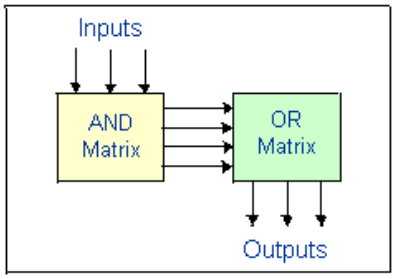
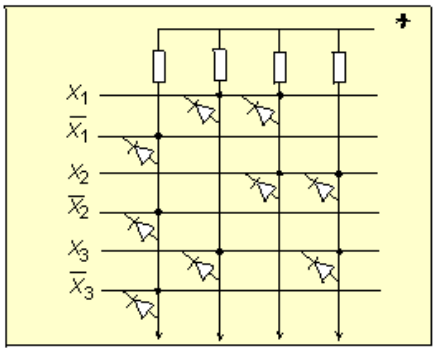
Programmable Logic Devices

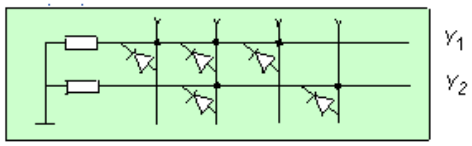
Basic Structure



AND Matrix



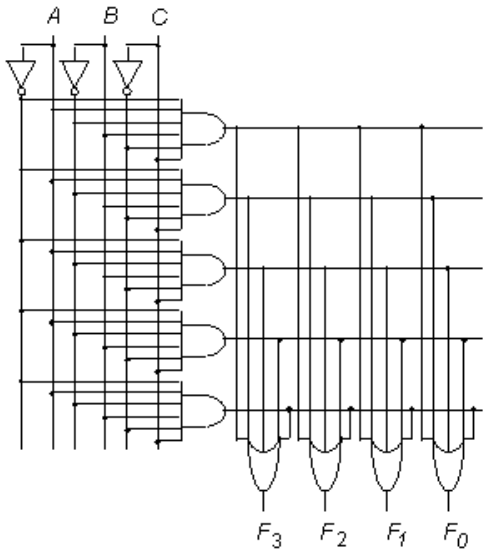
OR Matrix

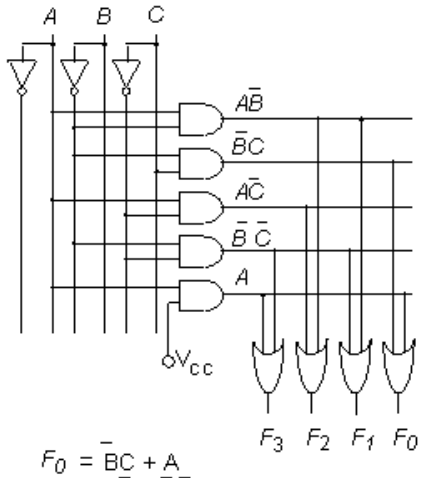


$Y1 = P1 + P2 + P3 = \overline{X1}.\overline{X2}.\overline{X3} + X1.X3 + X1.X2$

$Y2 = P2 + P4 = X1.X3 + X2X3$

Implementation Boolean Equations - an Example





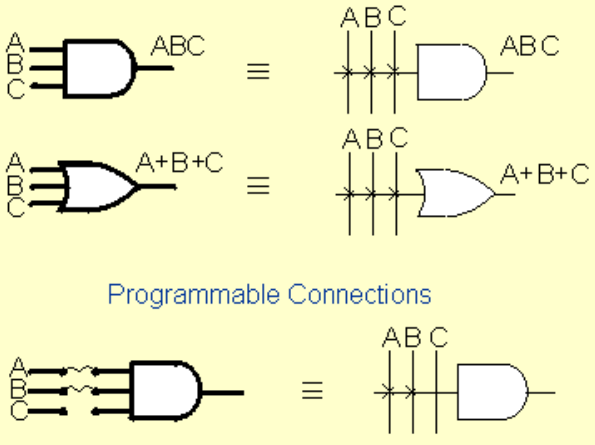
$F_0 = \bar{B}C + A$

$F_1 = A\bar{B} + \bar{B}\bar{C}$

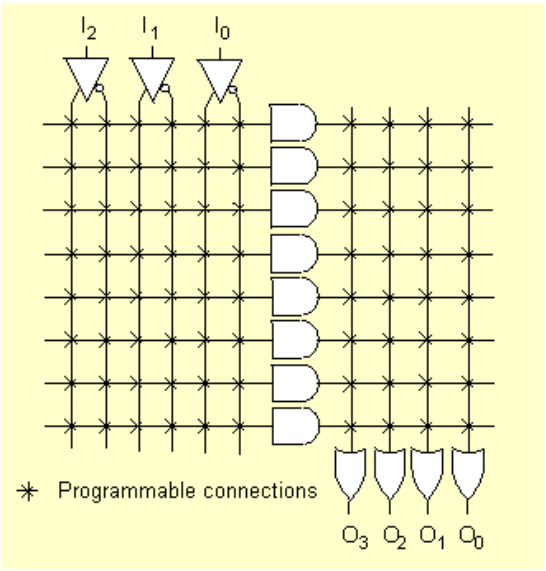
$F_2 = A\bar{B} + A\bar{C}$

$F_3 = \bar{B}\bar{C} + A$

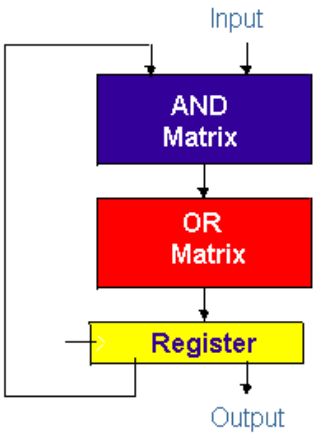
PLD Standard Notations



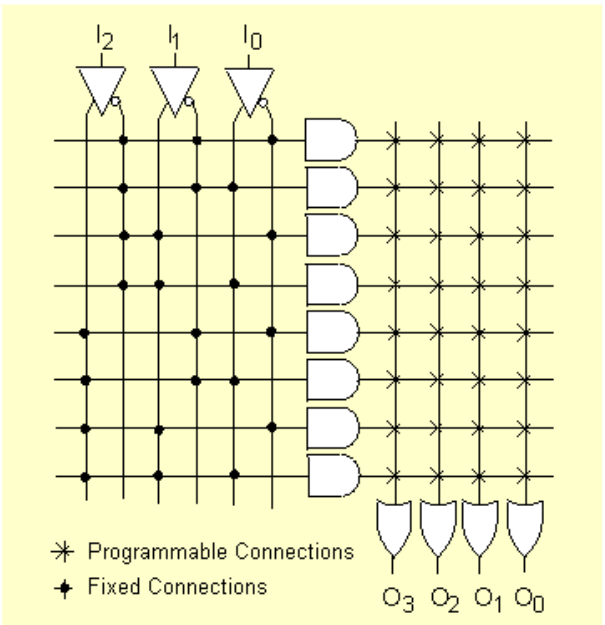
PLA Matrix



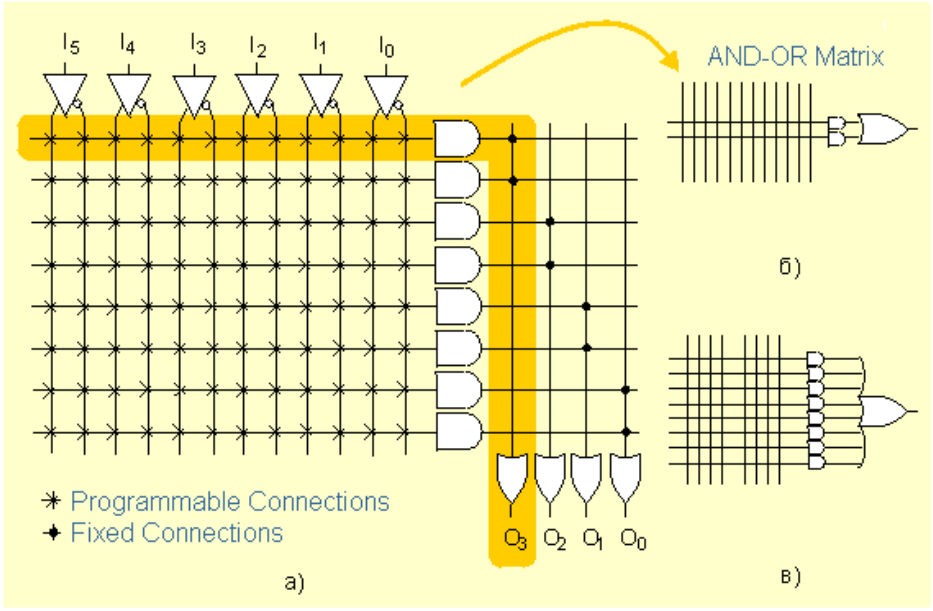
PLD with Register



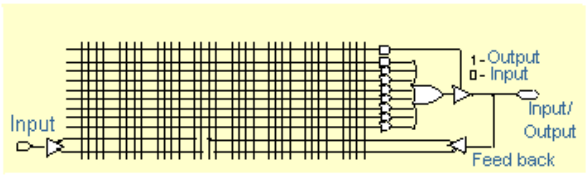
PROM Architecture



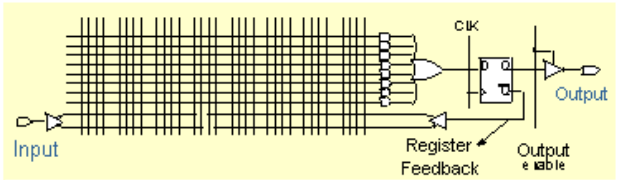
PAL Architecture



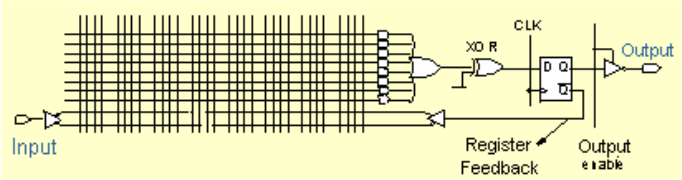
Input/Output Types in PAL



Programmable Input/Output

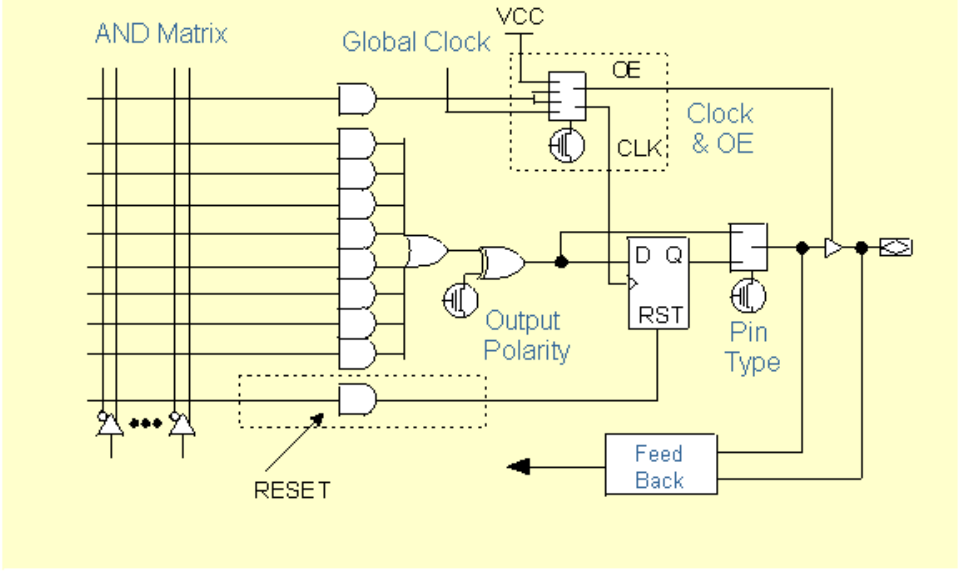


Register Output



Output Programmable Polarity

Input Output Macrocell



Complex Programmable Logic Devices - CPLD

Complex programmable logic devices (CPLD)

- Overall organization has evolved from purely combinational devices.

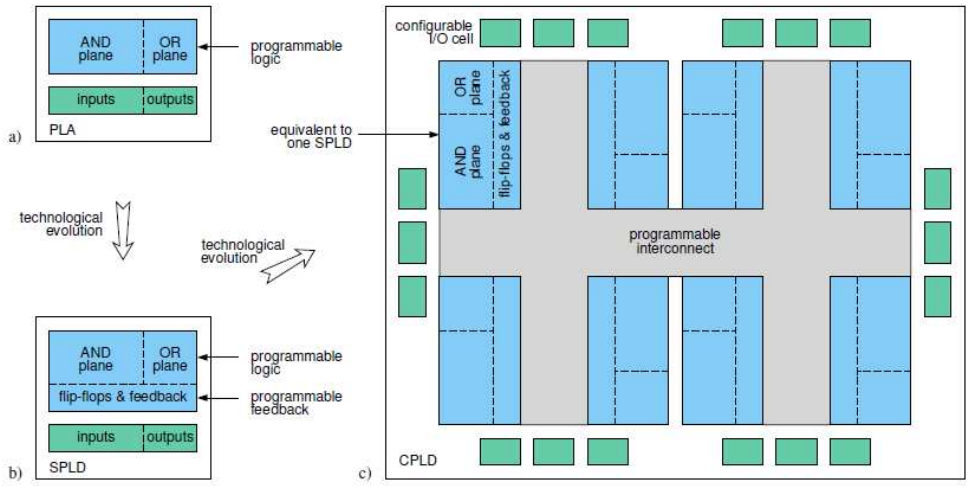
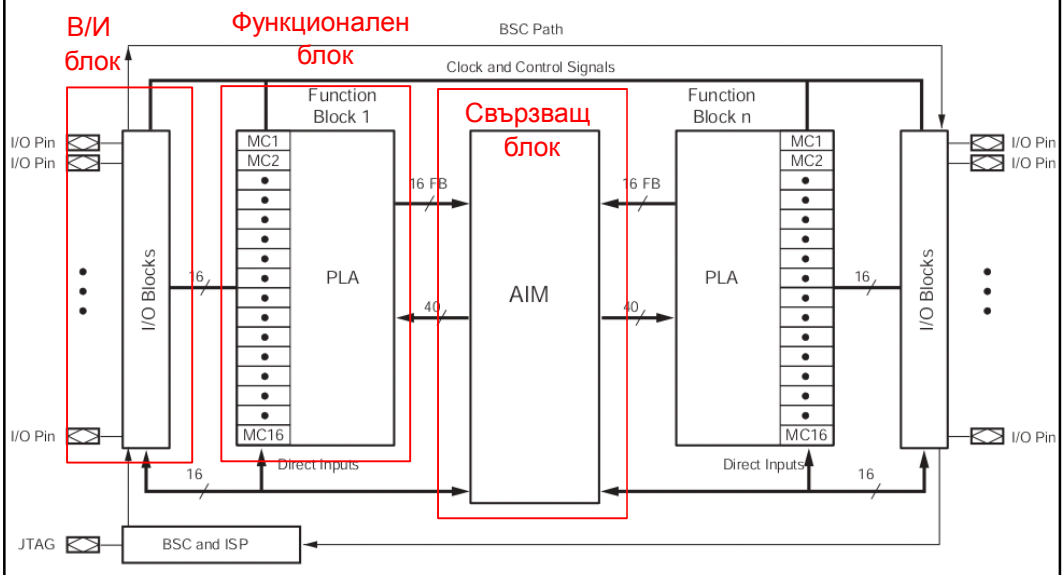


Figure: General architecture of CPLDs (c) along with precursors (a,b).

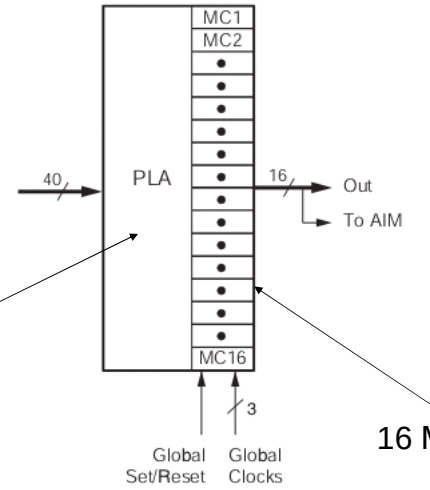
Типична CPLD Архитектура



Xilinx CoolRunner

CoolRunner – Функционален Блок

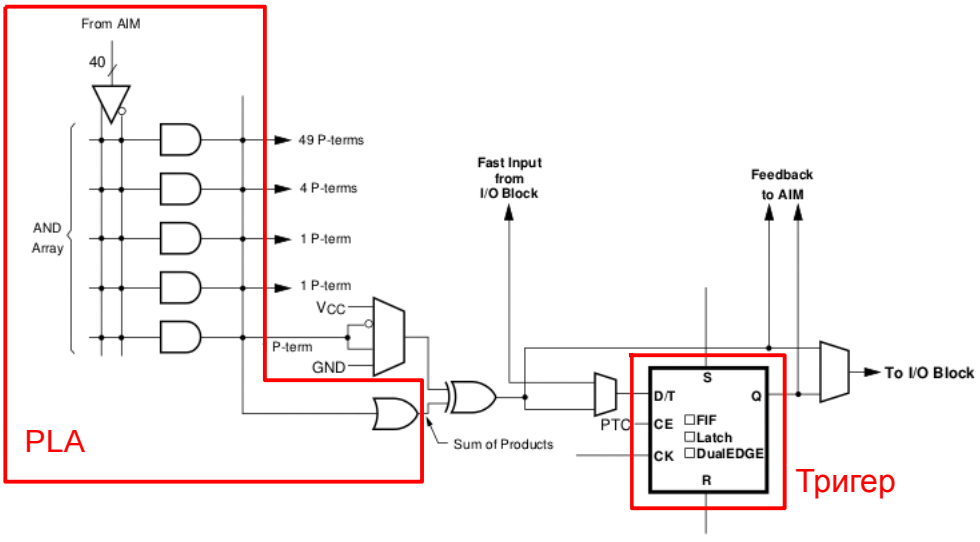
В един CoolRunner чип има от 2 до 32 функционални блока.



PLA с 40 входа и 56 произведения (product terms)

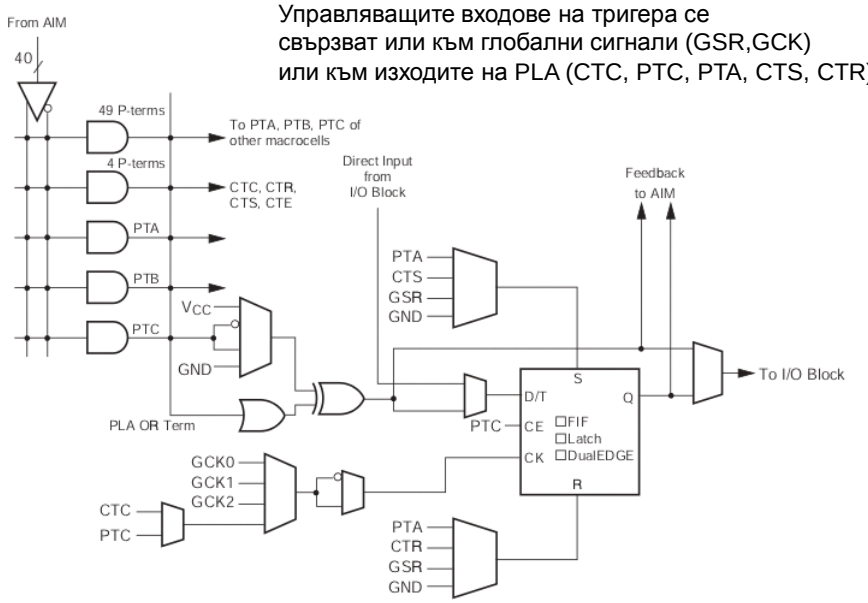
16 Макроклетки

CoolRunner – Макроклетка (опростена схема)

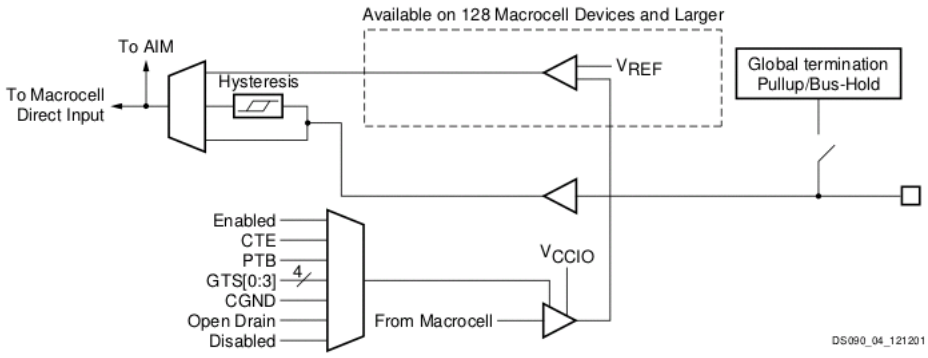


CoolRunner - Макроклетка (цялостна схема)

Управляващите входове на тригера се свързват или към глобални сигнали (GSR,GCK) или към изходите на PLA (CTC, PTC, PTA, CTS, CTR)

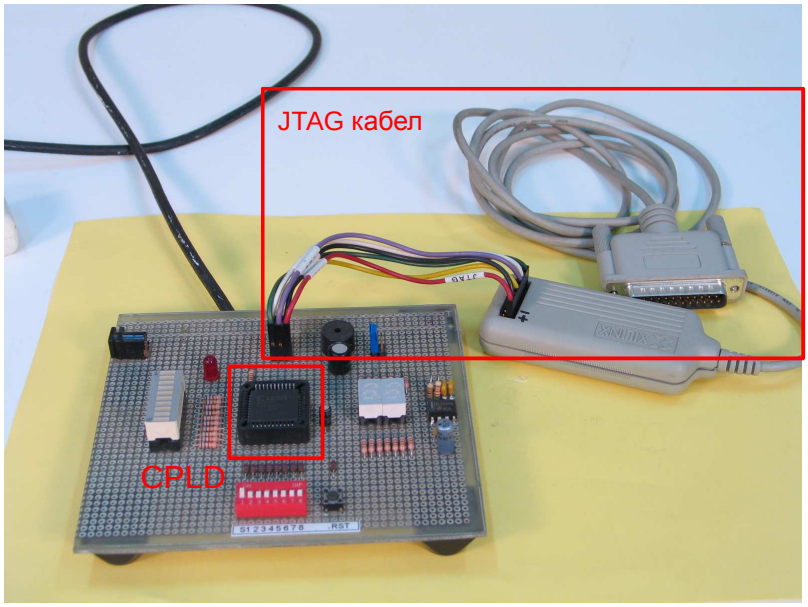


CoolRunner – Входно/Изходен Блок



Програмиране на CoolRunner

Записването на конфигурацията се извършва чрез JTAG интерфей.



Специални функции на CoolRunner-II

DualEDGE registers

```
Нарастващ фронт:
if(clock'event)and(clock='1')then

И по двата фронта:
if(clock'event)then
```

Делител

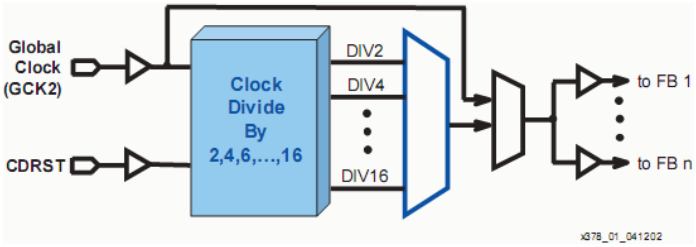


Figure 1: **CoolRunner-II Clock Divider**

```
Component CLK_DIV2 is
  port(
  CLKIN:inSTD_LOGIC;
  CLKDV:outSTD_LOGIC);
end component;
```

```
U1:CLK_DIV2
       portmap(
       CLKIN => clk,
       CLKDV => clk_div_by_2);
```

Тригер на Шмит

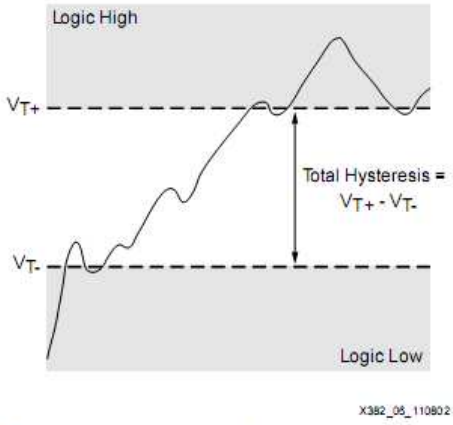
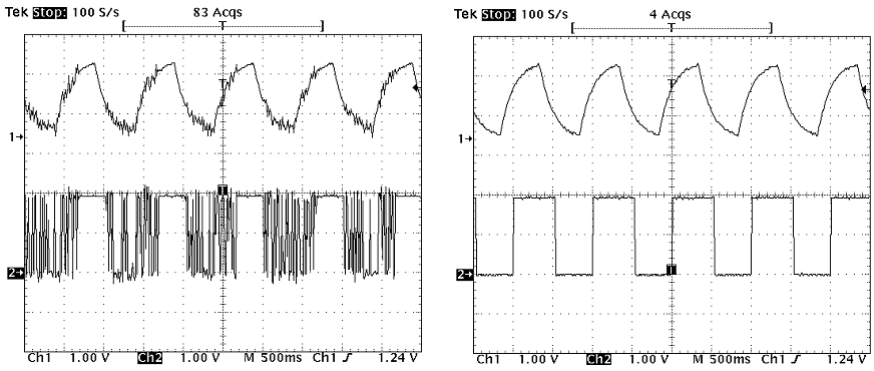


Figure 5: **Representation of a Schmitt Trigger Input Buffer**

Тригер на Шмит



Тригер на Шмит

UCF файл

```
NET <signalname> SCHMITT_TRIGGER;
```

Пример: `NET data_in SCHMITT_TRIGGER;`

VHDL

```
attribute SCHMITT_TRIGGER:STRING;
attribute SCHMITT_TRIGGER of <signalname>: signal is "TRUE";
```

Пример:

```
attribute SCHMITT_TRIGGER:STRING;
attribute SCHMITT_TRIGGER of data_in: signal is "TRUE";
```

В/И Стандарти

Table 9: **CoolRunner-II Supported I/O Standards**

	XC2C32	XC2C64	XC2C128	XC2C256	XC2C384	XC2C512
I/O Banks	1	1	2	2	4	4
LVTTL	Yes	Yes	Yes	Yes	Yes	Yes
LVCMOS33, LVCMOS25, & LVCMOS18	Yes	Yes	Yes	Yes	Yes	Yes
1.5V I/Os	Yes	Yes	Yes	Yes	Yes	Yes
SSTL2-1 & SSTL3-1	No	No	Yes	Yes	Yes	Yes
HSTL-1	No	No	Yes	Yes	Yes	Yes

```
NET data_in IOSTANDARD = LVCMOS18;
```

```
attribute IOSTANDARD : STRING;
attribute IOSTANDARD of data_in : signal is "LVCMOS18";
```