UNIT-6
Programmable Logic Devices (PLDs)

Programmable Logic Devices (PLDs)

- Programmable Logic Devices (PLDs) are ICs with a large number of gates and flip flops that can be configured with basic software to perform a specific logic function or perform the logic for a complex circuit.
- The term "programmable" means changing either hardware or software configuration of an internal logic and interconnects.
- The configuration of the internal logic is done by the user.
- PROM, EPROM, PAL, GAL etc. are examples of Programmable Logic Devices.

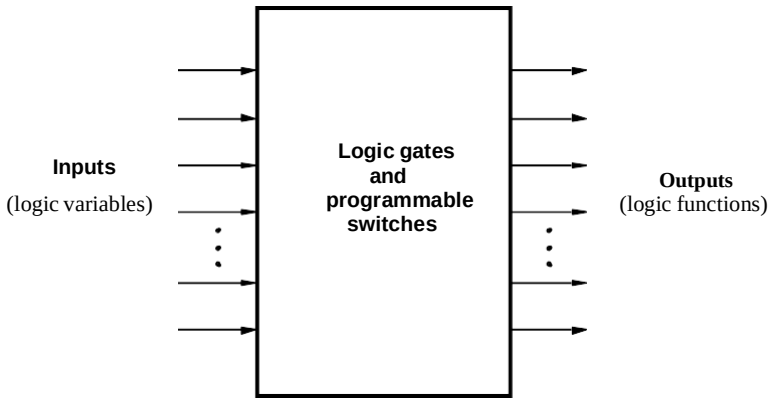
Major types of PLDs

- **SPLD:** (Simple PLDs) are the earliest type of array logic used for fixed functions and smaller circuits with a limited number of gates. (The PAL and PLA are both SPLDs).
- **CPLD:** (Complex PLDs) are multiple SPLDs arrays and inter-connection arrays on a single chip.
- **FPLD:** (Field Programmable Gate Array) are a more flexible arrangement than CPLDs, with much larger capacity.

Advantages to PLDs

- Reduced complexity of circuit boards
  - Lower power requirements
  - Less board space
  - Simpler testing procedures
- Higher reliability
- Design flexibility

- PLD as a Black Box



Programmable Logic Design

- A PLD is a general-purpose chip for implementing logic circuitry. It contains a collection of logic circuit elements that can be customized in different ways.
- A PLD can be viewed as a “black box” that contains logic gates and programmable switches.
- The programmable switches allow the logic gate inside the PLD to be connected together to implement whatever logic circuit is needed.

Programmable Logic Array (PLA):

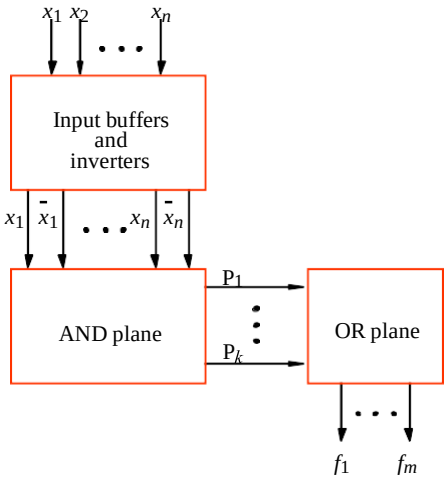
- The first developed was the programmable logic array (PLA*).*
- *The general structure of a PLA is* depicted in Figure
- Based on the idea that logic functions can be realized in sum-of-products form, a PLA comprises a collection of AND gates that feeds a set of OR gates

Programmable Logic Array (PLA)

– Use to implement circuits in SOP form

– The connections in the AND plane are programmable

– The connections in the OR plane are programmable



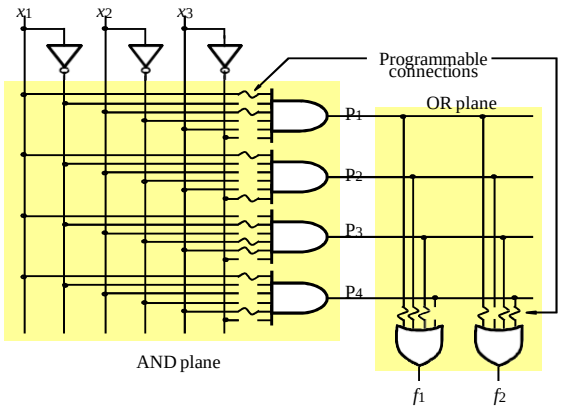
Programmable Logic Array (PLA):

- As shown in the figure, the PLA's inputs *x1, . . . , xn* pass through a set of buffers (which provide both the true value and complement of each input) into a circuit block called an *AND plane, or AND array. The AND plane produces a set of product terms P1, . . . , Pk . Each of* these terms can be configured to implement any AND function of *x1, . . . , xn.*
- The product terms serve as the inputs to an *OR plane, which produces the* outputs *f1, . . . , fm. Each output can be configured to realize any sum of P1, . . . , Pk and hence any sum-of-products function of the PLA inputs.*

- Gate Level Version of PLA

$f_1 = x_1x_2+x_1x_3'+x_1'x_2'x_3$

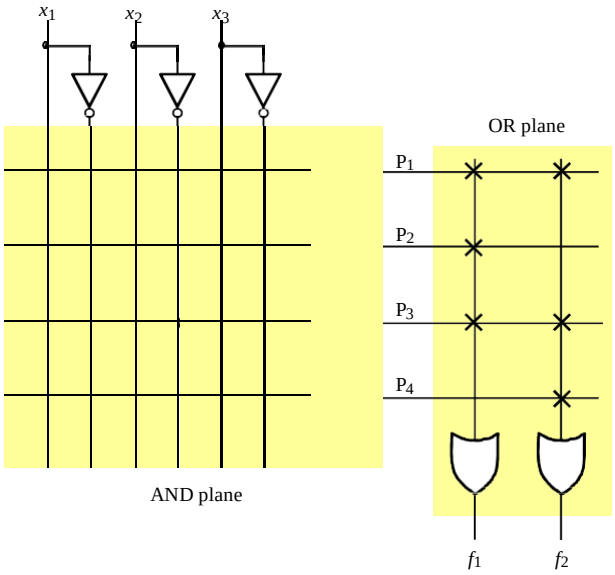
$f_2 = x_1x_2+x_1'x_2'x_3+x_1x_3$



- Customary Schematic of a PLA

$f_1 = x_1x_2+x_1x_3'+x_1'x_2'x_3$

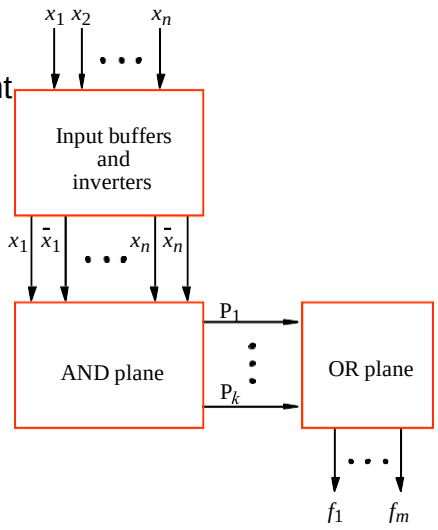
$f_2 = x_1x_2+x_1'x_2'x_3+x_1x_3$



x marks the connections left in place after programming

Programmable Array Logic (PAL)

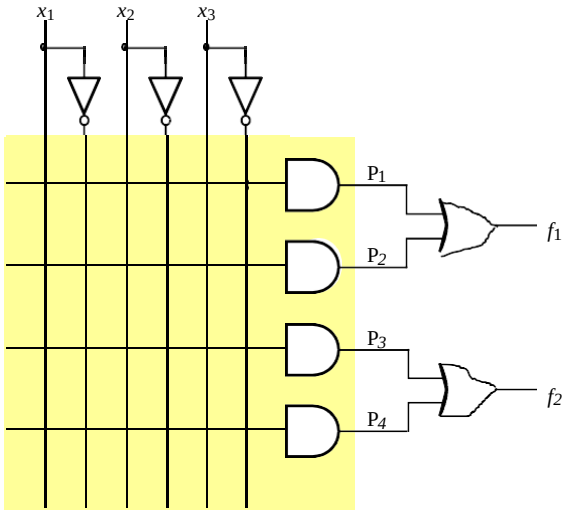
- Also used to implement circuits in SOP form
- The connections in the AND plane are programmable
- The connections in the OR plane are NOT programmable



- Example Schematic of a PAL

$f_1 = x_1x_2x_3' + x_1'x_2x_3$

$f_2 = x_1'x_2' + x_1x_2x_3$



AND plane

Example Schematic of a PAL

- An example of a PAL with three inputs, four product terms, and two outputs is given in Figure below. The product terms *P1 and P2 are* hardwired to one OR gate, and *P3 and P4 are hardwired to the other OR* gate.
- The PAL is shown programmed to realize the two logic functions f1 fand *f2.*
- *The PAL offers less flexibility.*
- *The* PLA allows up to four product terms per OR gate, where as the OR gates in the PAL have only two inputs. To compensate for the reduced flexibility, PALs are manufactured in a range of sizes, with various numbers of inputs and outputs, and different numbers of inputs to the OR gates.

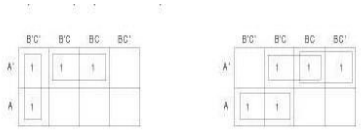
- Comparing PALs and PLAs
  - PALs have the same limitations as PLAs (small number of allowed AND terms) plus they have a fixed OR plane → less flexibility than PLAs
  - PALs are simpler to manufacture, cheaper, and faster (better performance)
  - PALs also often have extra circuitry connected to the output of each OR gate
    - The OR gate plus this circuitry is called a *macrocell*

Example

- Implement combinational logic circuits using PLA, consider the following expression

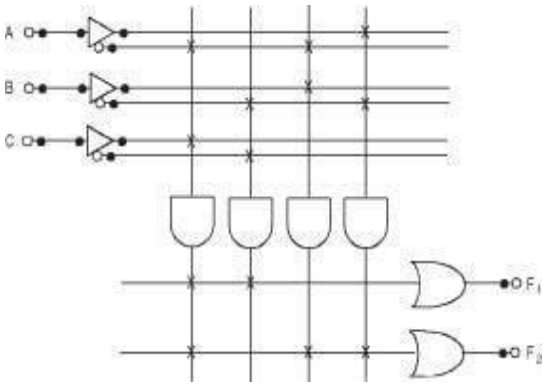
  F1 (A, B, C) = ∑m(0, 1, 3, 4) and

  F2 (A, B, C) = ∑m(1, 2, 3, 4, 5)



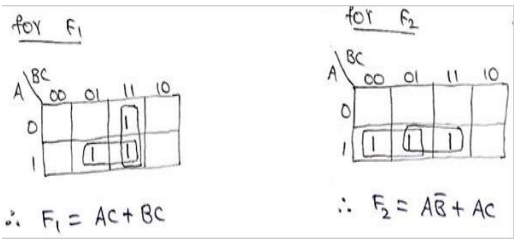
- The simplified Boolean expressions for the functions are
- F_1 = B'C' + A'C and
- F_2 = A'B + A'C + AB'.

Example(PLA)

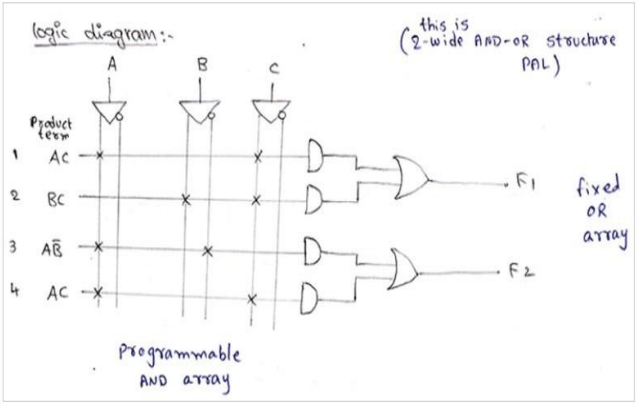


Example(PAL)

- **Implement the following Boolean expression using PAL,**
  F1= ∑ m (3,5,7) and F2 = ∑ m (4,5,7).

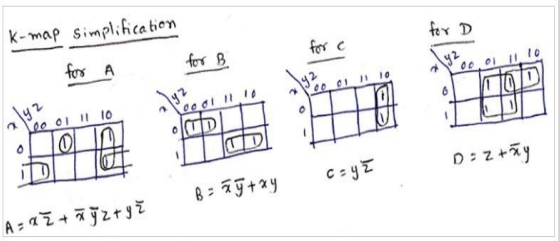


Example

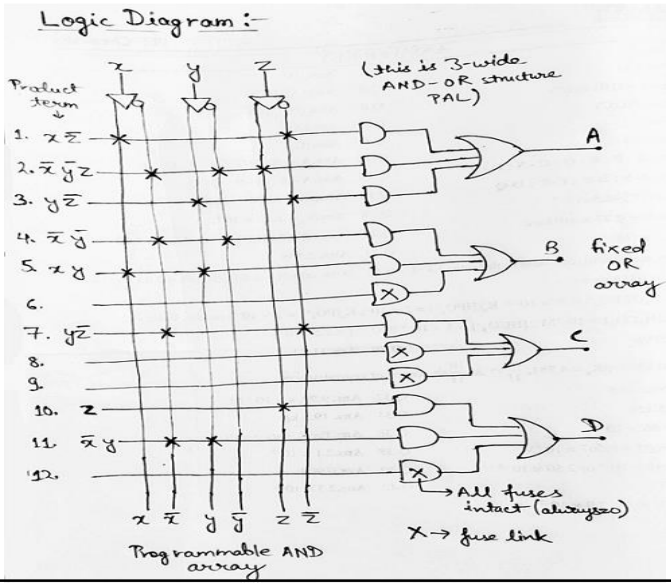


Example

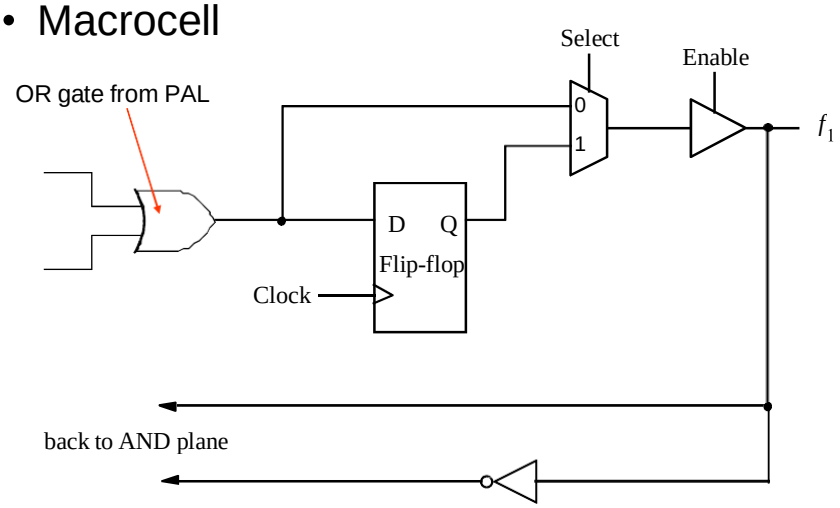
- **Implement the following Boolean expressions using a suitable PLA.**
  A (x,y,z) = ∑ m (1,2,4,6)
  B (x,y,z) = ∑ m (0,1,6,7)
  C (x,y,z) = ∑ m (2,6)
  D (x,y,z) = ∑ m (1,2,3,5,7)



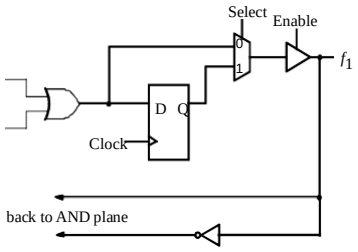
Example



- Macrocell



- Macrocell Functions
  - Enable = 0 can be used to allow the output pin for f_1 to be used as an additional input pin to the PAL
  - Enable = 1, Select = 0 is normal for typical PAL operation
  - Enable = Select = 1 allows the PAL to synchronize the output changes with a clock pulse



  - The feedback to the AND plane provides for multi-level design

Complex Programmable Logic Devices

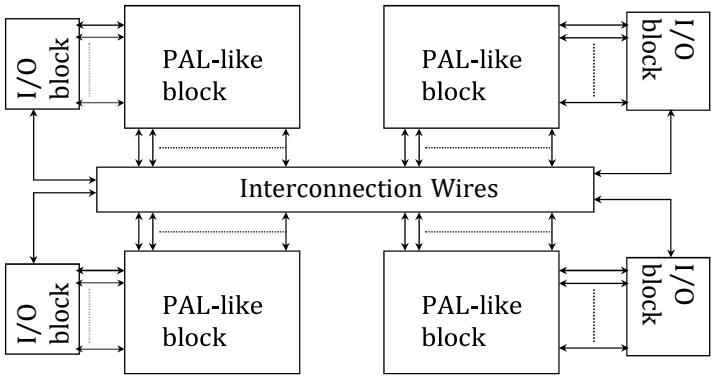
- Complex PLDs typically combine PAL combinational logic with FFs
  - Organized into logic blocks
  - Fixed OR array size
  - Combinational or registered output
  - Some pins are inputs only
- Usually enough logic for simple counters, state machines, decoders, etc.

Complex Programmable Logic Devices

- Chips containing PLDs are limited to modest sizes, typically supporting number of input and output more than 32.
- To accommodate circuits that require more input and outputs, either multiple PLAs or PALs can be used or a more sophisticated type of chip, called a complex programmable logic device (CLPD).
- CLPD is made up of multiple circuit blocks on a single chip, with internal wiring to connect the circuit blocks.

- The structure of CLPD is shown on the next slide. It includes four PAL-like blocks connected by interconnection wires.
- Each block in turn is connected to a sub-circuit I/O block, which is attached to a number of input and output pins.

CPLDS

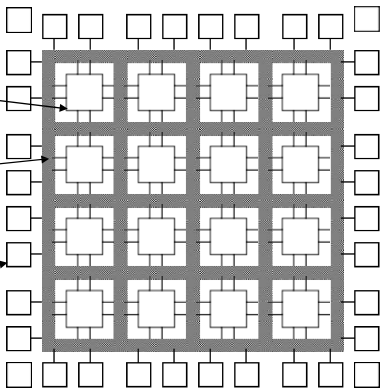


FPGA-Field Programmable Gate Array

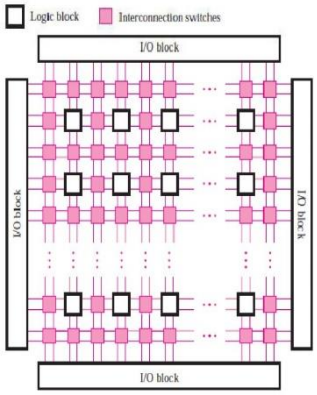
- SPLDs and CPLDs are relatively small and useful for simple logic devices
  - Up to about 20000 gates
- Field Programmable Gate Arrays (FPGA) can handle larger circuits
  - No AND/OR planes
  - Provide logic blocks, I/O blocks, and interconnection wires and switches
  - Logic blocks provide functionality
  - Interconnection switches allow logic blocks to be connected to each other and to the I/O pins

Field-Programmable Gate Arrays structure

- Logic blocks
  - To implement combinational and sequential logic
- Interconnect
  - Wires to connect inputs and outputs to logic blocks
- I/O blocks
  - Special logic blocks at periphery of device for external connections



General structure of FPGA



- The logic blocks are arranged in a two-dimensional array, and the interconnection wires are organized as horizontal and vertical *routing channels between rows and columns of logic blocks.*
- *The routing channels* contain wires and programmable switches that allow the logic blocks to be interconnected in many ways. Figure shows two locations for programmable switches; the pink boxes adjacent to logic blocks hold switches that connect the logic block input and output terminals to the interconnection wires, and the pink boxes that are diagonally between logic blocks connect one interconnection wire to another (such as a vertical wire to a horizontal wire).
- Programmable connections also exist between the I/O blocks and the interconnection wires. The actual number of programmable switches and wires in an FPGA varies in commercially available chips. FPGAs can be used to implement logic circuits of more than a million equivalent gates in size.

Logic Block:

- Each logic block in an FPGA typically has a small number of inputs and outputs. A variety of FPGA products are on the market, featuring different types of logic blocks. The most commonly used logic block is a lookup table (LUT), which contains storage cells that are used to implement a small logic function. Each cell is capable of holding a single logic value, either 0 or 1. The stored value is producedas the output of the storage cell. LUTs of various *sizes may be created, where the size is defined by the* number of inputs.
- The logic cell architecture varies between different device families. Each logic cell combines a few binary inputs (typically between 3 and 10) to one or two outputs according to a boolean logic function specified in the user program .The cell's combinatorial logic may be physically implemented as a small look-up table memory (LUT) or as a set of multiplexers and gates.
- LUT devices tend to be a bit more flexible and provide more inputs per cell than multiplexer cells at the expense of propagation delay.

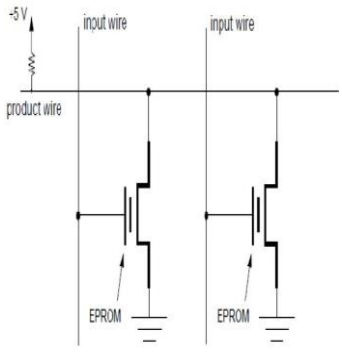
Switching Technology:

- Fuse
- Antifuse
- EEPROM or EPROM transistor
- SRAM

Switching Technology:

- The first type of user-programmable switch developed was the fuse used in PLAs. Although fuses are still used in some smaller devices, we will not discuss them here because they are quickly being replaced by newer technology.
- For higher density devices, where CMOS dominates the IC industry, different approaches to implementing programmable switches have been developed.
- For CPLDs the main switch technologies (in commercial products) are floating gate transistors like those used in EPROM and EEPROM, and for FPGAs they are SRAM and antifuse.

EEPROM or EPROM Programmable Switches

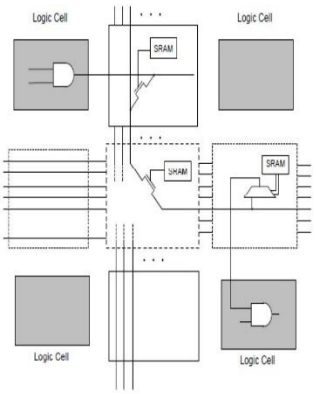


•An EEPROM or EPROM transistor is used as a programmable switch for CPLDs (and also for many SPLDs) by placing the transistor between two wires in a way that facilitates implementation of wired-AND functions.

•EPROM transistors as they might be connected in an AND-plane of a CPLD. An input to the AND-plane can drive a product wire to logic level '0' through an EPROM transistor, if that input is part of the corresponding product term. For inputs that are not involved for a product term, the appropriate EPROM transistors are programmed to be permanently turned off.

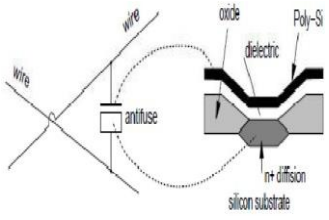
•A diagram for an EEPROM based device would look similar.

SRAM-controlled Programmable Switches



•An example of usage of SRAM-controlled switches is illustrated in Figure , showing two applications of SRAM cells: for controlling the gate nodes of pass-transistor switches and to control the select lines of multiplexers that drive logic block inputs.
•The figure gives an example of the connection of one logic block (represented by the AND-gate in the upper left corner) to another through two pass-transistor switches, and then a multiplexer, all controlled by SRAM cells.
•Whether an FPGA uses pass transistors or multiplexers or both depends on the particular product.

Actel Antifuse swich Structure



•Antifuses are originally open-circuits and take on low resistance only when programmed.
•Antifuses are suitable for FPGAs because they can be built using modified CMOS technology.
•As an example, Actel's antifuse structure,known as PLICE is depicted in Figure .
•The figure shows that an antifuse is positioned between two interconnect wires and physically consists of three sandwiched layers: the top and bottom layers are conductors, and the middle layer is an insulator.
•When unprogrammed, the insulator isolates the top and bottom layers, but when programmed the insulator changes to become a low-resistance link.
• PLICE uses Poly-Si and n+ diffusion as conductors and ONO as an insulator, but other antifuses rely on metal for conductors, with amorphous silicon as the middle layer

Summary of Programming Technologies:

Name	Re-programmable	Volatile	Technology
Fuse	no	no	Bipolar
EPROM	yes out of circuit	no	UVCMOS
EEPROM	yes in circuit	no	EECMOS
SRAM	yes in circuit	yes	CMOS
Antifuse	no	no	CMOS+

Comparison between CPLD and FPGA

	CPLD	FPGA
Logic Elements	Up to 500	Up to 250,000
Operating voltage	5V	3.3V external, 1.5V internal
Cost	Start at $10	Start at $15
Packaging	Some leaded	All SMT
Program retention	Yes	No – load at power up