

# Логичко пројектовање

Предавање 8/10

**Програмабилне логичке  
мреже**

---

## Увод

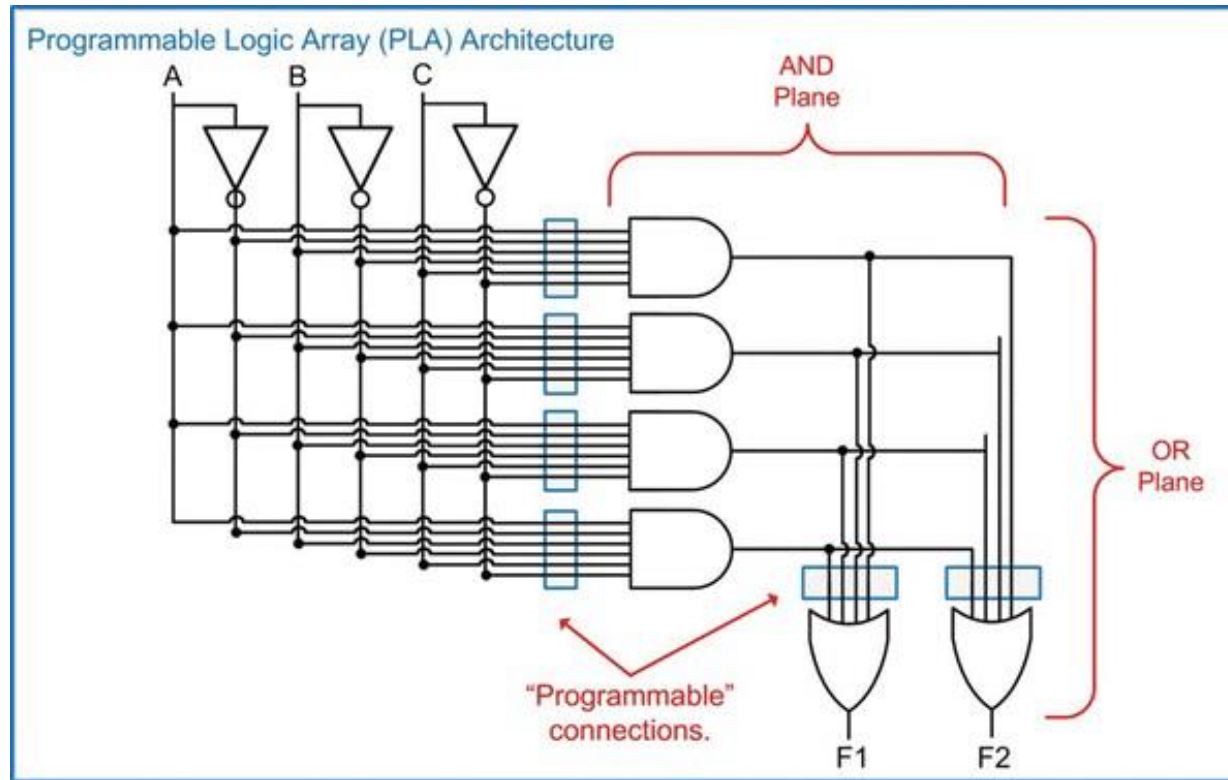
- Ова лекција даје преглед програмабилних логичких мрежа (PLD - Programmable Logic Devices).
- Термин PLD се користи као генерички опис за било које коло које се може програмирати тако да имплементира дигиталну логику.
- Технологија и архитектура PLD-ова временом стално напредује.
- Овде је дат историјски преглед како су први програмабилни уређаји еволуирали у програмабилне технологије које су данас присутне.
- Циљ ове лекције је да пружи основно разумевање о принципима програмабилних логичких уређаја.



## Програмабилни низови - PLA

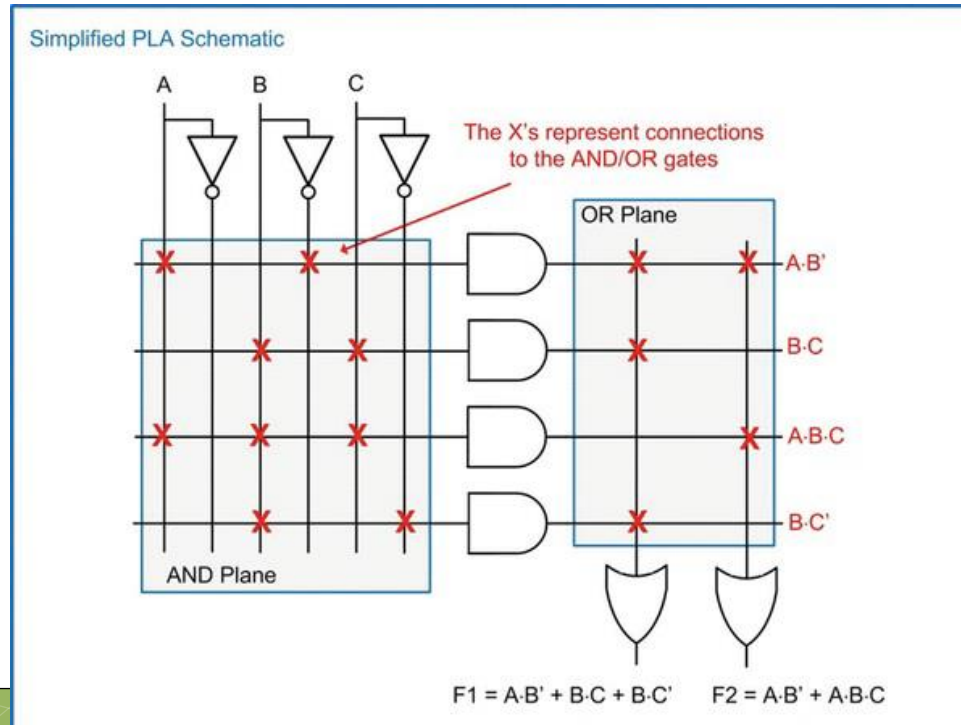
- Један од првих комерцијалних PLD-ова, развијених коришћењем технологије модерних интегрисаних кола, је програмибилни логички низ (PLA).
- 1970. Texas Instruments је представио PLA са архитектуром која подржава имплементацију произвољних збирова логичких производа.
- PLA је била направљена са густим низом AND кола, који се називају AND равни, и густим низом OR кола, који се назива OR раван.
- Сваки улаз у PLA имао је инвертор како би се обезбедила оригинална променљива и њен комплемент.
- Произвољни SOP логички изрази се имплементирају стварањем веза између улаза, AND равни и OR равни.
- Оригинални PLA су произведени са свим неопходним карактеристикама осим коначних прикључака за имплементацију SOP-а функције.
- Када је корисник има жељени SOP израз, везе се додају као последњи корак израде. Ова конфигурациона техника је била слична MROM приступу.
- Слика приказује основе архитектуре PLA.

## Архитектура PLA



## Упрошћена PLA шема

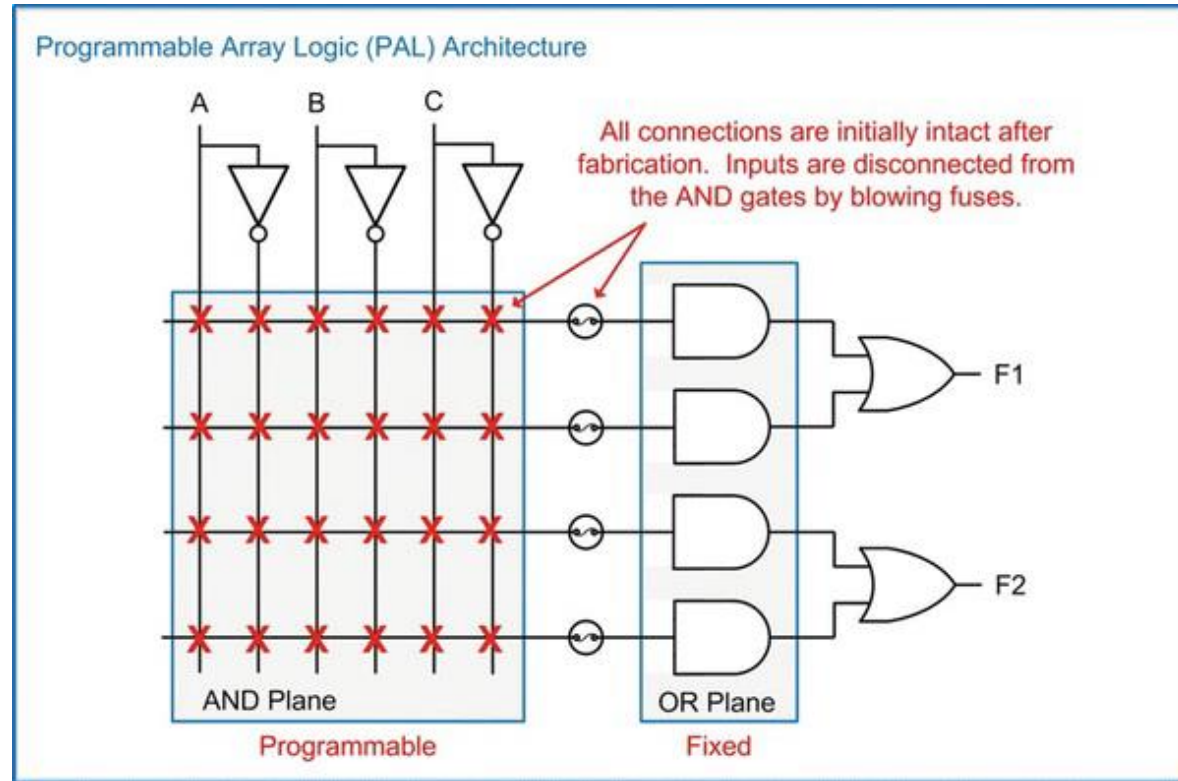
- Компактнија шема за PLA се црта представљањем свих улаза у AND и OR кола са једним проводником. Везе су назначене уметањем 'x' на пресецима жица.



## Програмабилни низови - PAL

- Један од недостатака оригиналног PLA је била програмибилност OR равни што је изазвало велика пропагациона кашњења кроз комбинациону логику кола.
- Да би се побољшале перформансе PLA-а, уведен је (PAL – Programmable Array Logic)
- Први PAL је 1978. године представила компанија Monolithic Memories, Inc.
- PAL је садржао програмабилну AND раван и фиксну OR раван.
- Фиксна OR раван је побољшала перформансе ове програмабилне мреже.
- Иако нема програмабилне OR равни где је смањена флексибилност мреже, већина SOP израза се може имплементирати са PAL-ом.
- Још један допринос PAL-а је да се AND раван може програмирати коришћењем осигурача. У почетку су све везе биле присутне у OR равни.
- За прегоривање осигурача се користи спољашњи програматор како би се искључили улази у OR коло. Док је приступ осигурачима омогућен само једнократно, способност пост-фабрикације за конфигурисање логике био је значајан напредак у односу на PLA, који је морао бити програмиран код произвођача.

# PAL архитектура



## Генерички логички низ

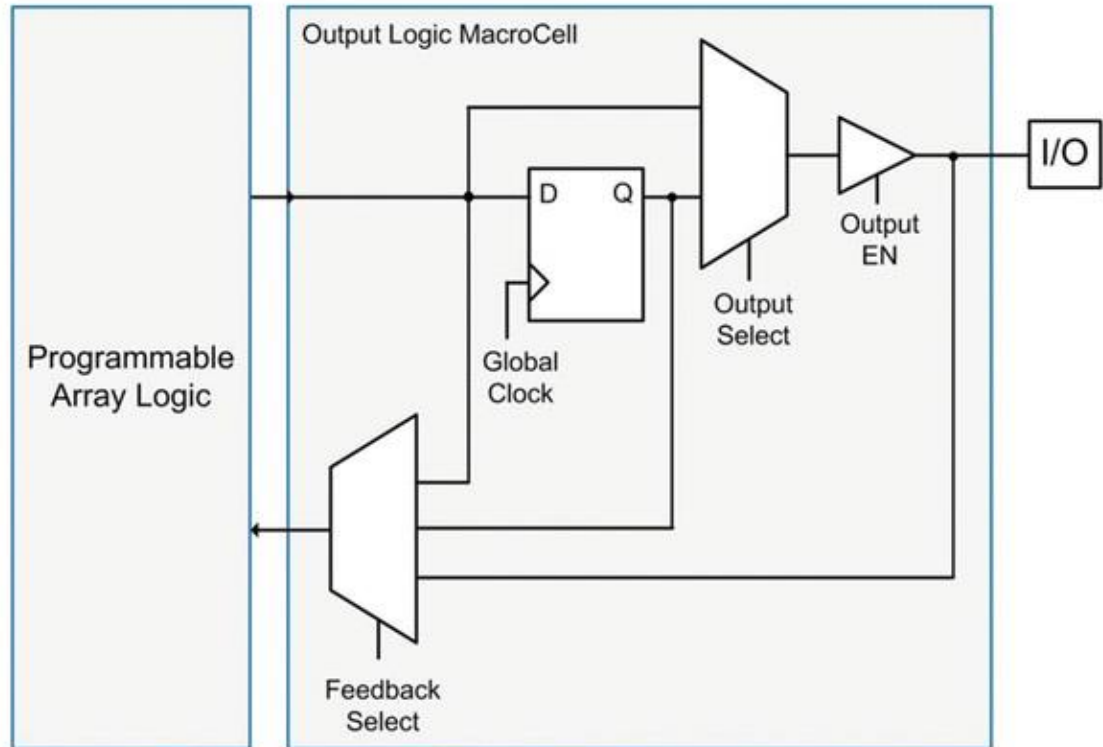
- Како је популарност PAL-а расла, имплементирана је додатна функционалност која подржава и софистицираније дизајне.
- Један од најзначајнијих побољшања је био додатак излазне логичке макро ћелије (OLMC – output logic macro cell).
- OLMC је изграђена од D флип-флопа и MUX-а где се излази из SOP кола из PAL-а могу користити или као системски излази или као улаз за D-флип-флопове.
- Ово омогућила имплементацију секвенцијалне логике и стања коначних аутомата.
- OLMC би се такође могао користити за усмеравање I/O пина назад у PAL да би се повећао број могућих улаза у SOP изразима.
- Коначно, OLMC је обезбедио мултиплексер који омогућава повратне информације било са PAL излаза или излаза D флип-флоп-а.
- Ова архитектура је названа генерички логики низ (GAL) која проширује карактеристике стандардног PAL-а.



# GAL архитектура

## Generic Array Logic (GAL) Architecture

Adding an output logic macrocell to a PAL allows the system to implement feedback, create sequential logic, or use the I/O pin as either an input or output.



## Чврсти логички низ

- За велике пројекте, PAL-ови и GAL-ови се могу имплементирати као чврсти логички низ (HAL – Hard Array Logic) мрежа.
- HAL је била верзија PAL-а и GAL-а која је имала везе у AND равни и које су се формирале током фабрикације уместо преко осигурача.
- Ова архитектура је била ефикаснија за апликације великог обима где се елиминише корак програмирања након израде и уређај није морао да садржи додатна кола за програмирање.
- 1983. основана је Altera Inc. као компанија за програмибилне логичке уређаје.
- Године 1984. Altera је објавила своју прву верзију PAL-а са јединственом карактеристиком где се може програмирати и брисати више пута помоћу програматора и УВ уређаја сличан EEPROM-у.

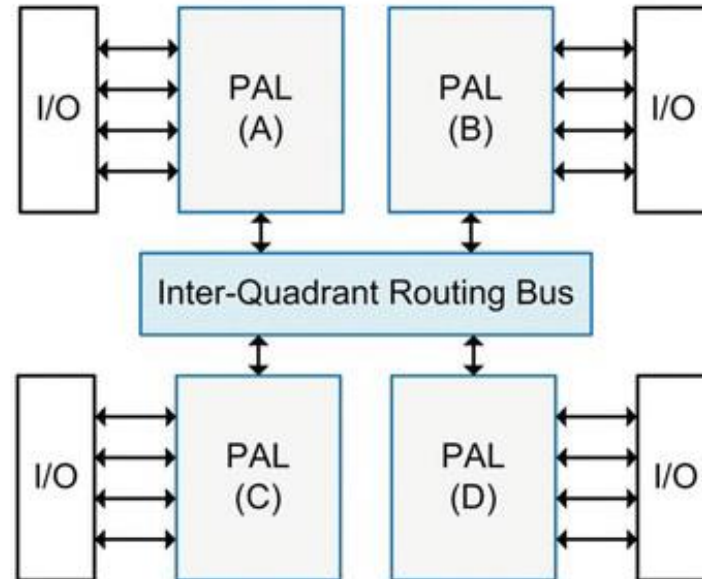
## Комплексни PLD

- Како је расла потражња за већим програмабилним мрежама, архитектура PAL-а није била у могућности да ефикасно скалира из више разлога:
  - величина комбинационе логике се повећала, PAL је наишао на проблеме са великим бројем улаза у AND равни;
  - за сваки улаз који је додат у PAL, број кола потребних на чипу је геометријски порасла због потребе за повезивањем сваког AND кола са областима повезаних са додатним OLMC-ом.
- Ово је довело до нове PLD архитектуре у којој је била интерконекиција на чипу подељеног на више PAL-ова на једном чипу. Ова подела је значила да PAL не може користити све улазе, тако да је сложеност дизајна повећана;
- Ова нова архитектура била је назван сложени програмибилни логички уређај (CPLD – Complex PLD). Термин једноставан PLD (SPLD – Simple PLD) је креиран да опише све претходне PLD архитектуре (тј. PLA, PAL, GAL и HAL).

## Архитектура комплексног PLD-а

### Complex PLD (CPLD) Architecture

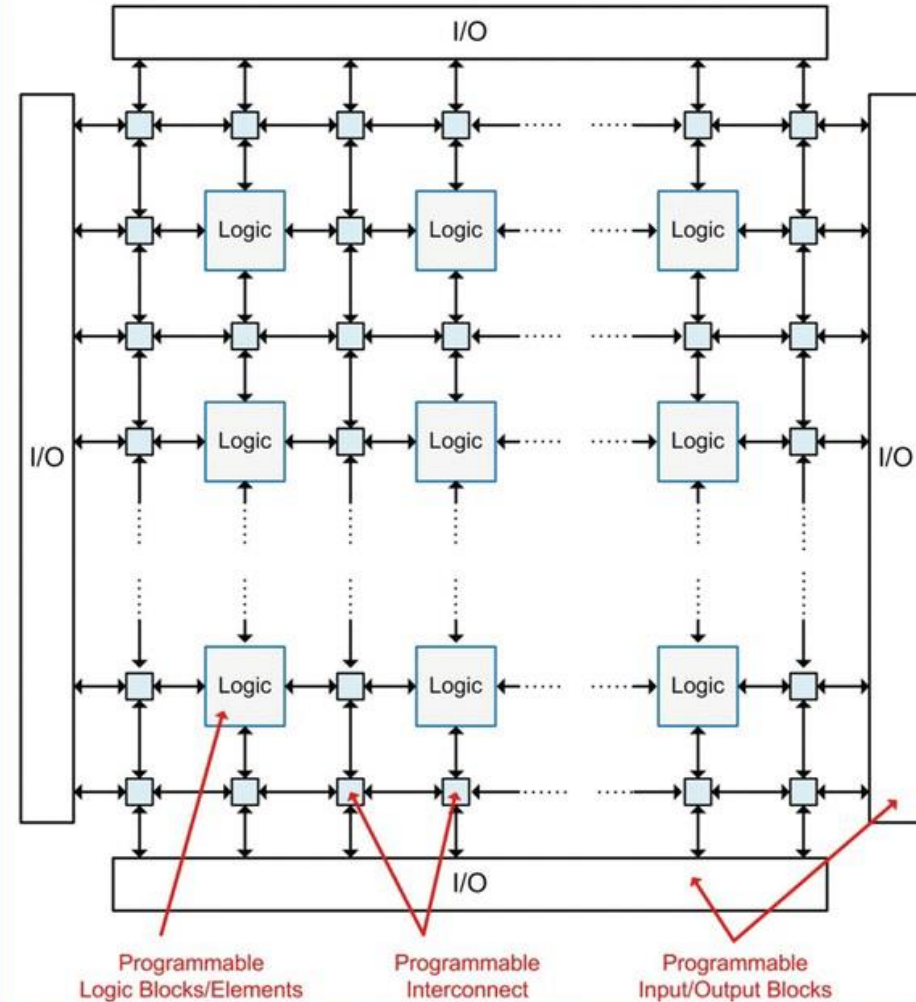
Partitioning the routing allowed more PAL blocks to be integrated on a single chip. This architecture also implied that not all SOP expressions had access to every I/O.



## FPGA

- Да бисмо одговорили на потребу за још више програмабилних ресурса, нову архитектуру је развио Xilinx Inc. 1985. године.
- Ова нова архитектура је названа поље програмабилних низова кола (FPGA – Field Programmable Gate Array).
- FPGA се састоји од низа програмабилних логичких блокова (или логичких елемената) и мреже програмабилних веза које се могу користити за повезивање било ког логичког елемента са било којом другом логичким елементом.
- Сваки логички блок садржи кола за имплементацију произвољне комбинационе логике поред D-флип-флоп-ова и мултиплексера за управљање сигнаlima.
- Ова архитектура је ефективно имплементирала OLMC унутар сваког од блокова, чиме се обезбеђује врхунска флексибилност и пружа знатно више ресурси за секвенцијалну логику.
- Данас се најчешће користе FPGA програмибилне логичке мреже, при чему су Altera Inc. и Xilinx Inc. највећи произвођачи.

# Архитектура FPGA



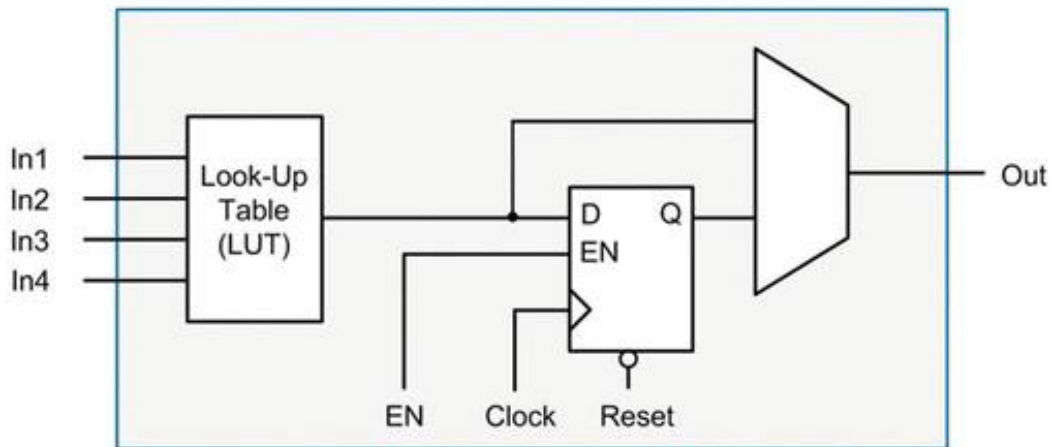
## Конфигурабилни логички блок (или логички елемент)

- Примарна реконфигурабилна структура у FPGA је конфигурабилни логички блок (CLB) или логички елемент (LE).
- Xilinx Inc. користи термин CLB док Altera Inc користи LE.
- Комбинациона логика се имплементира помоћу кола која се зову Look-Up Table (LUT), која може да имплементира било коју произвољну таблицу истинитости. Детаљи LUT-а су дати у следећем одељку.
- CLB/LE такође садржи D-флип-флоп-ове за секвенцијалну логику. Мултиплексер за управљање сигнаlima се користи за одабир да ли ће излаз CLB/LE доћи из LUT-а или из D-флип-флопа.
- LUT се може користити за имплементацију комбинационог логичког израза на D улаз од D флип-флоп-а, стварајући тако високо ефикасну топологију за коначне аутомате.
- Глобална мрежа за рутирање се користи за обезбеђивање заједничких сигнала за CLB/LE као што су clock, reset, и enable.
- Ова глобална мрежа за рутирање може то да обезбеди заједничке сигнале за цео FPGA или локалне групе CLB/LE.

# Архитектура FPGA ЛОГИЧКОГ ЕЛЕМЕНТА

## Simple FPGA Configurable Logic Block (or Logic Element) Architecture

The logic block contains a Look-Up Table (LUT) to implement any arbitrary combinational logic circuit. The output of the LUT can be selected as the block output or as the input to a D-Flip-Flop. When used as the input to the D-Flip-Flop, Q is selected as the block output.



These signals are from  
local/global routing networks

Not shown in this diagram are programming lines to configure the LUT and MUX. Lines for sequential logic come from local/global routing networks that can drive multiple blocks.



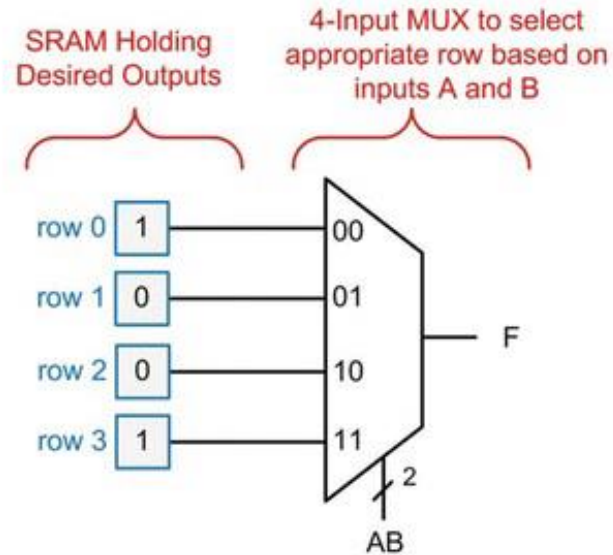
## LUT

- LUT је примарно коло које се користи за имплементацију комбинационе логике у FPGA.
- Ова топологија је такође усвојена у модерним CPLD-овима.
- У LUT-у, жељени излази таблице истинитости се учитавају у локалну конфигурациону SRAM меморију. SRAM меморија даје ове вредности улазима мултиплексора. Улази комбинационог логичког кола се затим користе као линије за избор мултиплексера.
- За произвољан улаз у комбинационо логичко коло, мултиплексер бира одговарајућу вредност која се налази у SRAM меморији и усмерава је на излаз кола.
- На овај начин, мултиплексер тражи одговарајући излаз вредност на основу улазног кода. Ова архитектура има предност од сваке логичке функције се може креирати без креирања прилагођеног логичког кола. Такође, кашњење кроз LUT је идентично без обзира на то каква је логичка функција имплементирана.
- На слици је приказано комб. логичко коло са 2 улаза имплементирано са 4-улазним мултиплексером.

## 2- улазни LUT

2-Input LUT Implemented with a 4-Input Multiplexer

| A | B | F |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |



## Ограничења код LUT-а

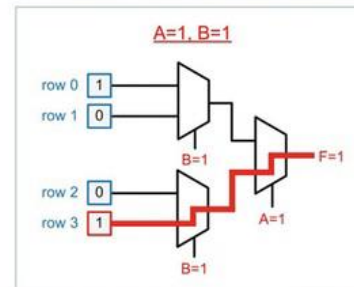
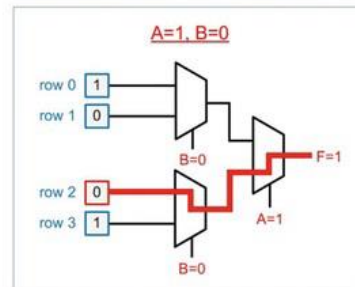
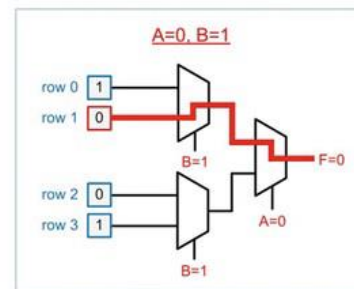
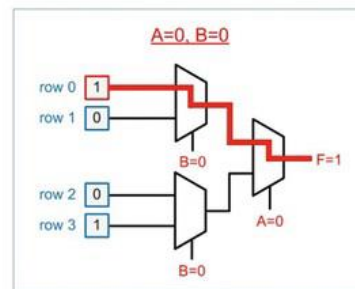
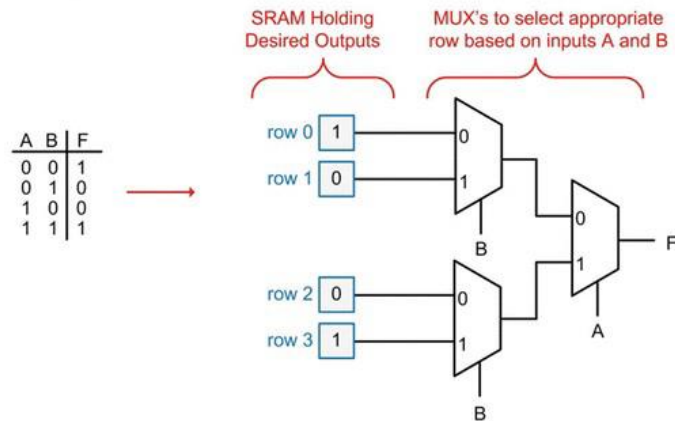
- Ограничења везана за максимални број улаза могу се наићи у ЛУТ-овима јер број комбинационих логичких кола који се имплементира расте.
- На пример, мултиплексори суимплементирани са SOP топологијом у којој сваки производ на првом нивоу логике има број улаза једнак броју селектованих линија плус један. Такође подсетимо да је збирни члан на другом нивоу логике код SOP топологија има број улаза једнак укупном броју улаза у мултиплексер.
- У примеру кола приказаног на слици сваки производ у изразу у мултиплексру ће имати три улаза, а збирни члан ће имати четири улаза.
- Као илустрацију колико брзо се наилазе на ограничења уласка броја линија, може се размотрити импликација повећања броја улаза на слици испод са два на три.
- И нове конфигурације, број улаза у термовима производа ће се повећати са три на четири и број улаза у збирном терму ће се повећати од четири до осам. Осам улаза често превазилази модерне спецификације за максимални број линија кола, што значи да ће наићи чак и комбинациона логичка кола са 3 улаза на проблеме са максималним бројем линија.

## Решавање проблема ограничења код LUT-а

- Да би се решио овај проблем, функционалност мултиплексера у LUT-овима је типично имплементиран као серија мањих, каскадних мултиплексера.
- Сваки од мањих мултиплексера прогресивно бира који ред таблице истинитости ће усмерити до излаз LUT-а.
- Ово елиминише проблеме са максималним бројем линија на рачун додавања додатних нивоа логике у коло.
- Док каскадни мултиплексери повећавају укупно кашњење кола, овим приступом се постиже високо конзистентно кашњење јер без обзира на излазну вредност таблице истинитости, број нивоа логике преко мултиплексера је увек исто.
- Слика показује како 2-улазна таблица истинитости са слике претходне слике може да се замени коришћењем каскадне везе од 2 нивоа од 2 улаза мултиплексера.
- Ако је потребно више улаза у LUT, додају се додатни мултиплексери. Друга наредна слика приказује архитектуру за LUT са 3 улаза имплементирану са 3-каскадна нивоа 2-улазних мултиплексера.

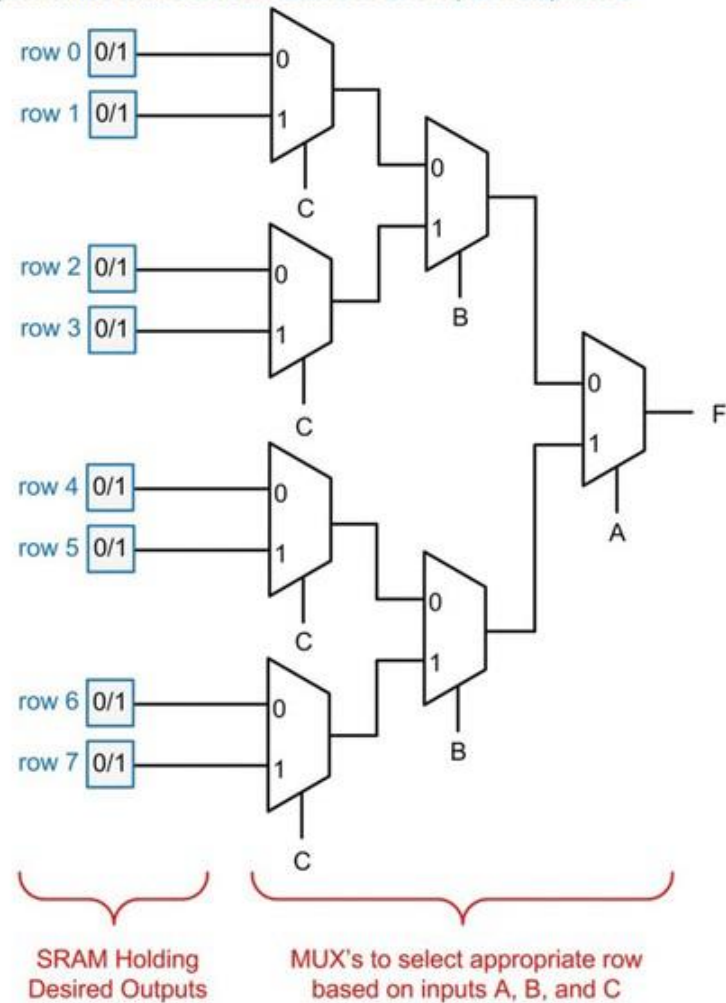
# Пример имплементације LUT-a

## 2-Input LUT Implemented with a 2-Level Cascade of 2-Input Multiplexers



# Архитектура LUT-a

3-Input LUT Implemented with a 3-Level Cascade of 2-Input Multiplexers



## Програмабилне унутрашње везе

### PIP – Programmable Interconnect Points

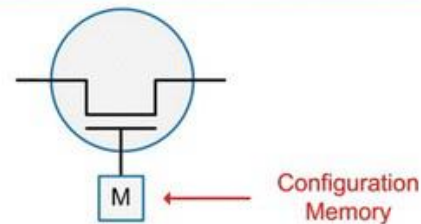
- Мрежа за рутирање која се може конфигурисати у FPGA се реализује коришћењем програмабилних прекидача.
- Једноставан модел за ове прекидаче је коришћење NMOS-транзистора.
- Конфигурациони SRAM бит чува да ли је прекидач отворен или затворен.
- На FPGA, интерконекиција се усмерава вертикално и хоризонтално између CLB/LE са тачкама пребацивања постављеним по целој површини FPGA да би се олакшала произвољна конфигурација рутирања.
- Слика показује како може бити конфигурисано рутирање у потпуну унакрсну конфигурацију помоћу програмабилних прекидача.

# PIP конфигурација

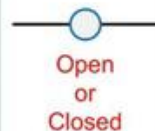
## FPGA Programmable Interconnect

A simple model for a programmable interconnection is an NMOS transistor that connects or disconnects two wires. The switch is controlled using a configuration bit.

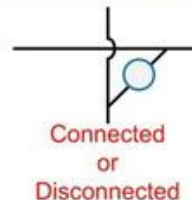
This can be used in a variety of configurations.



### Single Wire

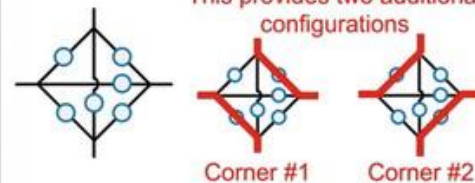


### Two Intersecting Wires

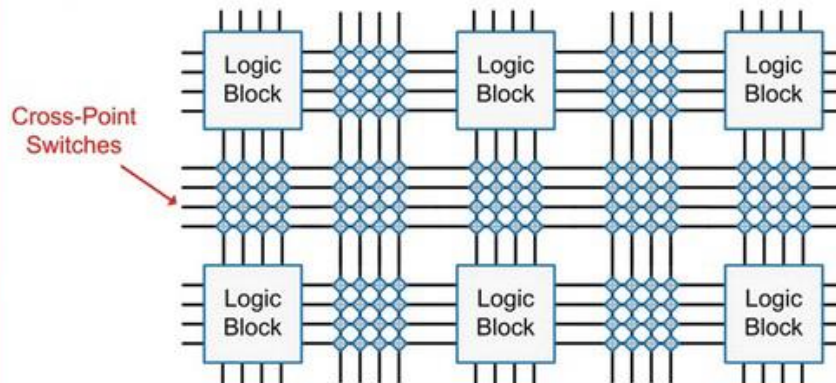


### 6-Position Cross-Point

This provides two additional configurations



The switches are placed at the intersections of horizontal and vertical routing lanes on the FPGA.



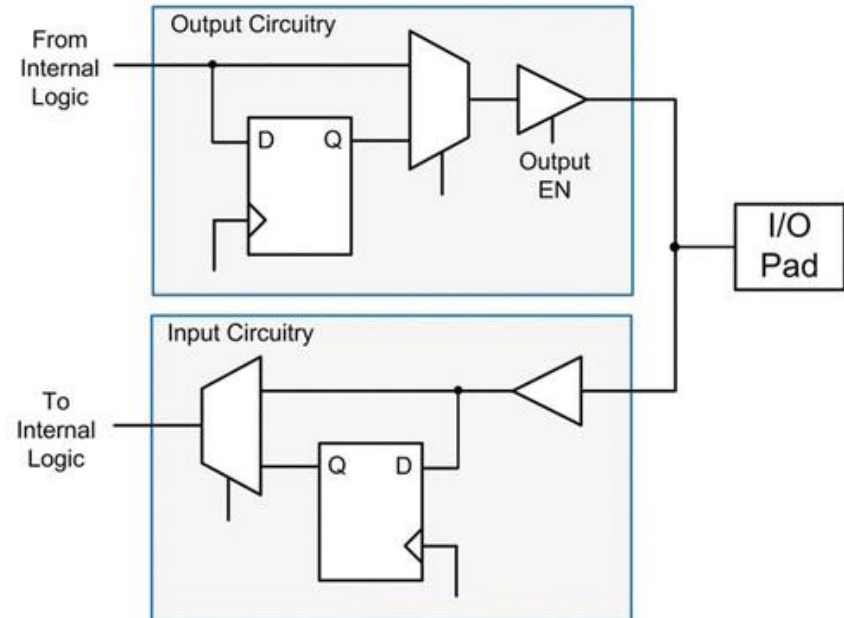


## Улазно/Изразни блок

- FPGA такође садрже и улазно/излазне блокове (IOB) који омогућавају програмирање функционалности за повезивање са спољним колима.
- IOB-ови садрже драјвере и кола пријемника тако да се могу програмирати да буду или улази или излази.
- D-флип-флопови су укључени и у улазна и излазна кола за подршку синхроне логике.
- Слика приказује архитектуру FPGA IOB.

### FPGA Input / Output Block (IOB)

The IOB can be programmed to either be an input or output. Both input and output circuits contain D-Flip-Flops to support synchronous logic. Placing the D-Flip-Flops close to the I/O pad reduces differences in propagation delays between package pins.



## Конфигурациона меморија

- Све информације о програмирању за FPGA се налазе унутар конфигурационе SRAM меморије која је дистрибуирана преко IC.
- Пошто је ова меморија привремена, FPGA ће изгубити своју конфигурацију када се искључи напајање.
- Након укључивања, FPGA мора бити програмиран са својим конфигурационим подацима. Ови подаци се обично чувају у сталној меморији као што је FLASH.
- „FP“ у FPGA односи се на способност програмирања уређаја на терену, или накнадну производњу.
- „GA“ у FPGA се односи на топологију низа програмабилних логичких блокова или елемената.