**Programmable Logic Devices**

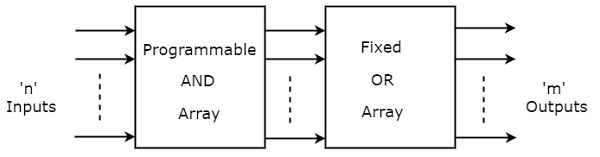
Programmable Logic Devices ***PLDs*** are the integrated circuits. They contain an array of AND gates & another array of OR gates. There are three kinds of PLDs based on the type of array*s*, which has programmable feature.

* Programmable Read Only Memory
* Programmable Array Logic
* Programmable Logic Array

The process of entering the information into these devices is known as **programming**. Basically, users can program these devices or ICs electrically in order to implement the Boolean functions based on the requirement. Here, the term programming refers to hardware programming.

**Programmable Array Logic PAL**

PAL is a programmable logic device that has Programmable AND array & fixed OR array. The advantage of PAL is that we can generate only the required product terms of Boolean function instead of generating all the min terms by using programmable AND gates. The **block diagram** of PAL is shown in the following figure.



Here, the inputs of AND gates are programmable. That means each AND gate has both normal and complemented inputs of variables. So, based on the requirement, we can program any of those inputs. So, we can generate only the required **product terms** by using these AND gates.

Here, the inputs of OR gates are not of programmable type. So, the number of inputs to each OR gate will be of fixed type. Hence, apply those required product terms to each OR gate as inputs. Therefore, the outputs of PAL will be in the form of **sum of products form**.

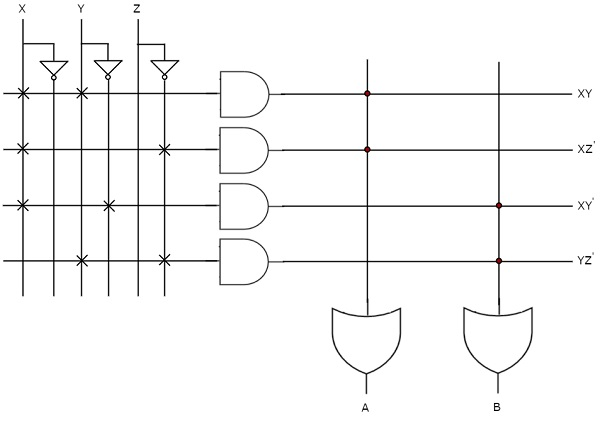
### Example

Let us implement the following **Boolean functions** using PAL.

*A*=*XY*+*XZ*′

*A*=*XY*′+*YZ*′

The given two functions are in sum of products form. There are two product terms present in each Boolean function. So, we require four programmable AND gates & two fixed OR gates for producing those two functions. The corresponding **PAL** is shown in the following figure.

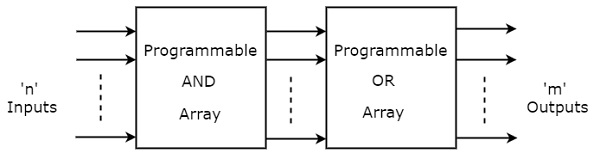


The **programmable AND gates** have the access of both normal and complemented inputs of variables. In the above figure, the inputs X, *X*′, Y, *Y*′, Z & *Z*′, are available at the inputs of each AND gate. So, program only the required literals in order to generate one product term by each AND gate. The symbol ‘X’ is used for programmable connections.

Here, the inputs of OR gates are of fixed type. So, the necessary product terms are connected to inputs of each **OR gate**. So that the OR gates produce the respective Boolean functions. The symbol ‘.’ is used for fixed connections.

**Programmable Logic Array PLA**

PLA is a programmable logic device that has both Programmable AND array & Programmable OR array. Hence, it is the most flexible PLD. The **block diagram** of PLA is shown in the following figure.



Here, the inputs of AND gates are programmable. That means each AND gate has both normal and complemented inputs of variables. So, based on the requirement, we can program any of those inputs. So, we can generate only the required **product terms** by using these AND gates.

Here, the inputs of OR gates are also programmable. So, we can program any number of required product terms, since all the outputs of AND gates are applied as inputs to each OR gate. Therefore, the outputs of PAL will be in the form of **sum of products form**.

### Example

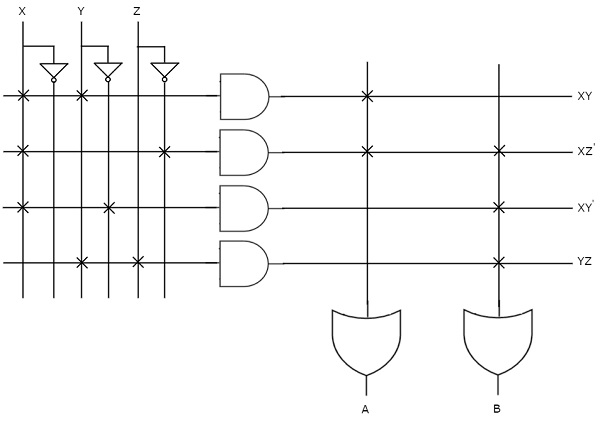
Let us implement the following **Boolean functions** using PLA.

*A*=*XY*+*XZ*′

*B*=*XY*′+*YZ*+*XZ*′

The given two functions are in sum of products form. The number of product terms present in the given Boolean functions A & B are two and three respectively. One product term, *Z*′*X* is common in each function.

So, we require four programmable AND gates & two programmable OR gates for producing those two functions. The corresponding **PLA** is shown in the following figure.



The **programmable AND gates** have the access of both normal and complemented inputs of variables. In the above figure, the inputs X, *X*′, Y, *Y*′, Z & *Z*′

, are available at the inputs of each AND gate. So, program only the required literals in order to generate one product term by each AND gate.

All these product terms are available at the inputs of each **programmable OR gate**. But, only program the required product terms in order to produce the respective Boolean functions by each OR gate. The symbol ‘X’ is used for programmable connections.

DIFFERENCE BETWEEN PLA AND PAL:

| **S.NO** | **PLA** | **PAL** |
| --- | --- | --- |
| **1.** | **PLA stands for Programmable Logic Array.** | **PAL stands for Programmable Array Logic.** |
| **2.** | **PLA speed is lower than PAL.** | **PAL’s speed is higher than PLA.** |
| **3.** | **The complexity of PLA is high.** | **PAL’s complexity is less.** |
| **4.** | **PLA has limited amount of functions implemented.** | **PAL has a huge number of functions implemented.** |
| **5.** | **The cost of PLA is also high.** | **the cost of PAL is low.** |
| **6.** | **Programmable Logic Array is less available.** | **Programmable Array Logic is more available than Programmable Logic Array.** |
| **7.** | **PLA design may be built using a programmable set of AND gates and a programmable set of OR gates.** | **PAL design may be built using a programmable set of AND and a fix set of OR gates** |
| **8.** | **The flexibility of PLA is high as compared to PAL.** | **Flexibility of PAL is less.** |