	**IF (cond) THEN GOTO TM (data), ELSE WAIT**
		Conditional Hold. The current instruction will be refetched and executed until the condition under test becomes true. When the condition is true, a branch to the T*M address (T[6:0] under bitwise mask from the DATA field) is executed. The EQ flag will be reset if the test field selects it and the condition passes. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.
			
			PF002660
				If (cond = true) Then
				PC ← T*M
				Else
				PC ← PC

Opcode	Mnemonic	Description	Execution Example	Register Transfer Description
0D	CONT	**CONTINUE** The next sequential instruction is fetched unconditionally. This instruction can also be used to reset the EQ flag by selecting EQ in the TEST field.	 PF001690	PC ← PC + 1 T-46-13-47

01	OUTPUT	**IF (cond) THEN OUTPUT**
		The CREG contents will be output on pins P[14:8] during the next clock cycle. Care should be taken to ensure that the outputs are enabled for the next sequential instruction by setting the microcode bit OE = 1. The instruction is executed with a forced PASS condition, if the test field is UNCOND and POL = 1. The instruction is executed with a forced FAIL condition, if the test field is UNCOND and POL = 0.
			
			If<160>
			PF002620
				If (cond = true) Then
				P[14 : 8] ← CREG
				PC ← PC + 1
				Else
				PC ← PC + 1

10 – 13 (100XX binary)	CMP	**CMP TM (data) TO PL (data)** This instruction performs bitwise exclusive-OR of T*M (T[6:0] under bitwise mask from the DATA field) with CONSTANT (P[29:23]). If T*M equals CONSTANT, the EQ flag is set to one, so that a branch may occur in a following instruction. If not equal, the EQ flag is unaffected. This allows sequences of compares, in a manner analogous to sum-of-products, to be performed which can be followed by a single conditional branch if one or more of the comparisons were true. Note: The EQ flag is set to zero on reset or when EQ is selected as the test condition in a branch. Conditional returns on EQ leave the flag unchanged. **Constant field bits that correspond to masked test-input bits must be zero.** This instruction does not depend on the pass/fail condition.	 PF001701	Compare T*M and PL(DATA) EQ = ((T [6:0] .AND. DATA) .XNOR. CONSTANT) .OR. EQ PC ← PC + 1

MICROINSTRUCTIONS BASED ON TEST CONDITIONS

T-46-13-47

Opcode	Mnemonic	Assembler Statement	Condition Pass				Condition Fail				Notes
			PC MUX	STACK	CREG	EQ FLAG	PC MUX	STACK	CREG	EQ FLAG	
00	RETPL	IF (cond) THEN RET, LOAD PL (data)	TOS	Pop	Load PL	NC	PC + 1	Hold	Hold	NC	5
01	OUTPUT	IF (cond) THEN OUTPUT	PC + 1	Hold	Hold	NC	PC + 1	Hold	Hold	NC	1
02	RET	IF (cond) THEN RET	TOS	Pop	Hold	NC	PC + 1	Hold	Hold	NC	5
03	GOTOSTK	IF (cond) THEN GOTO (STACK)	TOS	Hold	Hold	Reset	PC + 1	Hold	Hold	NC	3
04	LDPL	IF (cond) THEN LOAD PL (data)	PC + 1	Hold	Load PL	NC	PC + 1	Hold	Hold	NC	
05	PSHCNTR	IF (cond) THEN PUSH (CREG)	PC + 1	Push CREG	Hold	NC	PC + 1	Hold	Hold	NC	6
06	LDTM	IF (cond) THEN LOAD TM (data)	PC + 1	Hold	Load TM	NC	PC + 1	Hold	Hold	NC	
07	POP	IF (cond) THEN POP	PC + 1	Pop	Hold	NC	PC + 1	Hold	Hold	NC	5
0B	DEC	IF (cond) THEN DEC	PC + 1	Hold	DEC	NC	PC + 1	Hold	Hold	NC	
14	PSHPL	IF (cond) THEN PUSH, LOAD PL (data)	PC + 1	Push PC + 1	Load PL	NC	PC + 1	Hold	Hold	NC	6
15	PSH	IF (cond) THEN PUSH	PC + 1	Push PC + 1	Hold	NC	PC + 1	Hold	Hold	NC	6
16	PSHTM	IF (cond) THEN PUSH, LOAD TM (data)	PC + 1	Push PC + 1	Load TM	NC	PC + 1	Hold	Hold	NC	6
17	POPCNTR	IF (cond) THEN POP TO (CREG)	PC + 1	Pop	Load TOS	NC	PC + 1	Hold	Hold	NC	5
18	FORK	IF (cond) THEN GOTO PL (data) ELSE GOTO (STACK)	PL	Hold	Hold	Reset	TOS	Hold	Hold	NC	3
19	GOTOPL	IF (cond) THEN GOTO PL (data)	PL	Hold	Hold	Reset	PC + 1	Hold	Hold	NC	3
1A	WAITPL	IF (cond) THEN GOTO PL (data) ELSE WAIT	PL	Hold	Hold	Reset	PC	Hold	Hold	NC	3
1B	WAITTM	IF (cond) THEN GOTO TM (data) ELSE WAIT	TM	Hold	Hold	Reset	PC	Hold	Hold	NC	3
1C	CALPL	IF (cond) THEN CALL PL (data)	PL	Push PC + 1	Hold	Reset	PC + 1	Hold	Hold	NC	3, 6
1E	CALTM	IF (cond) THEN CALL TM (data)	TM	Push PC + 1	Hold	Reset	PC + 1	Hold	Hold	NC	3, 6
1F	GOTOTM	IF (cond) THEN GOTO TM (data)	TM	Hold	Hold	Reset	PC + 1	Hold	Hold	NC	3

Key: TOS = Top of Stack
PC = Program Counter
CREG = Counter Register
PL = Pipeline (DATA) Field
TM = Test Inputs Marked by PL (DATA) Field
DEC = Decrement
BOS = Bottom of Stack
NC = No Change

Notes: See notes on next page.

MICROINSTRUCTION DEPENDENT ON CREG

Opcode	Mnemonic	Assembler Statement	CREQ = 0				CREG ≠ 0				Notes
			PC MUX	STACK	CREG	EQ FLAG	PC MUX	STACK	CREG	EQ FLAG	
08	LPPL	WHILE (CREG<>0) LOOP TO PL (data)	PC + 1	Hold	Hold	NC	PL	Hold	DEC	Reset	4
0A	LPTM	WHILE (CREG<>0) LOOP TO TM (data)	PC + 1	Hold	Hold	NC	TM	Hold	DEC	Reset	4
0C	DECPL	WHILE (CREG<>0) WAIT ELSE LOAD PL (data)	PC + 1	Hold	Load PL	NC	PC	Hold	DEC	NC	
0E	DECTM	WHILE (CREG<>0) WAIT ELSE LOAD TM (data)	PC + 1	Hold	Load TM	NC	PC	Hold	DEC	NC	
0F	LPSTK	WHILE (CREG<>0) LOOP TO (STACK)	PC + 1	Hold	Pop	NC	TOS	Hold	DEC	Reset	4

MICROINSTRUCTION DEPENDENT ON TEST CONDITION AND CREG VALUE

Opcode	Mnemonic	Assembler Statement	CREG Content	Condition Pass				Condition Fail				Notes
				PC MUX	STACK	CREG	EQ FLAG	PC MUX	STACK	CREG	EQ FLAG	
1D	DECGOPL	IF (cond) THEN GOTO PL (data) ELSE WHILE (CREG<>0) WAIT	≠0	PL	Hold	Hold	Reset	PC	Hold	DEC	NC	3
			= 0	PL	Hold	Hold	Reset	PC + 1	Hold	Hold	NC	3

UNCONDITIONAL MICROINSTRUCTIONS

Opcode	Mnemonic	Assembler Statement	PC MUX	STACK	CREG	EQ FLAG	Notes
0D	CONT	CONTINUE	PC + 1	Hold	Hold	NC	2
10-13 (100XX) Binary	CMP	CMP TM (data) TO PL (data)	PC + 1	Hold	Hold	Set	7

Key: TOS = Top of Stack
BOS = Bottom of Stack
PC = Program Counter
CREG = Counter Register
PL = Pipeline (DATA) Field
TM = Test Inputs Marked by PL (DATA) Field
DEC = Decrement
NC = No Change

Notes: 1. If Condition Passes, Output CREG contents on next clock cycle.
2. If Condition = EQ, reset EQ flag.
3. If Condition = EQ and Condition Passes, reset EQ flag.
4. If Condition = EQ and CREG ≠ 0, reset EQ flag.
5. When Stack is popped, BOS is transferred to TOS.
6. When Stack is pushed, TOS is transferred to BOS before value is written into TOS.
7. Set EQ Flag if CONST field = T*M.

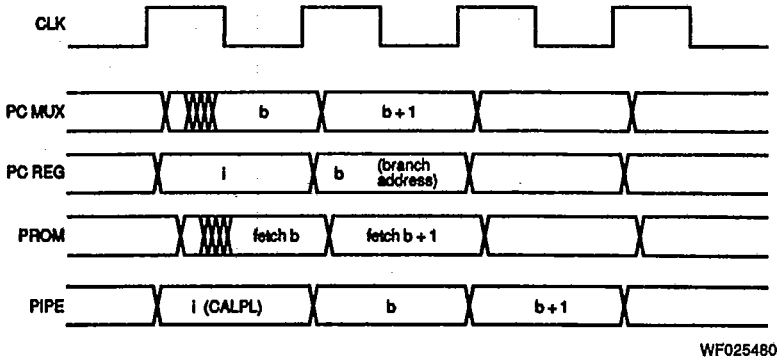


Figure 2. Conditional Branch/Jump Instruction (Example: CALPL)

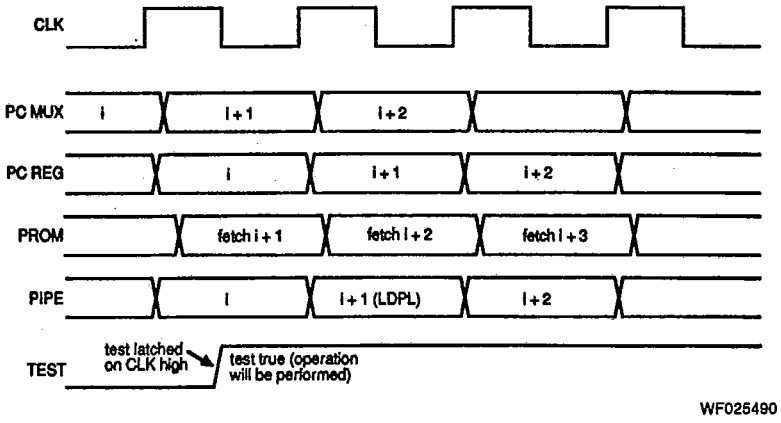


Figure 3. Conditional Instruction (Example: LDPL)

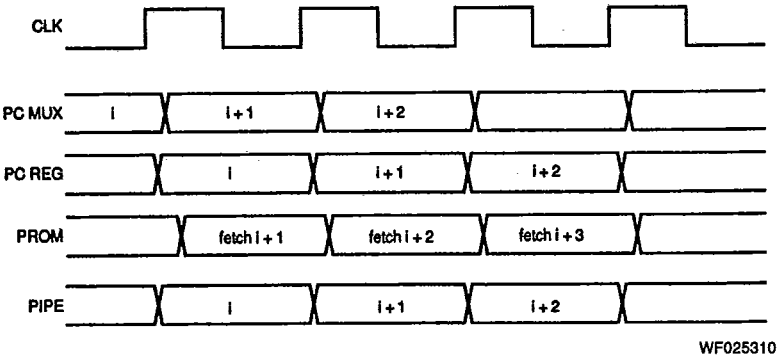


Figure 4. Unconditional Instruction (Example: CONT)

T-46-13-47

Using The Am29PL142 To Address External Registered PROM

When the EXP fuse is programmed, the program memory MUX is output over pins P[14:8]/A[6:0]. This feature can be used to extend the width of the output control word when external registered PROMs are used. In Figure 5 below, the Am29PL142 addresses external registered PROMs to provide an output control word (9 + N) bits wide (where N is the bit width of the PROMs).

When the OUTPUT instruction is executed, the CREG contents are output over pins P[14:8]/A[6:0] on the following cycle. Consequently, if the CREG contents must be read after programming the EXP fuse, the system design should be modified to handle this exception.

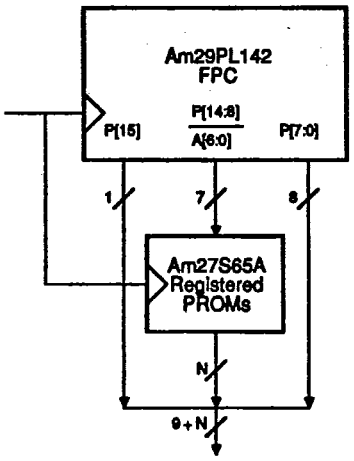


Figure 5. Using Am29PL142 to Address External Registered PROMs

PROGRAMMING

The Am29PL142 FPC is programmed and verified using a simple algorithm that is almost identical to that used for AMD's Programmable Array Logic family. The internal programmable array of the Am29PL142 is organized as a 128-word by 34-bit PROM. The fuse to be programmed is selected by its address (1 of 128), the byte at that address (1 of 5), and the bit in the byte (1 of 8). Control of programming and verifying is accomplished by applying a simple sequence of voltages on two control pins (CLK and CC).

The fuse address is selected using a full decode of the T[6 : 0] inputs, where T[6] is the MSB and T[0] the LSB. The 1-of-5 byte addressing is done on the P[7] (MSB), and P[6] and P[5] (LSB) outputs. The bit selection is done one output at a time by applying the programming voltage (V_{OP}) to the respective output pin P[15:8]. A graphic representation of the fuse array organization for programming, with fuse numbering compatible to the JEDEC standard programmable logic transfer format, is shown in Figure 6.

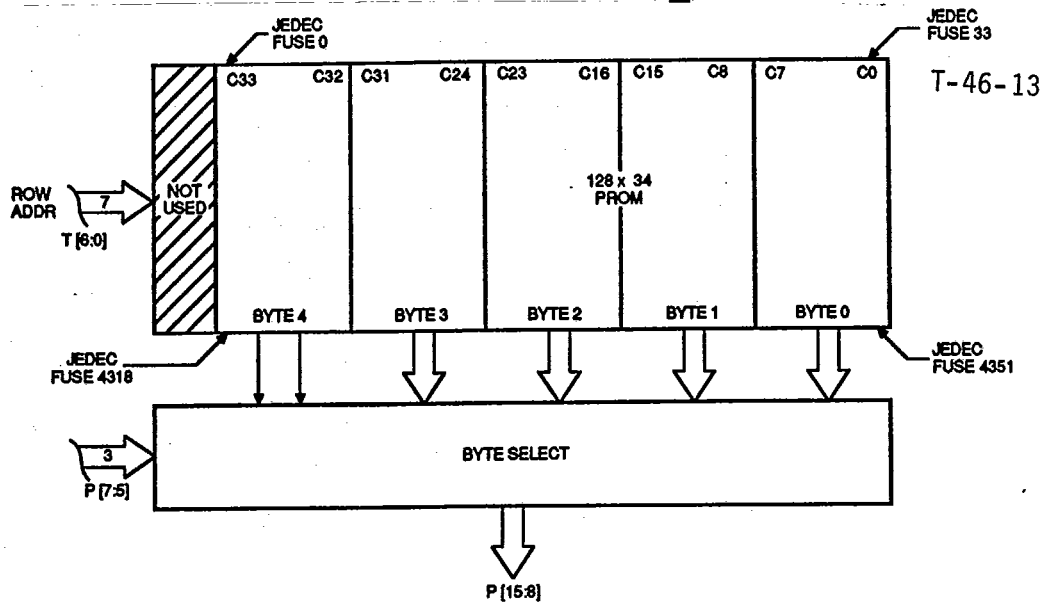
The complete program and verify cycle timing is shown in the programming waveform. A programming sequence is initiated by raising the CLK pin to V_{HH}. This places the device in the program mode and disables the output pins so that they may be used as fuse addressing inputs. The next step is to address the fuse to be blown as previously stated. Note that bit selection, with V_{OP}, should follow address and byte selection. Raising the CC pin to V_{HH} initiates programming and lowering V_{OP} terminates programming. Lowering the CLK pin to a TTL LOW level places the device in the fuse verification mode by enabling the programming outputs, P[15:8]. Following a clock pulse the fuse may be verified on the same output as bit selection was performed. Using this scheme, fuses can be verified in parallel as a byte if desired. The verification mode is terminated by lowering the CC pin back to a normal TTL level.

Programming Yield

AMD programmable logic devices have been designed to ensure extremely high programming yields (> 98%). To help ensure that a part was correctly programmed (once the programming sequence is completed) the entire fuse array should be reverified at both low and high V_{CC} (V_{CCL} and V_{CCH}). Reverification can be accomplished in a verification-only mode (CC at V_{HH}) by reading the outputs in parallel. This verification cycle checks that the array fuses have been blown correctly and can be sensed under varying conditions by the outputs.

AMD programmable logic devices contain extra fuses and circuitry so that it can be verified, before shipping, that all PROM fuses can be programmed. Additional circuitry is included for pre-shipment testing of the logic portion of the device. These added features assure high programming yields and correct logic operation.

T-46-13-47



BD007551

JEDEC Fuse Numbering: If Byte ⩽ 3, Fuse No. = 34 x (Row_Addr) + 8 x (3-Byte) + (7-Bit) + 2
If Byte = 4, Fuse No. = 34 x (Row_Addr) + (1-Bit)

Note: SSR fuse is JEDEC Fuse Number 4352,
EXP fuse is JEDEC Fuse Number 4353

Example Computations:

Row 0, Byte 4, Bit 1	Fuse No. = 34 (0) + (1 – 1) = 0
Row 127, Byte 0, Bit 0	Fuse No. = 34 (127) + 8 (3 – 0) + (7 – 0) + 2 = 4351

	Byte Select			Bit Select							
Byte	P[7]	P[6]	P[5]	P[15]	P[14]	P[13]	P[12]	P[11]	P[10]	P[9]	P[8]
0	1	1	0	C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	1	C15	C14	C13	C12	C11	C10	C9	C8
2	1	0	0	C23	C22	C21	C20	C19	C18	C17	C16
3	1	0	1	C31	C30	C29	C28	C27	C26	C25	C24
4	0	1	0	–	–	–	–	–	–	C33	C32
5	0	1	1	–	–	–	–	–	–	–	–
6	0	0	1	–	–	–	–	–	–	–	–
7	0	0	0	SSR	EXP	–	–	–	–	–	–

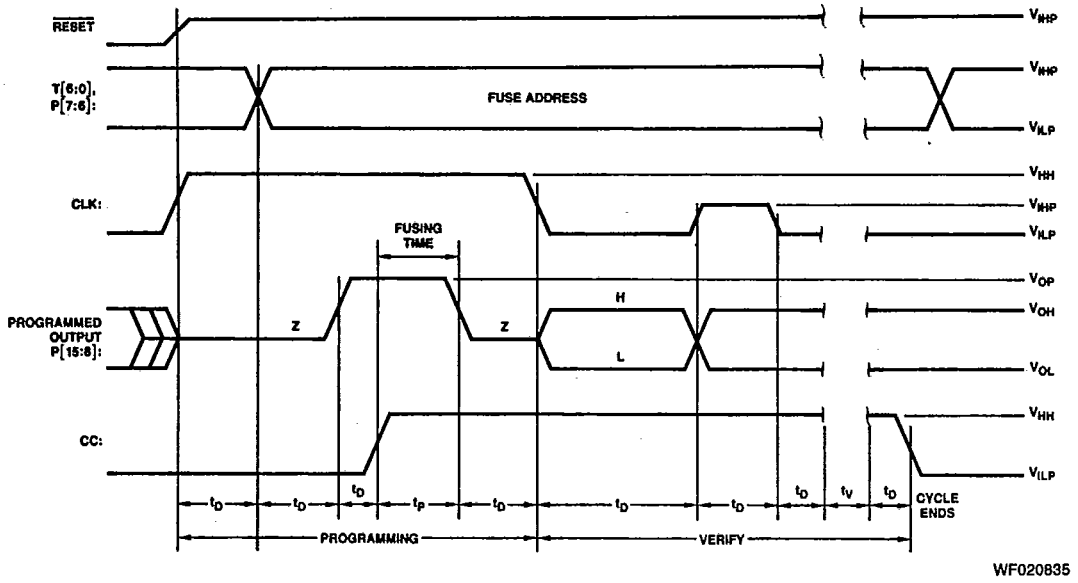
Notes: 1. Locations with hyphens (–) are reserved. Contents of those locations are unknown and should not be overwritten.
2. T[6:0] gives the row address for programming.
3. C33 is MSB of the Word, C0 is the LSB.
4. Unblown fuses are interpreted as HIGH (1), blown fuses are LOW (0).

Figure 6. Programming Configuration

PROG

T-46-13-47

Parameter Symbol	Parameter Description		Min.	Typ.	Max	Unit
V_{HH}	Control Pin Extra High Level	CC @ 5–10 mA	15.5	16	16.5	V
		CLK@ 5–10 mA	15.5	16	16.5	
V_{OP}	Program Voltage, P [15:8] @ 15–200 mA		19.5	20	20.5	V
V_{IHP}	Input High Level During Programming and Verify		2.4	5	5.5	V
V_{ILP}	Input Low Level During Programming and Verify		0.0	0.3	0.5	V
V_{CCP}	V_{CC} During Programming @ I_{CC} = 500 mA		5	5.2	5.5	V
V_{CCL}	V_{CC} During First Pass Verification @ I_{CC} = 500 mA		4.5	4.7	5	V
V_{CCH}	V_{CC} During Second Pass Verification @ I_{CC} = 545 mA		5.4	5.7	6.0	V
V_{Blown}	Successful Blown Fuse Sense Level @ Output			0.3	0.5	V
dV_{OP}/dt	Rate of Output Voltage Change		20		250	V/μs
dV_{FE}/dt	Rate of Fusing Enable Voltage Change (CC Rising Edge)		100		1000	V/μs
t_P	Fusing Time First Attempt		40	50	100	μs
	Subsequent Attempts		4	5	10	ms
t_D	Delays Between Various Level Changes		100	200		ns
t_V	Period During which Output is Sensed for V_{Blown} Level		500			ns
V_{ONP}	Pull-Up Voltage on Outputs Not Being Programmed		V_{CCP} – 0.3	V_{CCP}	V_{CCP} + 0.3	V
R	Pull- Up Resistor on Outputs Not Being Programmed		1.9	2	2.1	kΩ
dV_{CLK}/dt	Rate of CLK Voltage Change (Programming)		40	50	250	V/μs
	Rate of CLK Voltage Change (Verify)		250	300	1000	V/μs



WF020835

Programming Waveforms

ABSOLUTE MAXIMUM RATINGS

Storage Temperature ..... −65 to +150°C
(Ambient) Temperature Under Bias ..... −55 to +125°C
Supply Voltage to Ground Potential
(Pin 28 to Pin 14) Continuous ..... −0.5 V to +7.0 V
DC Voltage Applied to Outputs
(Except During Programming) ..... −0.5 V to +V_{CC} Max.
DC Voltage Applied to Outputs
During Programming ..... 21 V
DC Output Current, Into Outputs During
Programming (Max Duration of 1 s) ..... 200 mA
DC Input Voltage ..... −0.5 V to +5.5 V
DC Input Current ..... −30 mA to +5.0 mA

Stresses above those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

OPERATING RANGES

Commercial (C) Devices
Ambient Temperature (T_A) ..... 0 to +70°C
Supply Voltage (V_{CC}) ..... +4.75 to +5.25 V

Military* (M) Devices
Case Temperature (T_C) ..... −55 to +125°C
Supply Voltage (V_{CC}) ..... +4.5 V to +5.5 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

*Military Product 100% tested at T_C = +25°C, +125°C and −55°C.

Typical θ_{JA} ..... 42°C/W
Typical θ_{JC} ..... 3°C/W

DC CHARACTERISTICS over operating range unless otherwise specified (for APL Products, Group A, Subgroups 1, 2, 3 are tested unless otherwise noted)

Parameter Symbol	Parameter Description	Test Conditions				Min.	Max.	Unit
V_{OH}	Output HIGH Voltage	V_{CC} = Min., V_{IN} = V_{IH} or V_{IL}	COM'L	29PL142	I_{OH} = −3.0 mA	2.4		V
			MIL	29PL142	I_{OH} = −1.0 mA			
V_{OL}	Output LOW Voltage	V_{CC} = Min., V_{IN} = V_{IL} or V_{IH}	COM'L	29PL142	I_{OL} = 16 mA		0.50	V
			MIL	29PL142	I_{OL} = 12 mA			
V_{IH} (Note 1)	Input HIGH Level	Guaranteed Input Logical HIGH Voltage for All Inputs				2.0		V
V_{IL} (Note 1)	Input LOW Level	Guaranteed Input Logical LOW Voltage for All Inputs					0.8	V
I_{IL}	Input LOW Current	V_{CC} = Max. V_{IN} = 0.5 V	CLK P [15:6] All Other Inputs				−1.5 −0.55 −0.50	mA
I_{IH}	Input HIGH Current	V_{CC} = Max. V_{IN} = 2.4 V	CLK P [15:6] All Other Inputs				150 100 25	μA
I_I	Input HIGH Current	V_{CC} = Max., V_{IN} = 5.5 V					1.0	mA
I_{SC}	Output Short-Circuit Current	V_{CC} = Max + 0.5 V V_{OUT} = 0.5 V (Note 2)				−20	−80	mA
I_{CC}	Power Supply Current	V_{CC} = Max.	COM'L	29PL142	T_A = 0 to +70°C		500	mA
					T_A = +70°C		450	
			MIL	29PL142	T_C = −55 to +125°C		570	
					T_C = +125°C		475	
V_I	Input Clamp Voltage	V_{CC} = Min, I_{IN} = −18 mA					−1.2	V
I_{OZH}	Output Leakage Current (Note 3)	V_{CC} = Max., V_{IL} = 0.8 V V_{IH} = 2.0 V			V_O = 2.4 V		100	μA
I_{OZL}					V_O = 0.5 V		−550	

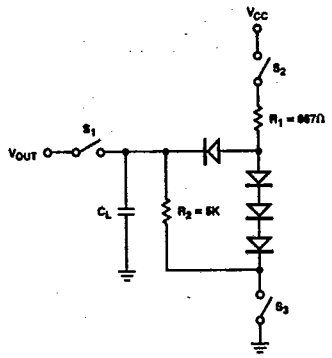
Notes: 1. These are absolute values with respect to device ground and all overshoots due to system or tester noise are included.
2. Not more than one output should be tested at a time. Duration of the short-circuit test should not be more than one second. V_{OUT} = 0.5 V has been chosen to avoid test problems caused by tester ground degradation.
3. I/O pin leakage is the worst-case of I_{OZX} or I_{IX} (where X = H or L).

SWITCHING CHARACTERISTICS over operating range unless otherwise specified (for APL Products, Group A, Subgroups 9, 10, 11 are tested unless otherwise noted)

No.	Parameter Symbol	Parameter Description	Test Conditions	COM'L Min.	COM'L Max.	MIL Min.	MIL Max.	Unit
1	t_{PD}	CLK to P[15:0]	See Test Output Load Conditions		20		25	ns
2		CLK to A[6:0]			50		55	ns
3		DCLK to SDO			30		30	ns
4		Mode to SDO			30		30	ns
5		SDI to SDO			30		30	ns
6	t_S	T[6:0] to CLK (Note 1)		15		15		ns
7		CC to CLK (Note 1)		15		15		ns
8		$\overline{\text{RESET}}$ to CLK		15		15		ns
9		Mode to CLK		30		30		ns
10		Mode to DCLK		30		30		ns
11		SDI to DCLK		30		30		ns
12		P[15:8] to DCLK		30		30		ns
13	t_H	T[6:0] to CLK		0		0		ns
14		CC to CLK		0		0		ns
15		$\overline{\text{RESET}}$ to CLK		4		0		ns
16		Mode to CLK		0		0		ns
17		Mode to DCLK		0		0		ns
18		SDI to DCLK		0		0		ns
19		P[15:8] to DCLK		0		0		ns
20	t_{PZX}	CLK to P[15:8] Enable			20		25	ns
21	t_{PXZ}	CLK to P[15:8] Disable			30		30	ns
22	t_{PWH}	CLK Pulse Width (HIGH)		15		15		ns
23	t_{PWL}	CLK Pulse Width (LOW)		25		30		ns
24	t_{PW}	DCLK Pulse Width (HIGH and LOW)		30		30		ns
25	t_P	CLK Period (Notes 1 & 2)		50		65		ns

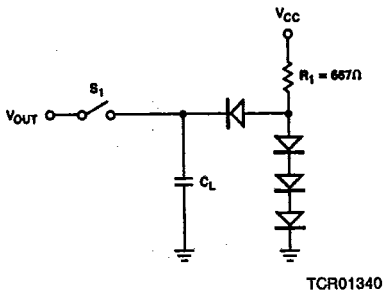
Notes: 1. These parameters are measured indirectly on unprogrammed devices. They are determined as follows:
a. Measure delay from input (T[6:0], or CLK) to PROM address out in test mode. This will measure the delay through the sequence logic.
b. Measure setup time from T[6:0] input through PROM test columns to pipeline register in verify test column mode. This will measure the delay through the PROM and register setup.
c. Measure delay from T[6:0] input to PROM address out in verify test column mode. This will measure the delay through the logic and P[15:0] outputs.
To calculate the desired parameter measurement the following formula is used:
Measurement (a) + Measurement (b) – Measurement (c)
CLK PERIOD:
CLK (a) + (b) – (c) = CLK PERIOD
2. CLK rise time must not exceed 7 ns.

SWITCHING TEST CIRCUITS



TCR01330

A. Three State Outputs

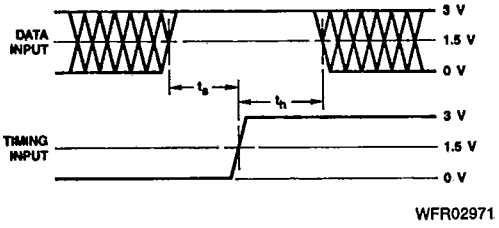


TCR01340

B. Normal Outputs

Notes: 1. $C_L = 50$ pF includes scope probe, wiring and stray capacitances without device in test fixture.
2. S_1, S_2, and S_3 are closed during function tests and all AC tests except output enable tests.
3. S_1 and S_3 are closed while S_2 is open for t_{PZH} test.
4. $C_L = 5.0$ pF for output disable tests.

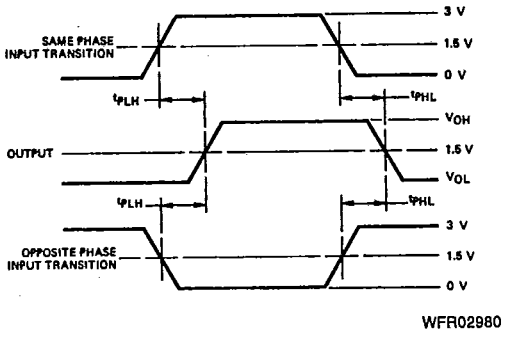
SWITCHING TEST WAVEFORMS



WFR02971

Setup, Hold, and Release Times

Notes: 1. Diagram shown for HIGH data only. Output transition may be opposite sense.
2. Cross hatched area is don't care condition.

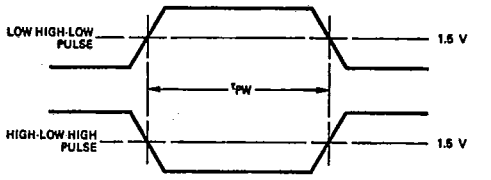


WFR02980

Propagation Delay

SWITCHING TEST WAVEFORMS (Cont'd.)

T-46-13-47



WFR02791

Pulse Width

Test	V_X	Output Waveform – Measurement Level
All t_{PD}s	5.0V	V_{OH}, V_{OL}, 1.5V
t_{PHZ}	0.0V	V_{OH}, 0.5V, 0.0V
t_{PLZ}	5.0V	3V, 0.5V, V_{OL}
t_{PZH}	0.0V	V_{OH}, 1.5V, 0.0V
t_{PZL}	5.0V	3V, 1.5V, V_{OL}

WFR02680

Enable and Disable Times

Notes: 1. Diagram shown for input Control Enable-LOW and input Control Disable-HIGH.
2. S_1, S_2, and S_3 of Load Circuit are closed except where shown.

Note: Pulse generator for all pulses: Rate $\leq$ 1.0 MHz; $Z_0 = 50\ \Omega$; $t_r \leq 2.5$ ns.

Test Philosophy and Methods

The following points give the general philosophy that we apply to tests that must be properly engineered if they are to be implemented in an automatic environment. The specifics of what philosophies applied to which test are shown.

1. Ensure the part is adequately decoupled at the test head. Large changes in supply current when the device switches may cause function failures due to V_{CC} changes.
2. Do not leave inputs floating during any tests, as they may oscillate at high frequency.
3. Do not attempt to perform threshold tests at high speed. Following an input transition, ground current may change by as much as 600 mA in 5–8 ns. Inductance in the ground cable may allow the ground pin at the device to rise by hundreds of millivolts momentarily. Current level may vary from product to product.
4. Use extreme care in defining input levels for AC tests. Many inputs may be changed at once, so there will be significant noise at the device pins that may not actually reach V_{IL} or V_{IH} until the noise has settled. AMD recommends using $V_{IL} \leq 0$ V and $V_{IH} \geq 3.0$ V for AC tests.
5. To simplify failure analysis, programs should be designed to perform DC, Function, and AC tests as three distinct groups of tests.
6. Capacitive Loading for AC Testing

   Automatic testers and their associated hardware have stray capacitance that varies from one type of tester to another but is generally around 50 pF. This, of course, makes it impossible to make direct measurements of parameters that call for smaller capacitive load than the associated stray capacitance. Typical examples of this are the so-called "float delays," which measure the propagation delays into the high impedance state and are usually specified at a load capacitance of 5.0 pF. In these cases, the test is performed at the higher load capacitance (typically 50 pF) and engineering correlations based on data taken with a bench set up are used to predict the result at the lower capacitance.

   Similarly, a product may be specified at more than one capacitive load. Since the typical automatic tester is not capable of switching loads in mid-test, it is impossible to make measurements at both capacitances even though they may both be greater than the stray capacitance. In these cases, a measurement is made at one of the two capacitances. The result at the other capacitance is predicted from engineering correlations based on data taken with a bench set up and the knowledge that certain DC measurements (I_{OH}, I_{OL}, for example) have already been taken and are within spec. In some cases, special DC tests are performed in order to facilitate this correlation.
7. Threshold Testing T-46-13-47

   The noise associated with automatic testing (due to the long inductive cables), and the high gain of the tested device when in the vicinity of the actual device threshold, frequently give rise to oscillations when testing high-speed speed circuits. These oscillations are not indicative of a reject device, but instead, of an overtaxed test system. To minimize this problem, thresholds are tested at least once for each input pin. Thereafter, "hard" high and low levels are used for other tests. Generally this means that function and AC testing are performed at "hard" input levels rather than at V_{IL} Max. and V_{IH} Min.
8. AC Testing

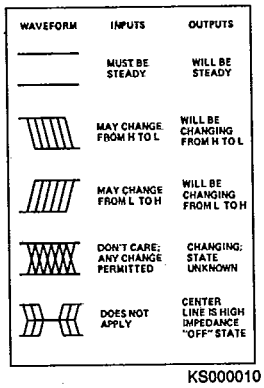
   Occasionally, parameters are specified that cannot be measured directly on automatic testers because of tester limitations. Data input hold times often fall into this category. In these cases, the parameter in question is guaranteed by correlating these tests with other AC tests that have been performed. These correlations are arrived at by the cognizant engineer by using data from precise bench measurements in conjunction with the knowledge that certain DC parameters have already been measured and are within spec.

   In some cases, certain AC tests are redundant since they can be shown to be predicted by other tests that have already been performed. In these cases, the redundant tests are not performed.
9. Output Short-Circuit Current Testing

   When performing I_{OS} tests on devices containing RAM or registers, great care must be taken that undershoot caused by grounding the high-state output does not trigger parasitic elements which in turn cause the device to change state. To avoid this effect, it is common to make the measurement at a voltage (V_{output}) that is slightly above ground. The V_{CC} is raised by the same amount so that the result (as confirmed by Ohm's law and precise bench testing) is identical to the $V_{OUT} = 0$, V_{CC} = Max. case.

SWITCHING WAVEFORMS

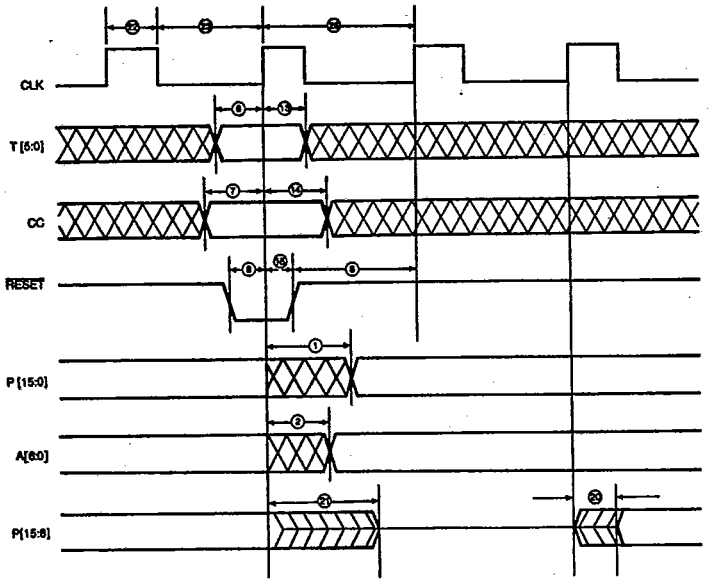
KEY TO SWITCHING WAVEFORMS



KS000010

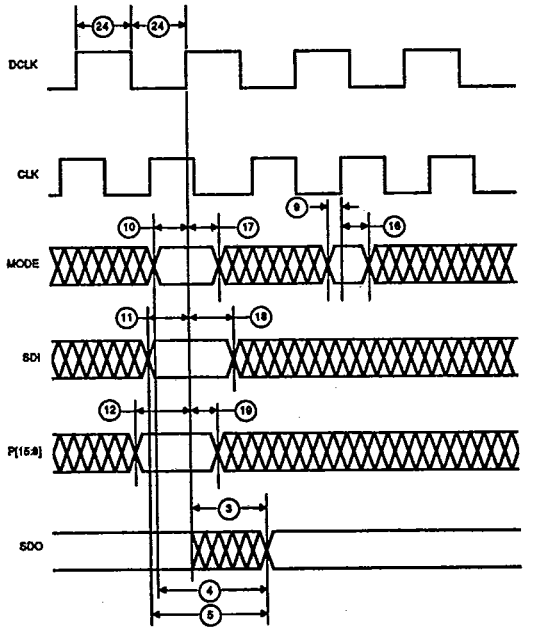
SWITCHING WAVEFORMS (Cont'd.)

T-46-13-47



WF020855

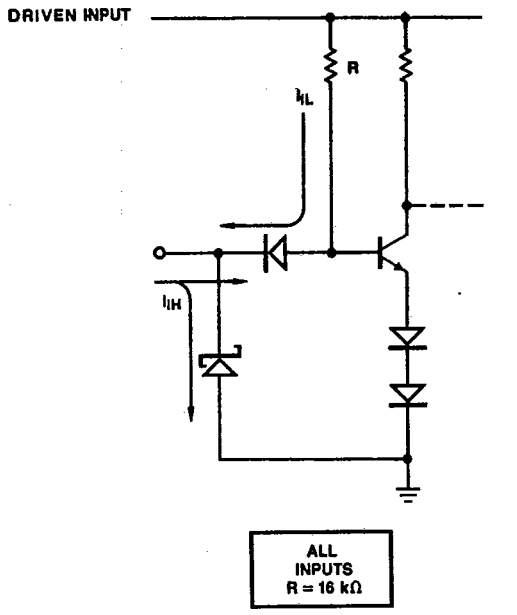
Normal Configuration



WF025501

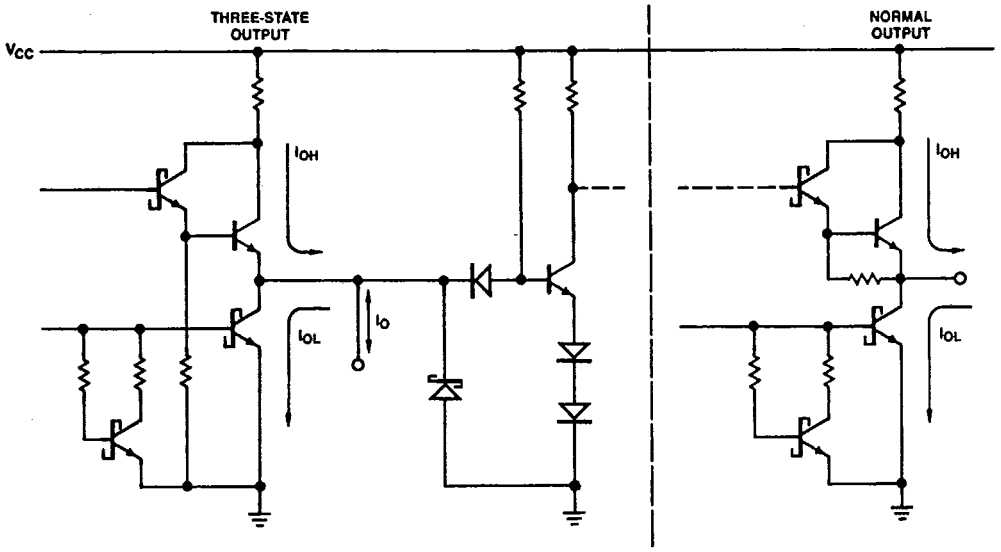
SSR Configuration

INPUT/OUTPUT CIRCUIT DIAGRAMS



ICR00536

$C_O \cong 5.0$ pF, all inputs



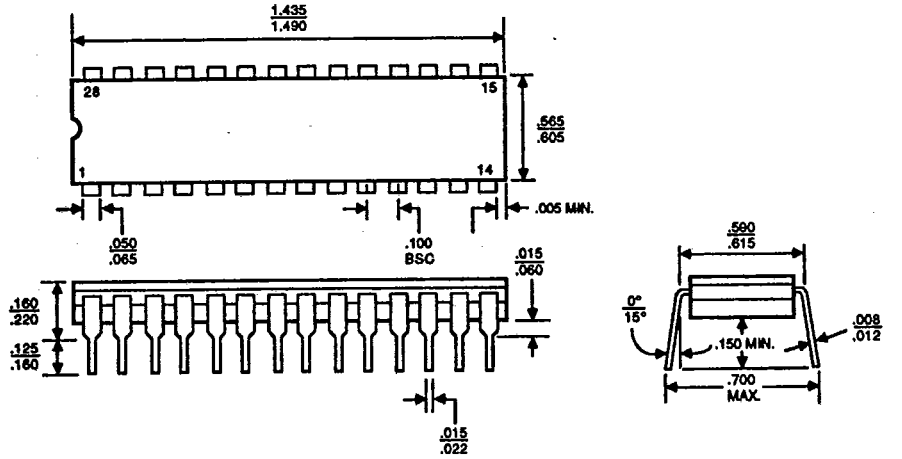
ICR00524

$C_O \cong 5.0$ pF, all outputs

NOTE: Actual current flow direction shown.

PHYSICAL DIMENSIONS*

CD 028



PID# 06837B

*For reference only.

Advanced Micro Devices reserves the right to make changes in its product without notice in order to improve design or performance characteristics. The performance characteristics listed in this document are guaranteed by specific tests, correlated testing, guard banding, design and other practices common to the industry. For specific testing details, contact your local AMD sales representative. The company assumes no responsibility for the use of any circuits described herein.

ADVANCED MICRO DEVICES 901 Thompson Pl., P.O. Box 3453, Sunnyvale, CA 94088, USA
TEL: (408) 732-2400 • TWX: 910-339-9280 • TELEX: 34-6306 • TOLL FREE: (800) 538-8450

© 1988 Advanced Micro Devices, Inc.
Printed in U.S.A. AIS-CP-15M-03/88-0