Razi University

# Logic Circuits Design


Computer Engineering Department of
Razi University

**Dr. Abdolhossein Fathi**

# پیاده سازی توابع با ابزارهای آماده و قابل برنامه ریزی

# انواع شکل مدارات دو طبقه

هر تابع منطقی با هر شکل و اندازه ای با استفاده از یک جدول درستی قابل نمایش است؛ و می توان آن را با دوطبقه از گیتهای منطقی مختلف پیاده سازی نمود. البته در شکل های استاندارد از فرم های دوطبقه **AND-OR** یا **OR-AND** استفاده می شود. کل ترکیبات ممکن از گیتها مختلف در جدول زیر آورده شده است.

| طبقه ۱ \ طبقه ۲ | AND | OR | NAND | NOR |
|---|---|---|---|---|
| AND | | ★ | | ★ |
| OR | ★ | | ★ | |
| NAND | ★ | | ★ | |
| NOR | | ★ | | ★ |

برخی از این ترکیبات را می توان با استفاده از دیکدر و مالتی پلکسر پیاده سازی نمود

# پیاده سازی توابع منطقی با دیکدر ها

**مثال:** $F(A, B, C) = \sum m(0, 1, 4, 6, 7) = \prod M(2, 3, 5)$

بسته نوع دیکدر تابع را به چندین طریق می توانیم پیاده نماییم:

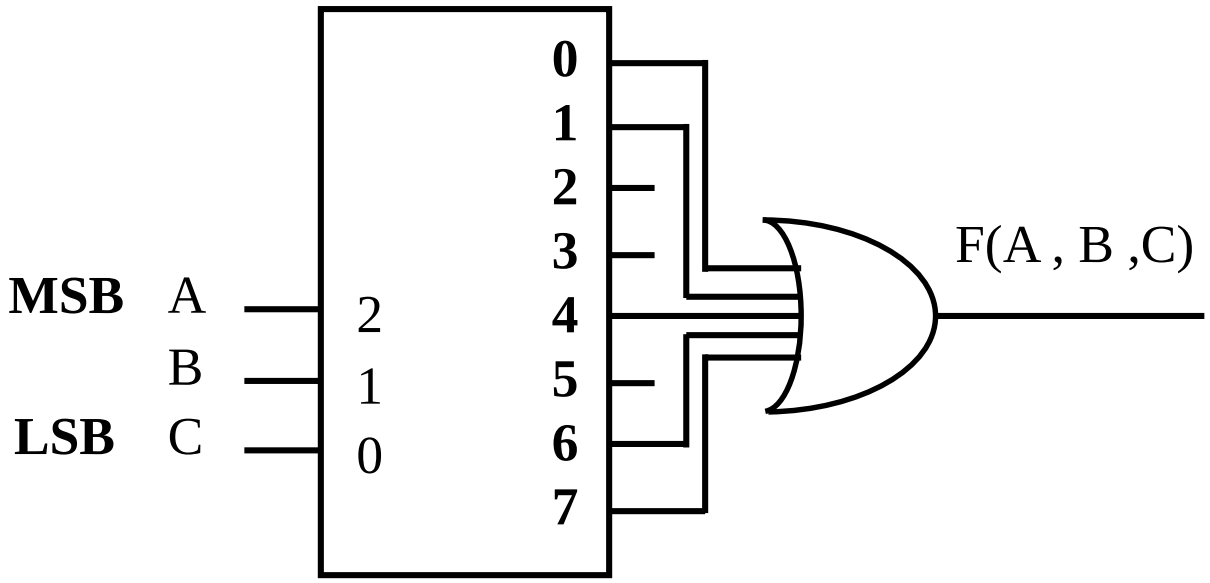
۱. یک دیکدر (با خروجی فعال بالا) ویک گیت OR بکار بریم.
۲. یک دیکدر (با خروجی فعال پایین) ویک گیتNAND بکار بریم.
۳. یک دیکدر (با خروجی فعال بالا) ویک گیت NOR بکار بریم.
۴. یک دیکدر (با خروجی فعال پایین) ویک گیت AND بکار بریم.

# پیاده سازی توابع منطقی با دیکدر ها

## یک دیکدر (با خروجی فعال بالا) ویک گیت OR بکار بریم.

$$F(A, B, C) = \sum m(0, 1, 4, 6, 7) = \prod M(2, 3, 5)$$

$$F(A, B, C) = m_0 + m_1 + m_4 + m_6 + m_7$$

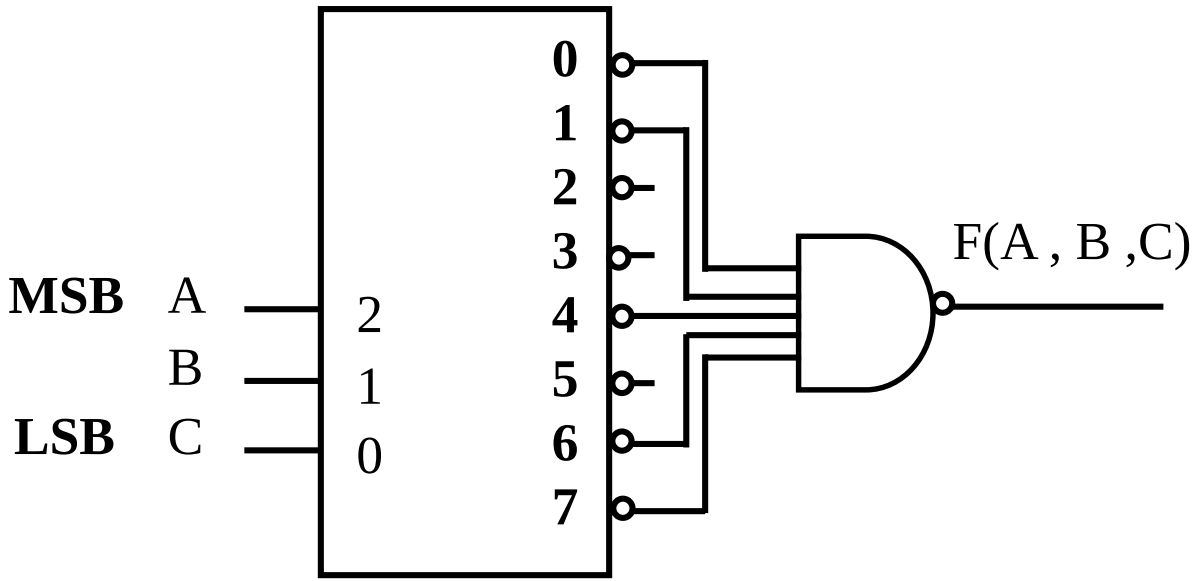


# پیاده سازی توابع منطقی با دیکدر ها

## یک دیکدر (با خروجی فعال پایین) ویک گیت NAND بکار بریم.

$$F(A, B, C) = \sum m(0, 1, 4, 6, 7) = \prod M(2, 3, 5)$$

$$F(A, B, C) = \overline{\bar{m}_0 . \bar{m}_1 . \bar{m}_4 . \bar{m}_6 . \bar{m}_7}$$

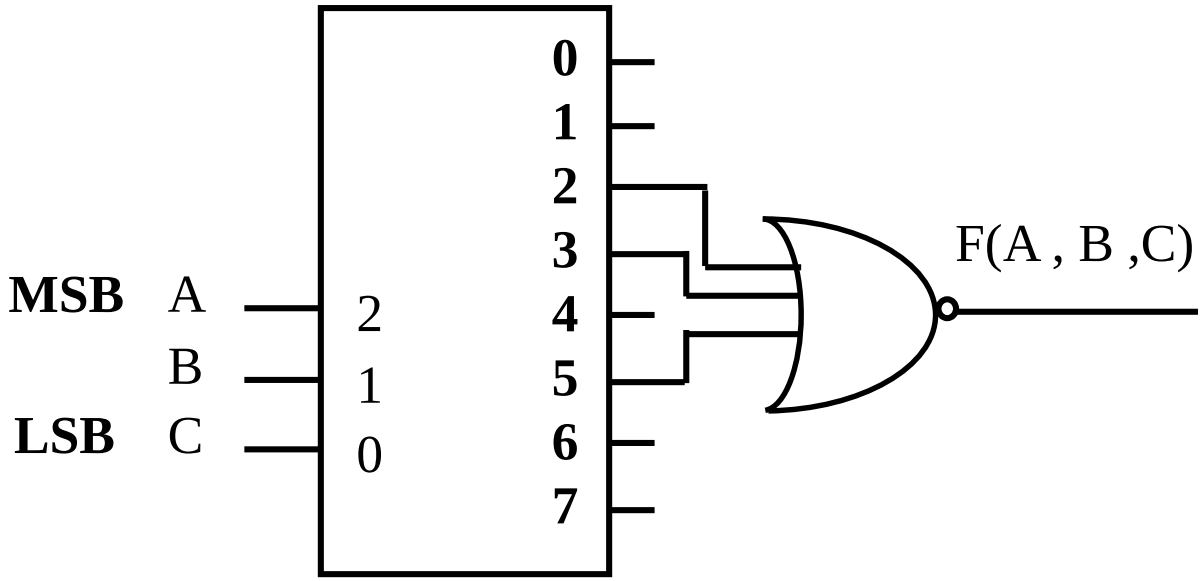


# پیاده سازی توابع منطقی با دیکدر ها

## یک دیکدر (با خروجی فعال بالا) ویک گیت NOR بکار بریم.

$$F(A, B, C) = \sum m(0, 1, 4, 6, 7) = \prod M(2, 3, 5)$$

$$F(A, B, C) = \overline{m_2 + m_3 + m_5}$$

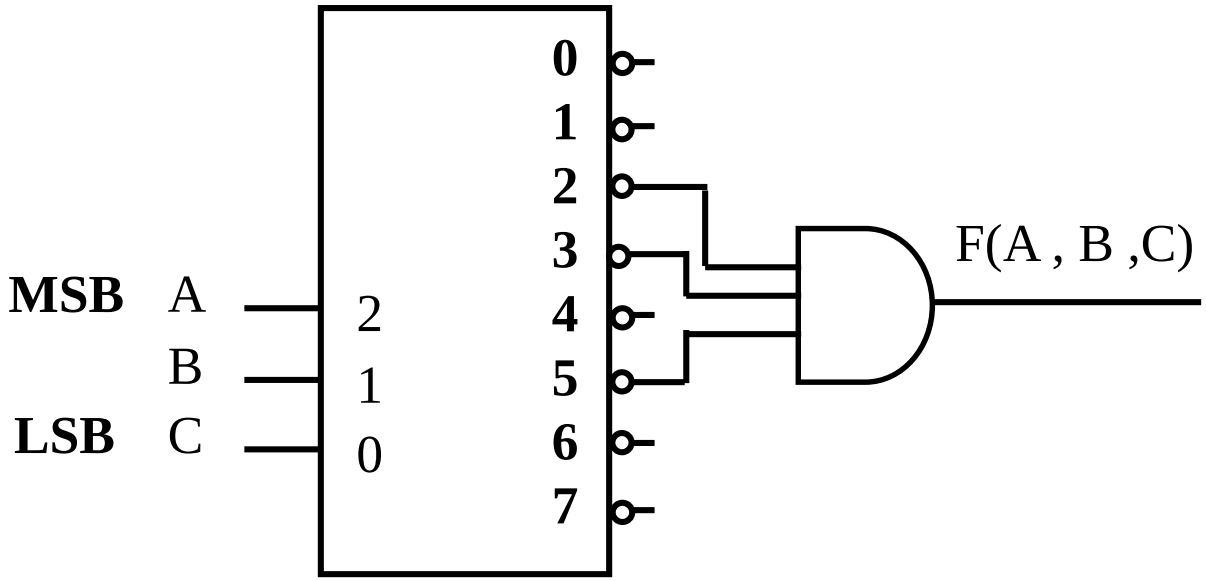


# پیاده سازی توابع منطقی با دیکدر ها

## یک دیکدر (با خروجی فعال پایین) ویک گیت AND بکار بریم.

$$F(A, B, C) = \sum m(0, 1, 4, 6, 7) = \prod M(2, 3, 5)$$

$$F(A, B, C) = \bar{m}_2 . \bar{m}_3 . \bar{m}_5$$

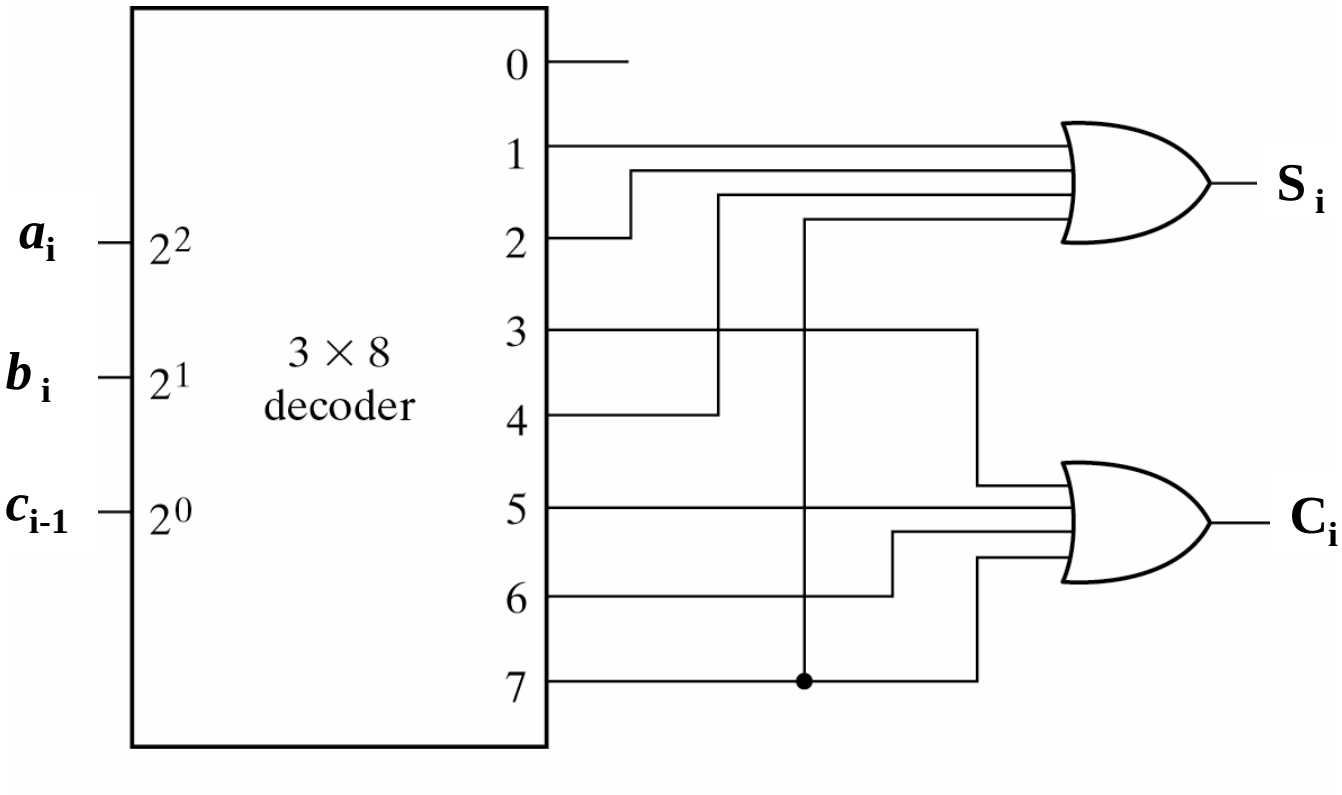


# پیاده سازی توابع منطقی با دیکدر ها

## ساختن Full Adder به وسیله دیکدر

### جدول درستی

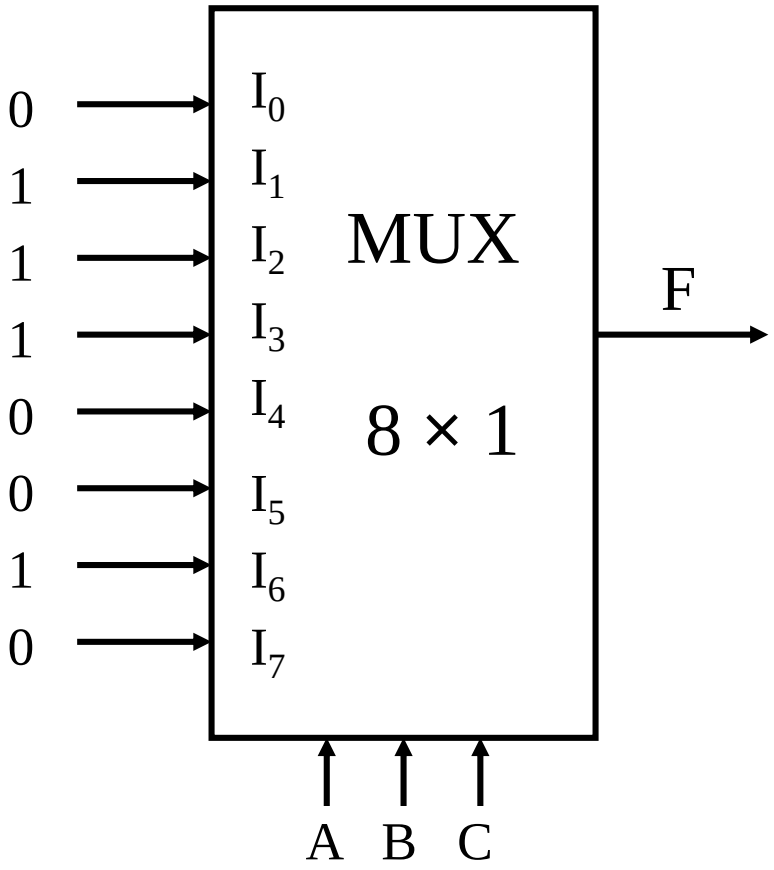
| $a_i$ | $b_i$ | $c_{i-1}$ | $C_i$ | $S_i$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |



# پیاده سازی تابع با مالتی پلکسر

$$F(A, B, C) = \sum m(1, 2, 3, 6)$$

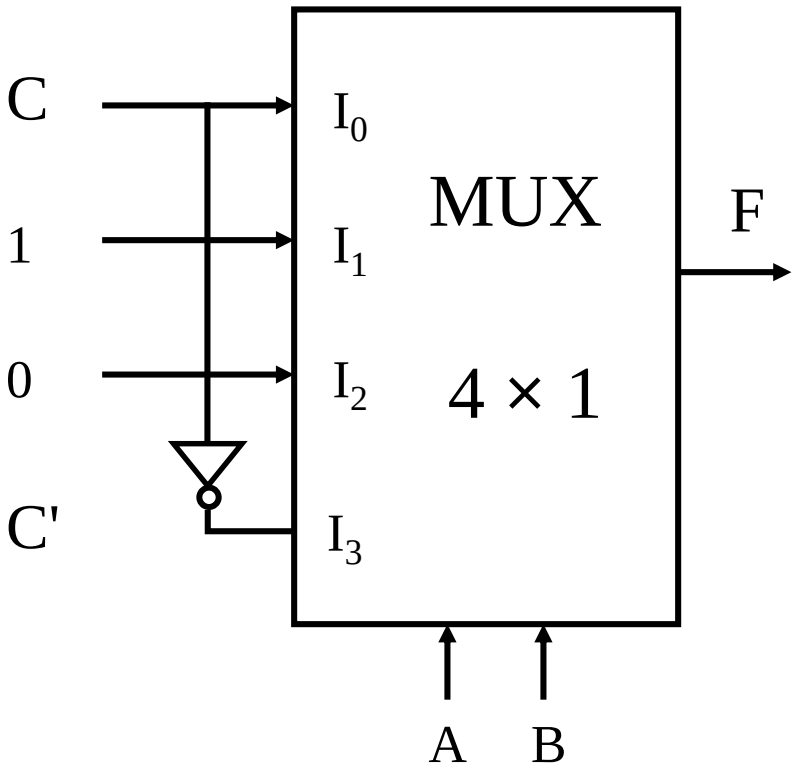
| A | B | C | F |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |



# پیاده سازی تابع با مالتی پلکسر

$$F(A, B, C) = \sum m(1, 2, 3, 6)$$

| | A | B | C | F | |
|---|---|---|---|---|---|
| $I_0$ | 0 | 0 | 0 | 0 | F=C |
| | 0 | 0 | 1 | 1 | |
| $I_1$ | 0 | 1 | 0 | 1 | F=1 |
| | 0 | 1 | 1 | 1 | |
| $I_2$ | 1 | 0 | 0 | 0 | F=0 |
| | 1 | 0 | 1 | 0 | |
| $I_3$ | 1 | 1 | 0 | 1 | F=C’ |
| | 1 | 1 | 1 | 0 | |



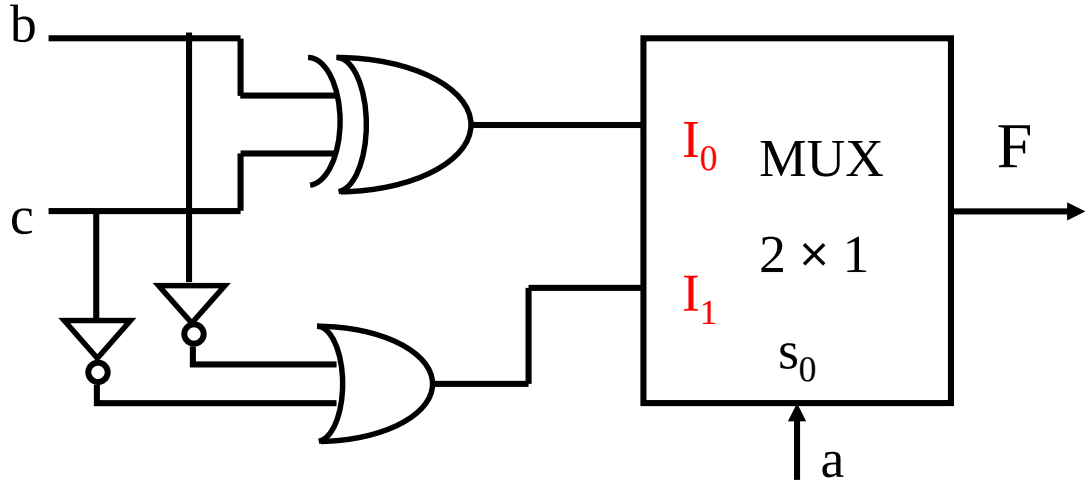
# پیاده سازی تابع با مالتی پلکسر

$F(A, B, C) = \sum m(1, 2, 4, 5, 6)$

| | a | b | c | F |
|---|---|---|---|---|
| $I_0$ | 0 | 0 | 0 | 0 |
| | 0 | 0 | 1 | 1 |
| | 0 | 1 | 0 | 1 |
| | 0 | 1 | 1 | 0 |
| $I_1$ | 1 | 0 | 0 | 1 |
| | 1 | 0 | 1 | 1 |
| | 1 | 1 | 0 | 1 |
| | 1 | 1 | 1 | 0 |

| a \ b c | 00 | 01 | 11 | 10 | |
|---|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 1 | $I_0$ |
| 1 | 1 | 1 | 0 | 1 | $I_1$ |

$$\begin{cases} I_0 = b \oplus c \\ I_1 = \bar{b} + \bar{c} = \overline{bc} \end{cases}$$

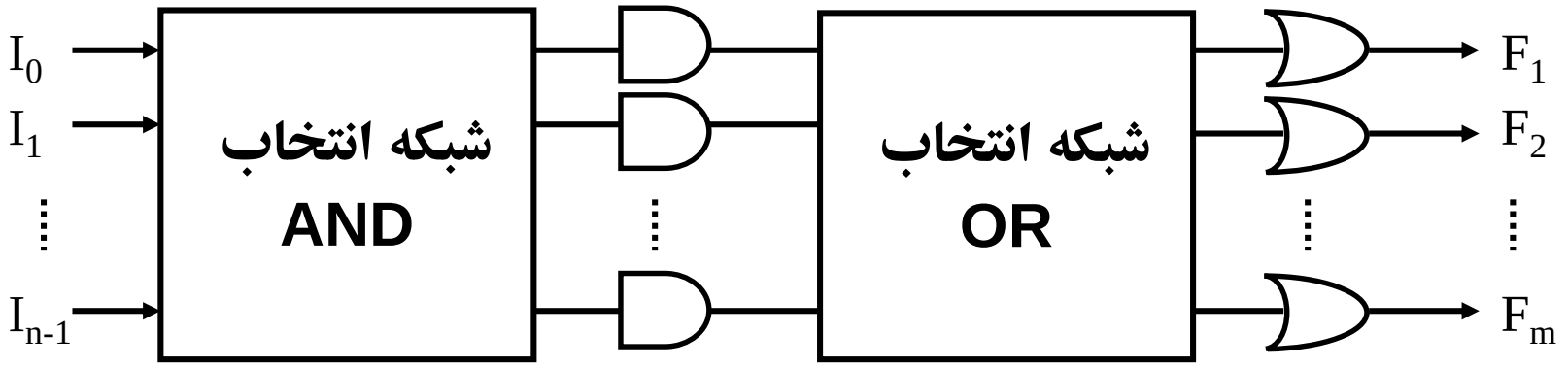


# ابزارهای قابل برنامه ریزی

ابزارهای قابل برنامه ریزی با توجه به شکل استاندارد پیاده سازی توابع منطقی از فرم دوطبقه AND-OR استفاده می کنند.

برای ساخت ابزاری که بتواند با استفاده از این شکل استاندارد هر تابع منطقی را پیاده سازی کند، ابزار ساخته شده باید قابلیت انتخاب ورودیها برای ایجاد ترمهای مورد نظر را داشته باشد و یا از بین ترمهای موجود بتواند ترمهای مربوط به هر تابع را انتخاب کند. (به این انتخاب ها اصطلاحا برنامه ریزی هم گویند)

شکل کلی این ابزار بصورت زیر خواهد بود:



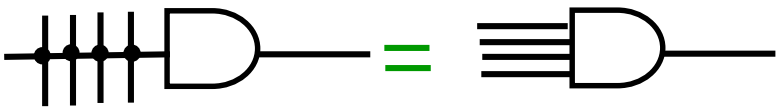
# ابزارهای قابل برنامه ریزی

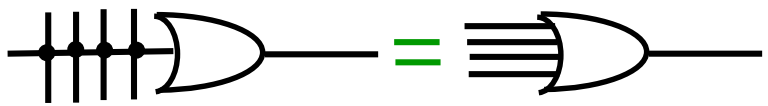
سه نوع ابزار ساده قابل برنامه ریزی برای پیاده سازی توابع وجود دارد که هرکدام دارای قابلیت های مختص به خود هستند.

این ابزارها به همراه بخش قابل برنامه ریزی آنها در جدول زیر آورده شده است.

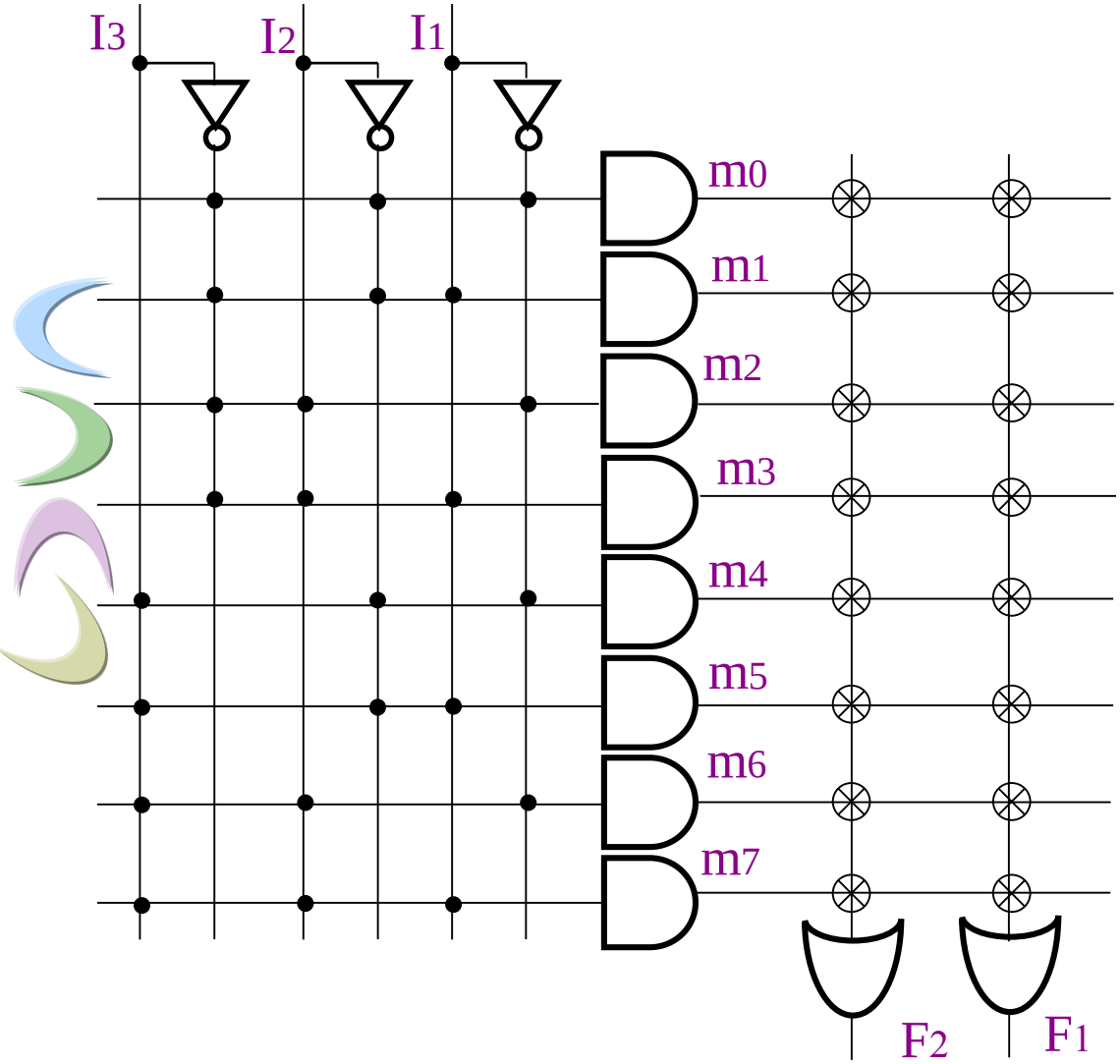
| ابزار | ماتریس AND | ماتریس OR |
|---|---|---|
| PROM | ثابت | برنامه پذیر |
| PAL | برنامه پذیر | ثابت |
| PLA | برنامه پذیر | برنامه پذیر |

در این ابزارها داریم:





# ابزارهای قابل برنامه ریزی




## ساختار یک PROM $2^3 \times 2$

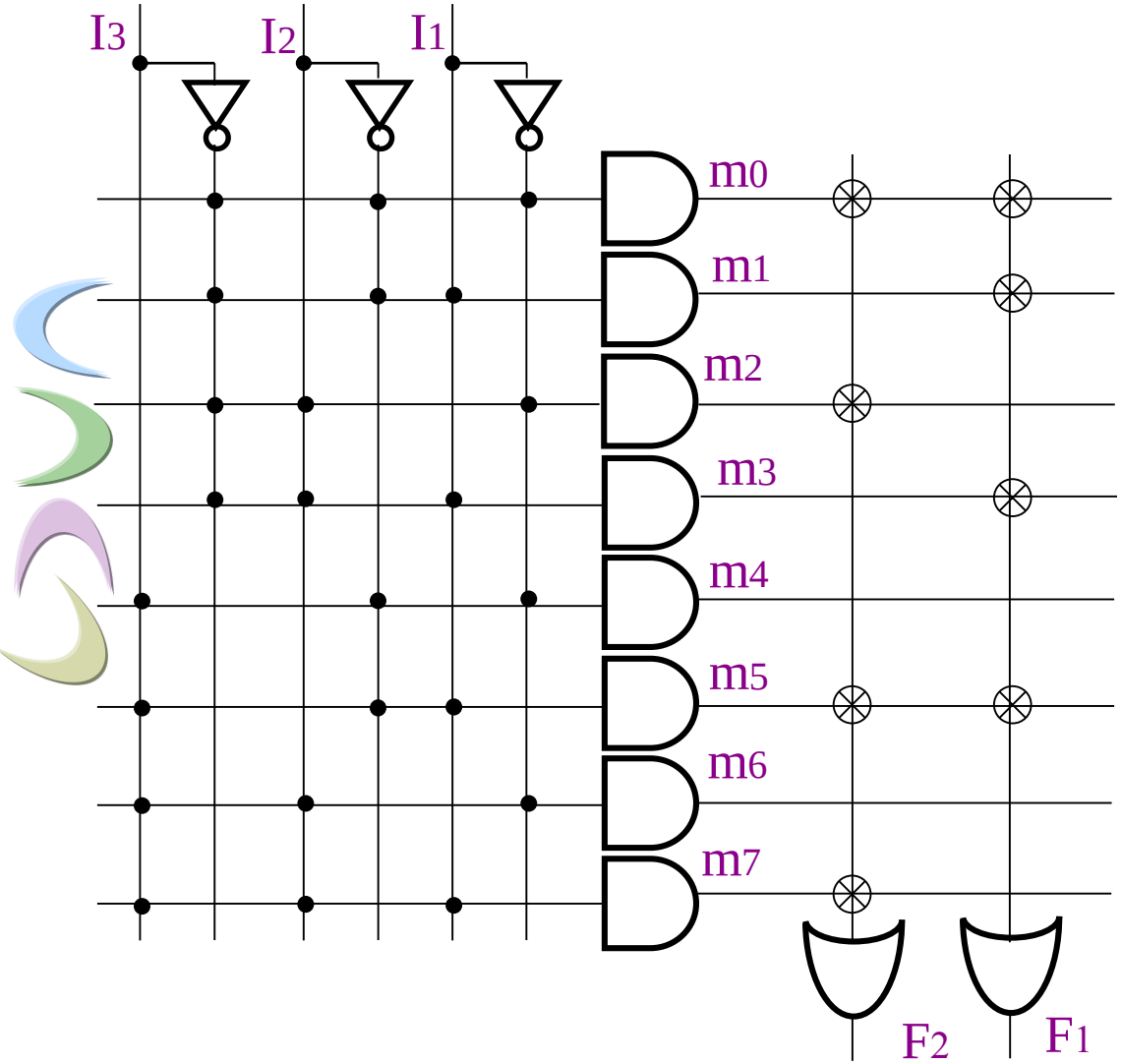
این ابزار دارای ۳ ورودی و ۲ خروجی می باشد.

تعداد AND ها برابر تعداد کل مینترمهای مربوط به ورودیها می باشد.

خروجی هر AND یکی از مینترمهای می باشد که اتصالات آن توسط کارخانه ثابت شده و قابل تغییر نمی باشد.

اتصالات مربوط به انتخاب مینترمها برای هر تابع قابل برنامه ریزی و تغییر می باشد

# ابزارهای قابل برنامه ریزی




توابع زیر را با استفاده از یک PROM $2^3 \times 2$ پیاده سازی کنید.

$$F_1(A,B,C) = \sum m(0,1,3,5)$$

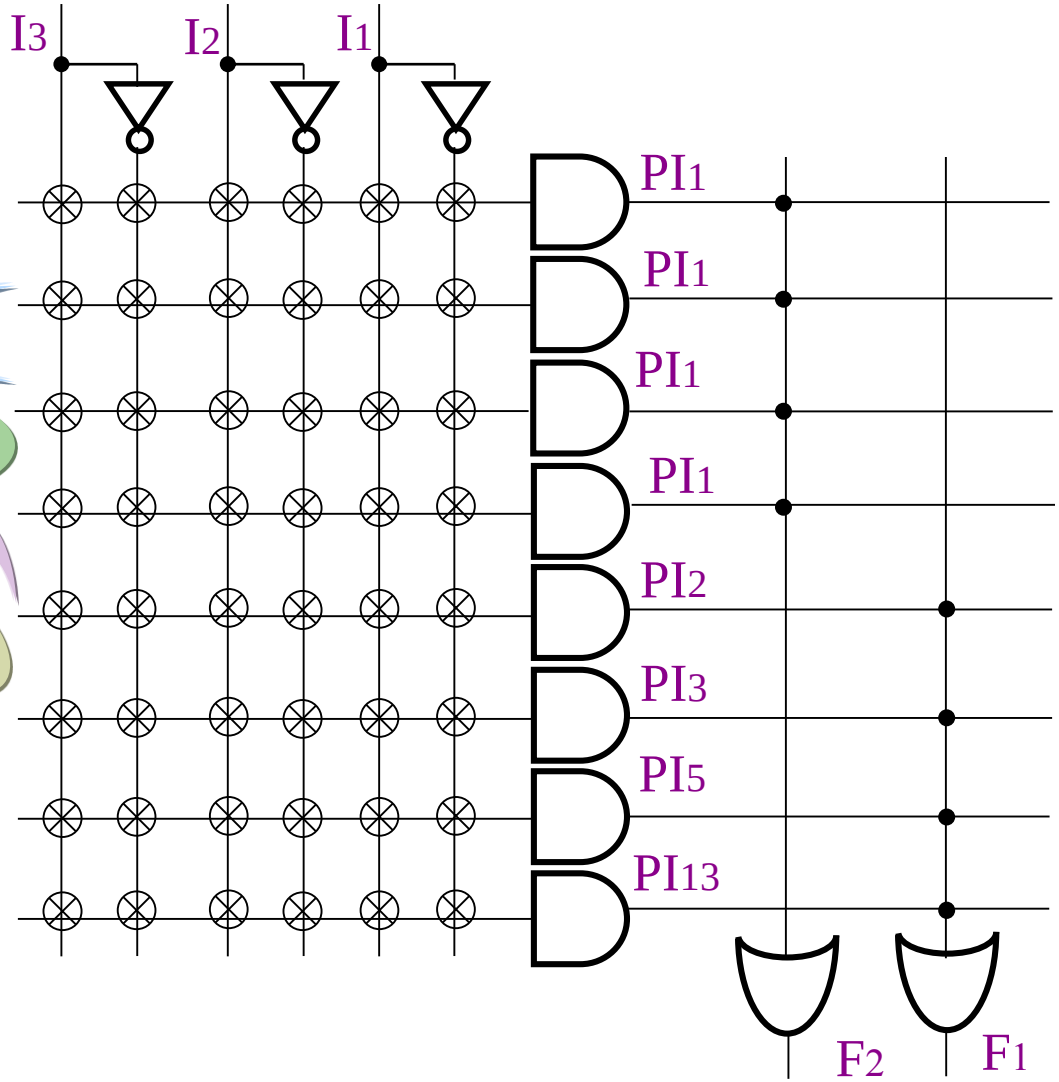
$$F_2(A,B,C) = \sum m(0,2,5,7)$$

باید متغیرهای دو تابع یکسان باشند.

ترمهای توابع می تواند مشترک باشند و هر دو از آنها اشتفاده کنند.

برخی ترمها ممکن است اصلا استفاده نشود.

# ساده سازی برای سیستم های چند خروجی




## ساختار یک PAL 3×2×4

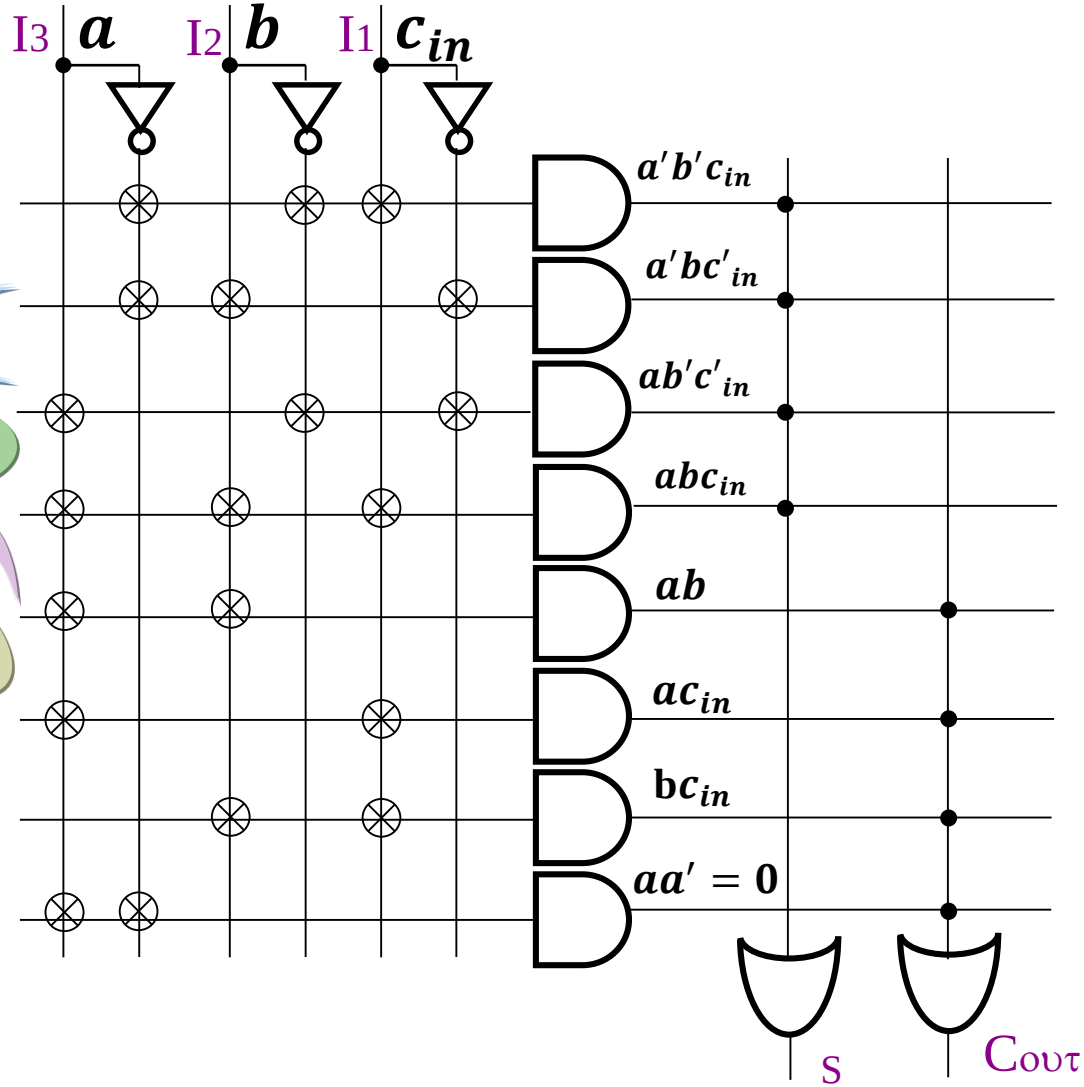
این ابزار دارای ۳ ورودی و ۲ خروجی و ۴ ترم برای هر خروجی می باشد.

تابع باید به نحوی ساده گردد که تعداد ترمهای آن حداکثر ۴ ترم گردد.

خروجی هر AND قابل برنامه ریزی بوده و می تواند هر ترکیبی از ورودیها باشد (حتی یکی از مینترمها).

اتصالات قسمت OR آن توسط کارخانه ثابت شده و قابل تغییر نمی باشد.

# ساده سازی برای سیستم های چند خروجی




مدار یک تمام جمع کننده را با استفاده از یک PAL 3×2×4 پیاده سازی کنید.

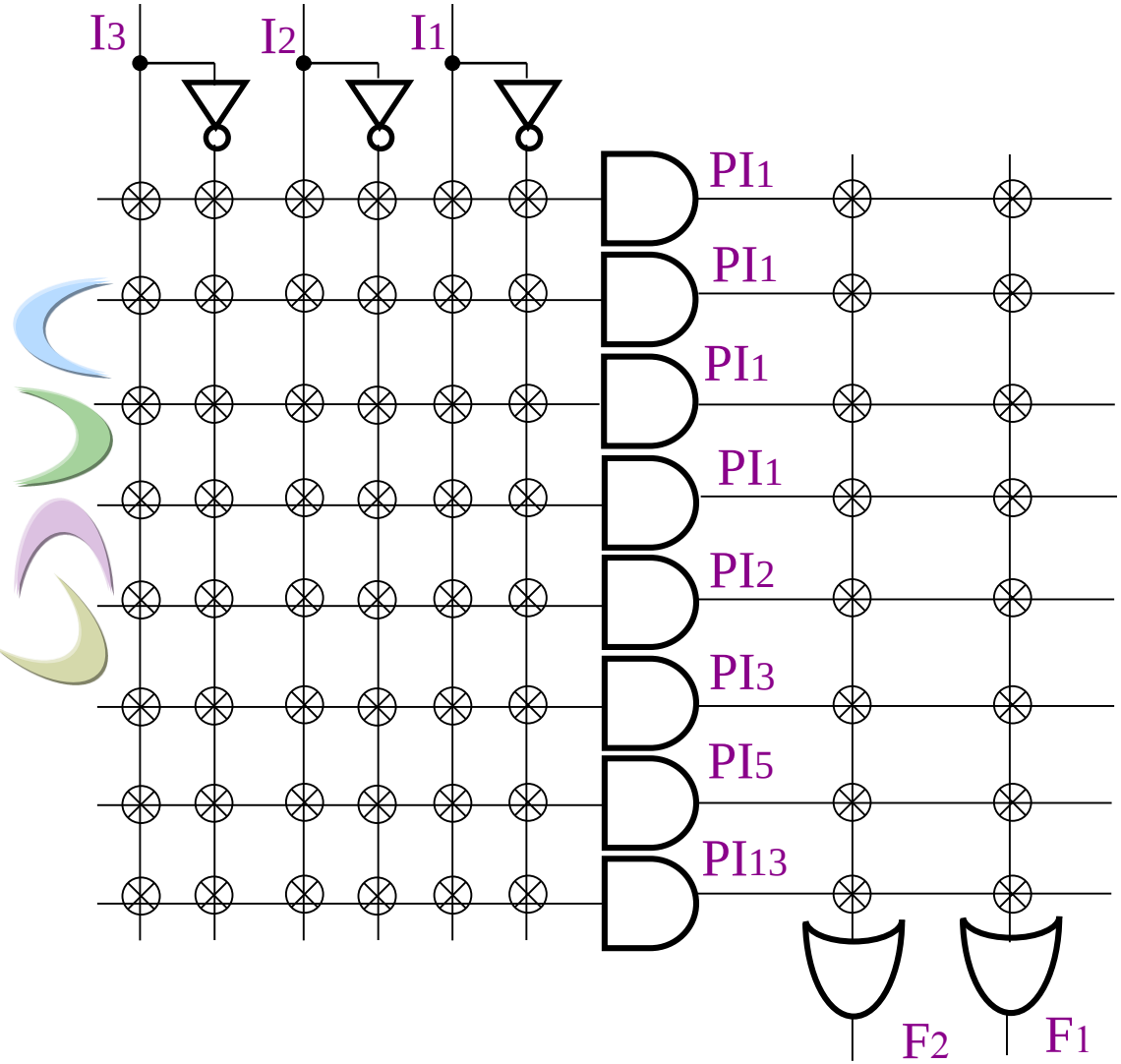
$$S(a,b,c_{in}) = a'b'c_{in} + a'bc'_{in} + ab'c'_{in} + abc_{in}$$

$$C_{out}(a,b,c_{in}) = ab + ac_{in} + bc_{in}$$

باید متغیرهای دو تابع یکسان باشند.

اگر تعداد ترمهای توابع کمتر از ۴ ترم باشد به ازای هر ترم کمتر باید یک ترم xx' (مقدار بی تاثیر صفر) استفاده نمود..

# ساده سازی برای سیستم های چند خروجی




## ساختار یک PLA 3×2×8

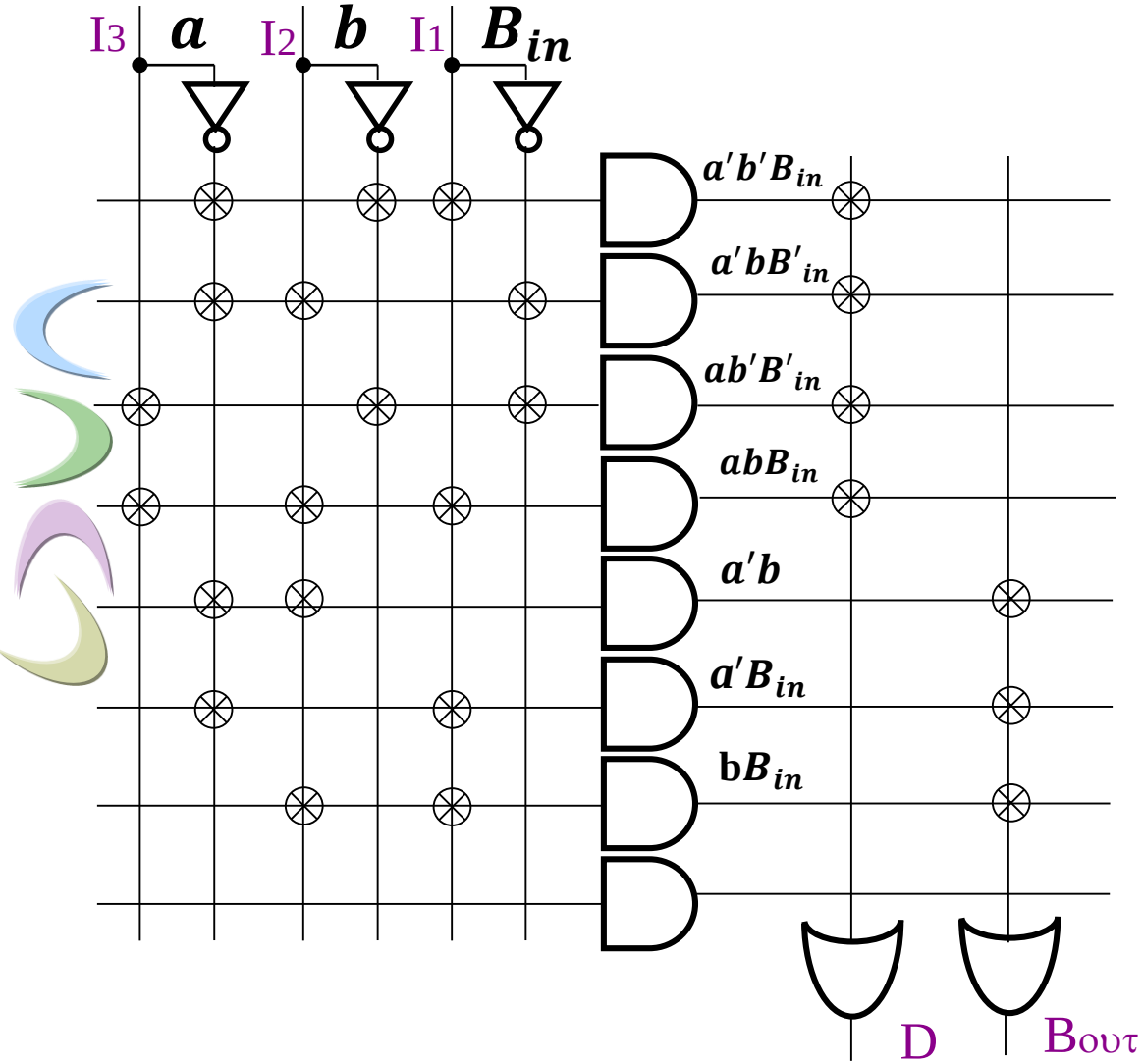
این ابزار دارای ۳ ورودی و ۲ خروجی و ۸ ترم یا AND برای همه خروجی ها می باشد.

تابع با هم باید به نحوی ساده گردند که تعداد ترمهای آنها حداکثر ۸ ترم گردد.

خروجی هرAND قابل برنامه ریزی بوده و می تواند هر ترکیبی از ورودیها باشد (حتی یکی از مینترمها).

اتصالات قسمت OR هم قابل برنامه ریزی و تغییر می باشد. حتی ترمهای مشترک می توان داشت.

# ساده سازی برای سیستم های چند خروجی




مدار یک تمام تفریق کننده را با استفاده از یک PLA 3×2×8 پیاده سازی کنید.

$$D(a,b,B_{in}) = a'b'B_{in} + a'bB'_{in} + ab'B'_{in} + abB_{in}$$

$$B_{out}(a,b,B_{in}) = a'b + a'B_{in} + bB_{in}$$

باید متغیرهای دو تابع یکسان باشند.

اگر تعداد کل ترمهای توابع می تواند کمتر از ۸ ترم باشد و نیازی به اضافه کردن ترم xx' (مقدار بی تاثیر صفر) نیست.