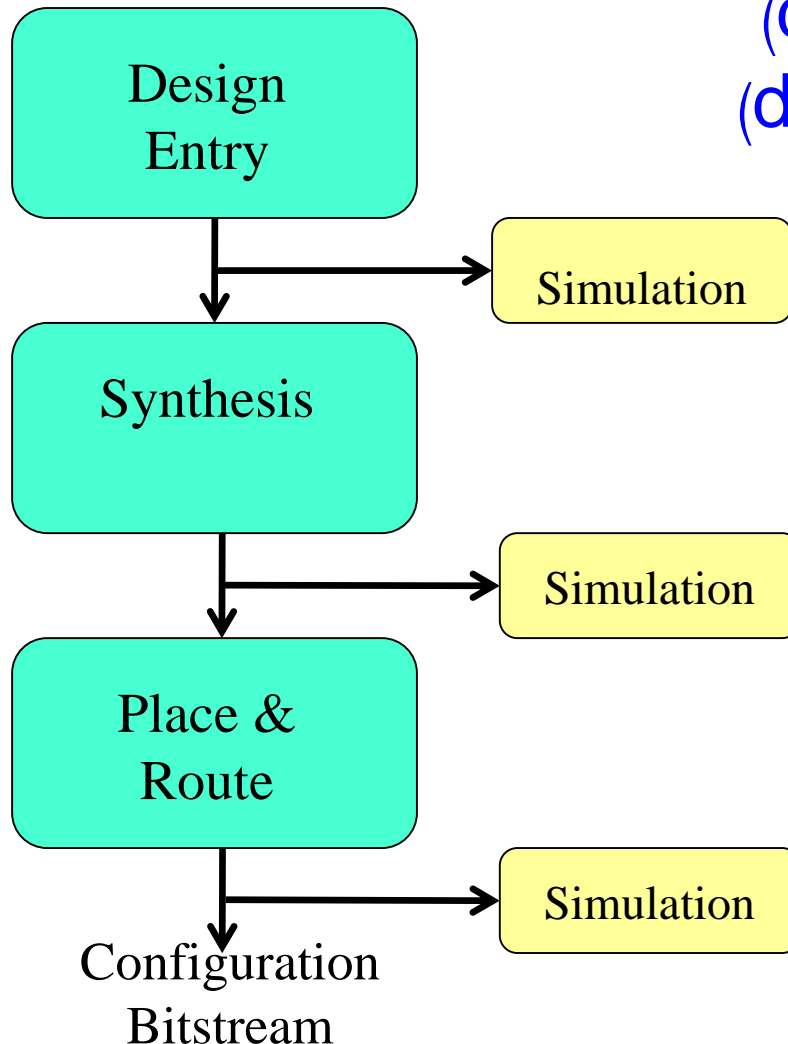


# چرخه طراحی با PLD

# چرخه طراحی

□ ورود طرح (design entry)  
یا توصیف آن (description)



- جدول درستی

- شماتیک

- شکل موج

- دیاگرام حالت

- زبان توصیف سخت افزار

# چرخه طراحی

```
process(CUR, D, P)
begin
  case CUR is
    when START =>
      if (D, P) = "00" then
        NEXT <= START;
      elsif (D, P) = "01" then
        NEXT <= FULL;
      elsif (D, P) = "10" then
        NEXT <= P200;
      end if;
    when P200 =>
      if (D, P) = "00" then
        NEXT <= P200;
      elsif (D, P) = "01" then
        NEXT <= FULL;
      elsif (D, P) = "10" then
        NEXT <= P400;
      end if;
    when P400 =>
      if (D, P) = "00" then
        NEXT <= P400;
      else
        NEXT <= FULL;
      end if;
    when FULL =>
      NEXT <= START;
  end case;
end process;
```

## • زبان توصیف سخت افزار

پردازش آسان ☐

استاندارد ☐

VHDL-

Verilog-

SystemC-

AHDL-

...

# چرخه طراحی

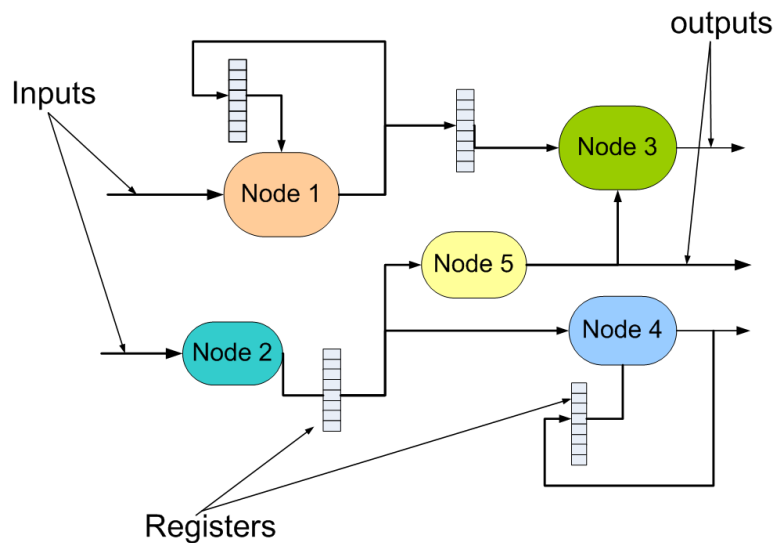
• سنتز:

□ تولید مدار

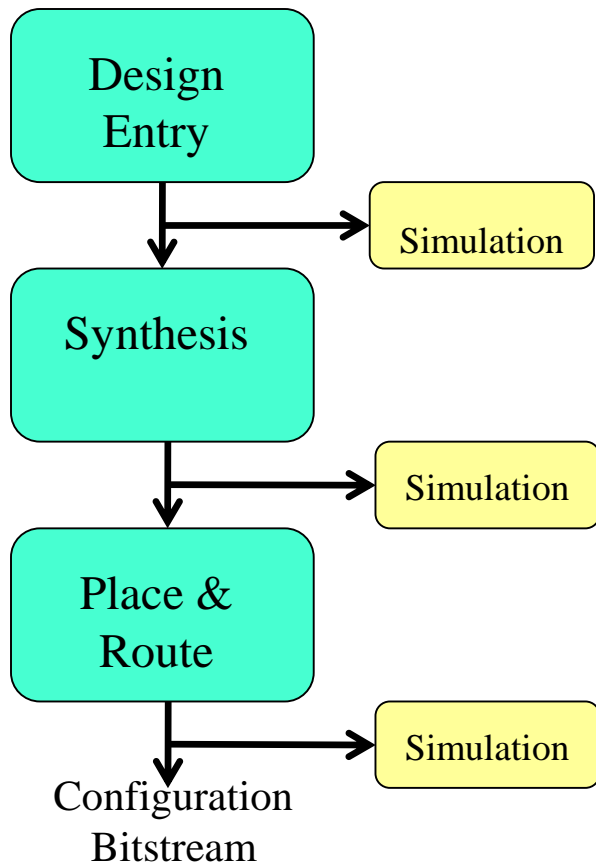
- فلیپ فلاپها + معادلات بولین  
مدارهای ترکیبی

□ نگاشت فناوری ( technology  
(mapping

□ بهینه سازی مدار



# چرخه طراحی



• سنتز:

□ تولید مدار

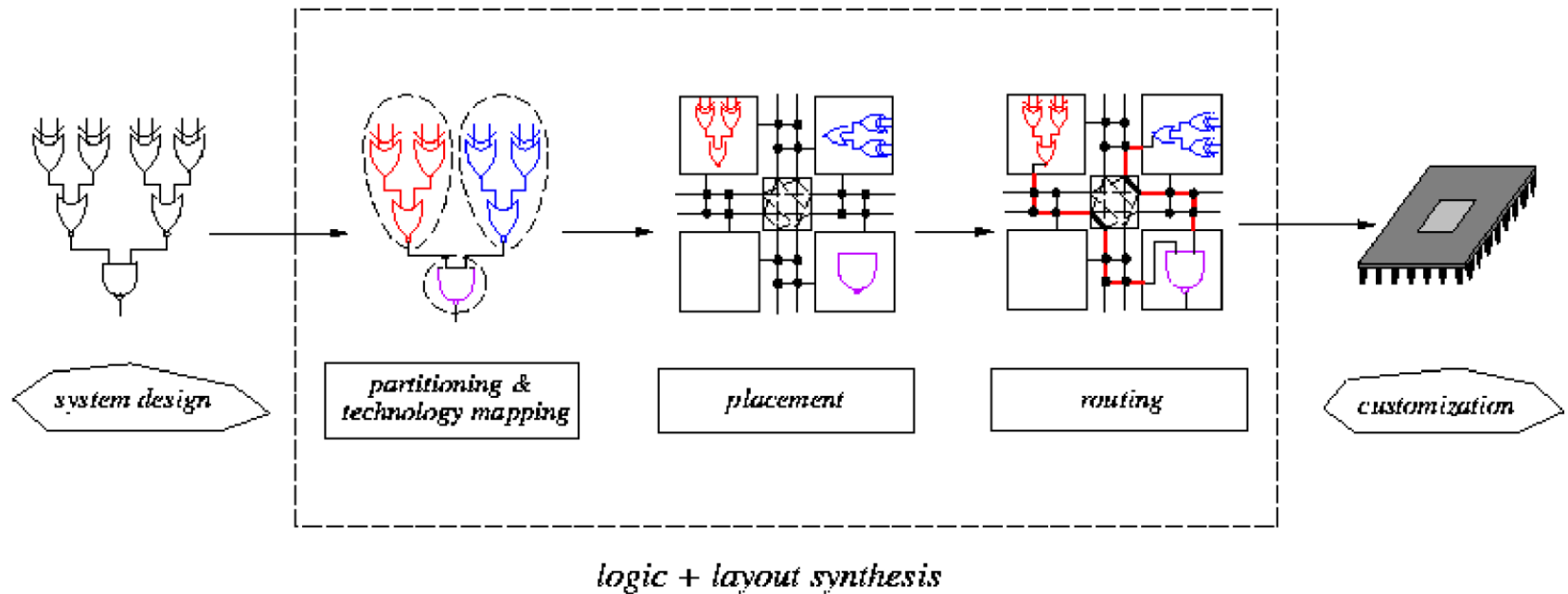
- فلیپ فلاپها + معادلات بولین  
مدارهای ترکیبی

□ نگاشت فناوری ( technology  
(mapping

- تبدیل معادلات بولین به منابع  
سختافزاری موجود در تکنولوژی

□ بهینه‌سازی مدار

# سنتز و نگاشت فناوری

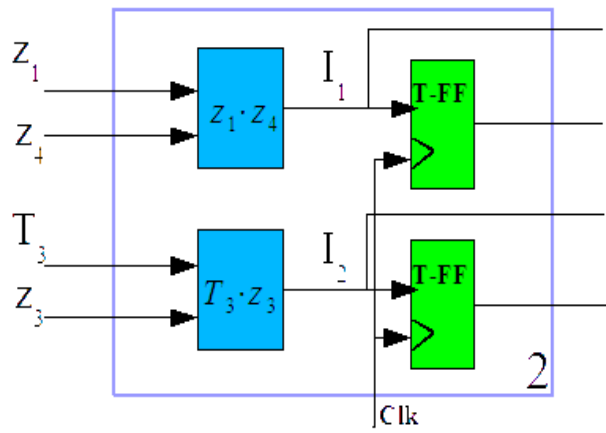
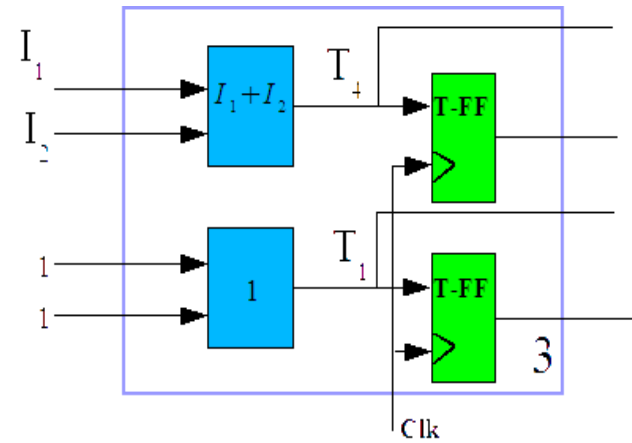
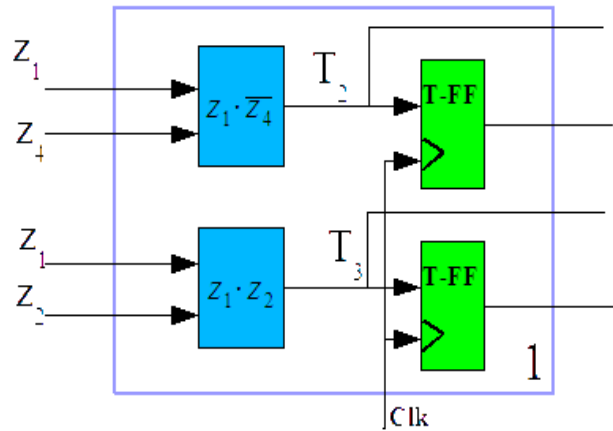


## • خروجی سنتز:

□ نت لیست مدار بر حسب اجزای موجود در تراشه

– چه اجزایی و نحوه اتصال آنها (به هم و به پورتها)

# خروجی ابزار سنتز



Common product term

$$T_1 = 1$$

$$T_2 = z_1 \cdot \overline{z_4}$$

$$T_3 = z_1 \cdot z_2$$

$$T_4 = z_1 \cdot z_4 + z_1 \cdot z_2 \cdot z_3$$

Arrows point from the text "Common product term" to the circled terms  $z_1 \cdot z_2$  in the equations for  $T_3$  and  $T_4$ .

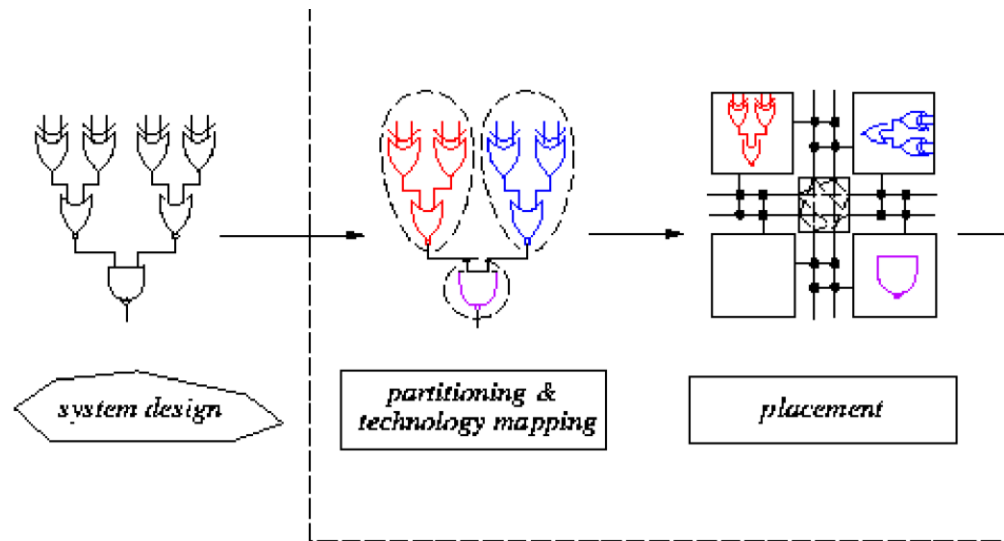
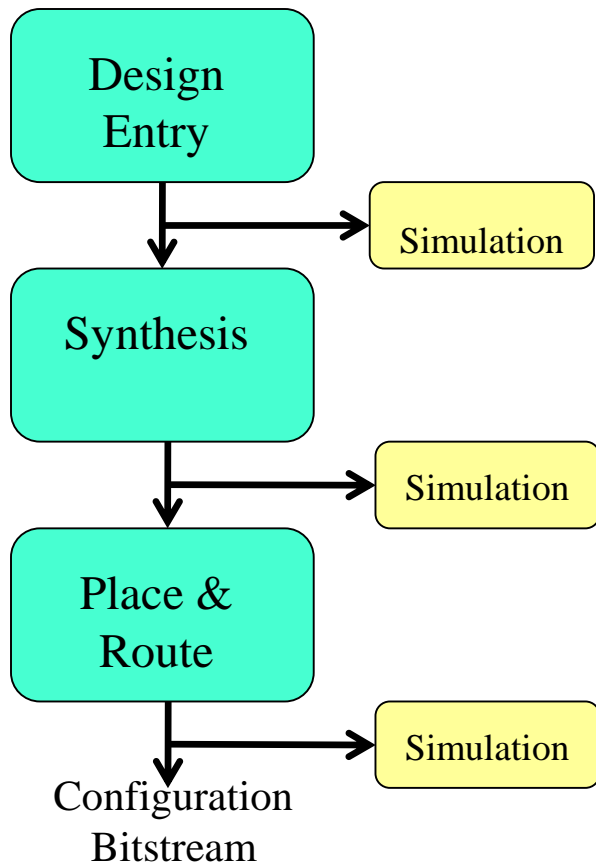
# چرخه طراحی

## • جایابی (placement):

□ اختصاص محل به اجزای مدار

- تخصیص بلوک‌های منطقی به  
اجزای مدار

- تلاش برای چیدمان بهینه



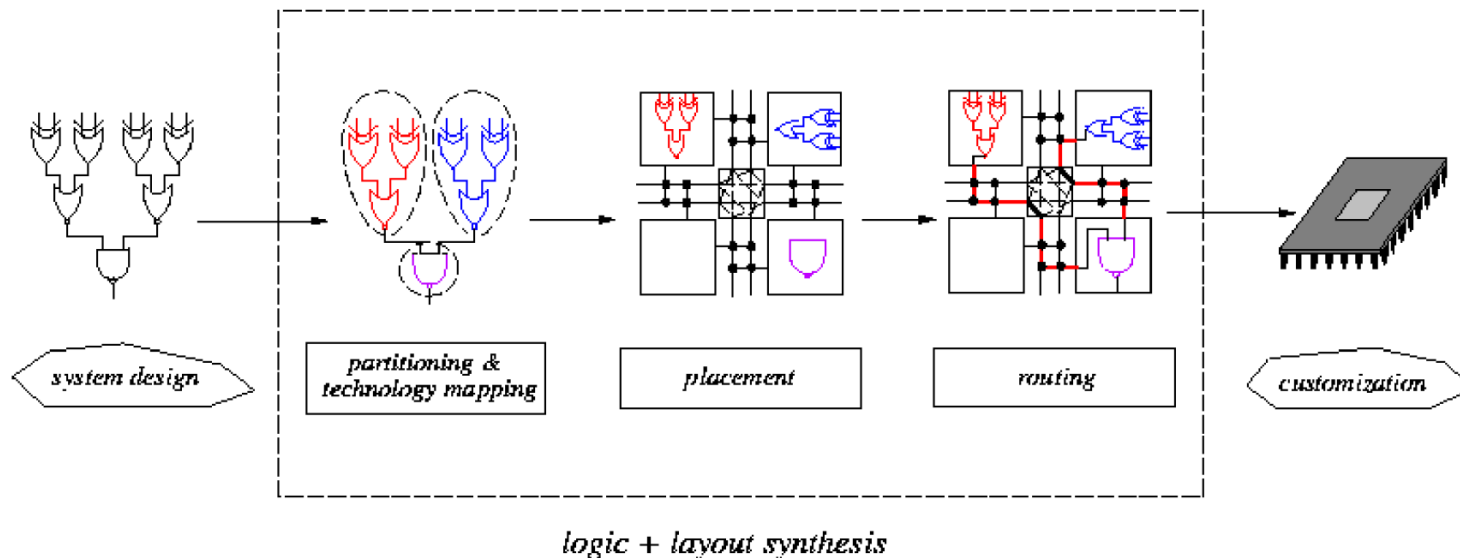


# چرخه طراحی

## • مسیریابی (routing):

□ تعیین مسیر برای اتصالات

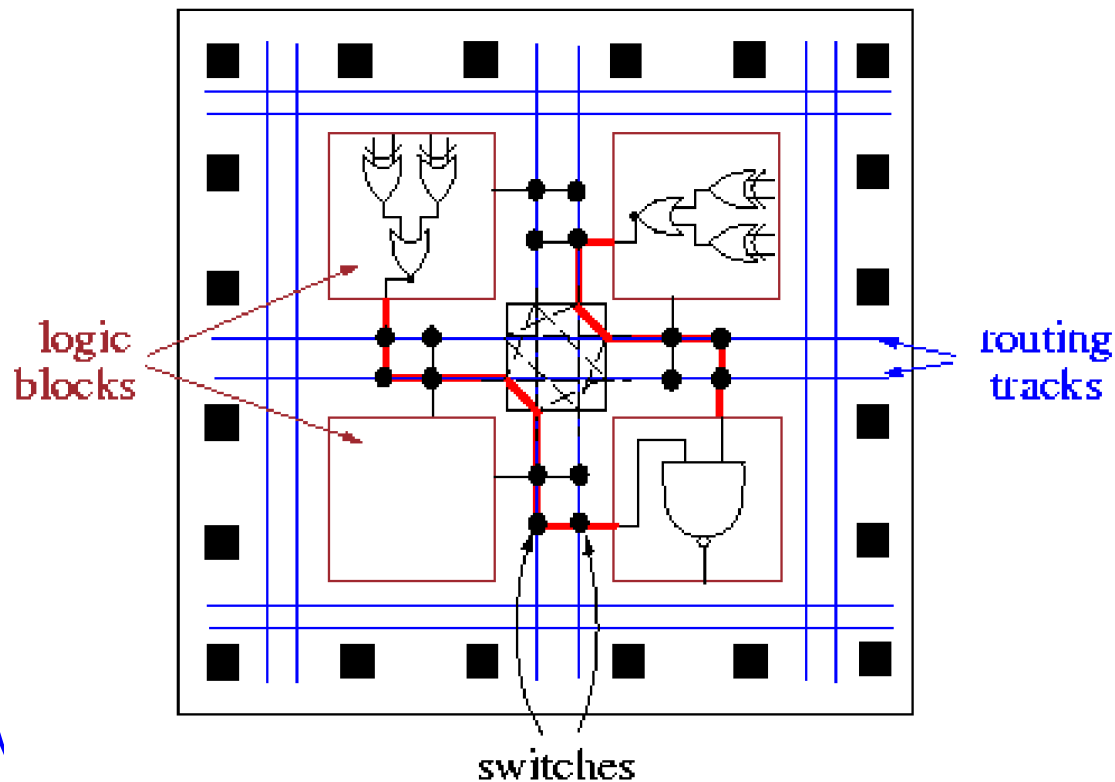
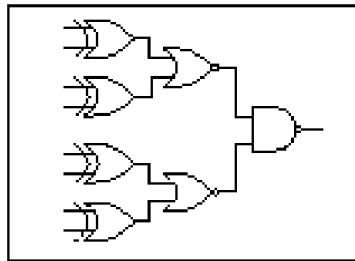
- تعیین قطع یا وصل بودن سویچها
- تلاش برای تکمیل مسیریابی همه اتصالات



# چرخه طراحی

• مسیریابی  
:(routing)

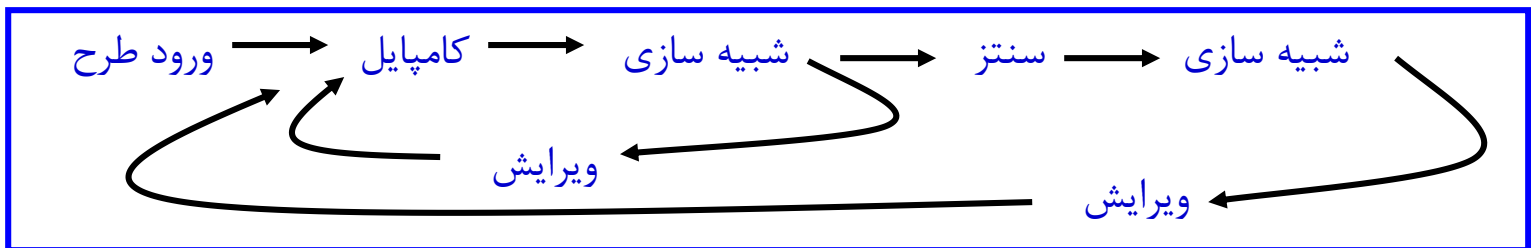
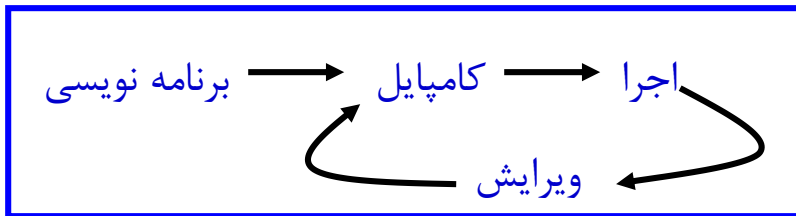
□ خروجی مسیریابی:  
bitstream



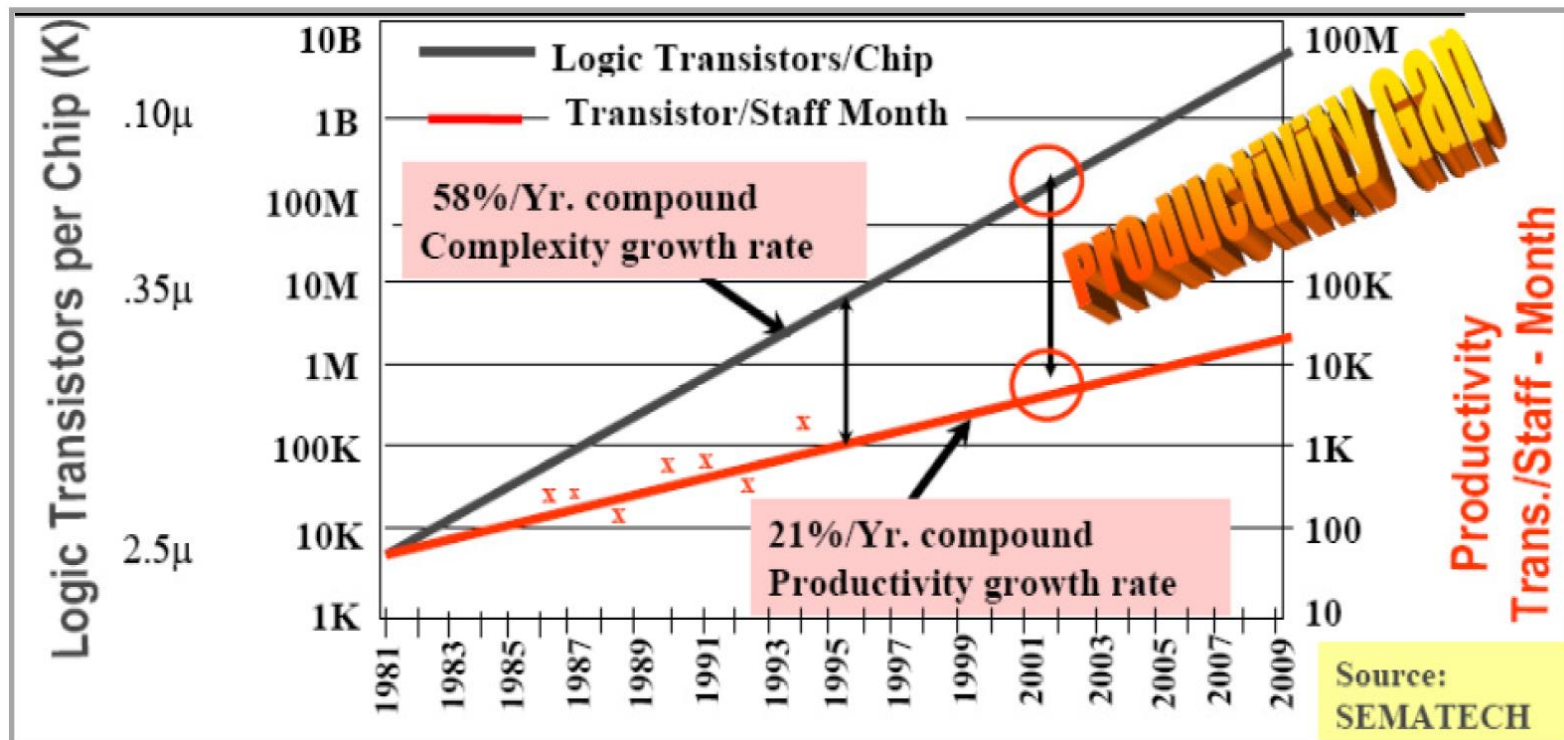
# چرخه طراحی

## • چرخه طراحی:

□ نیاز به تکرار زیاد



# Productivity Gap



• تکنولوژی ساخت:

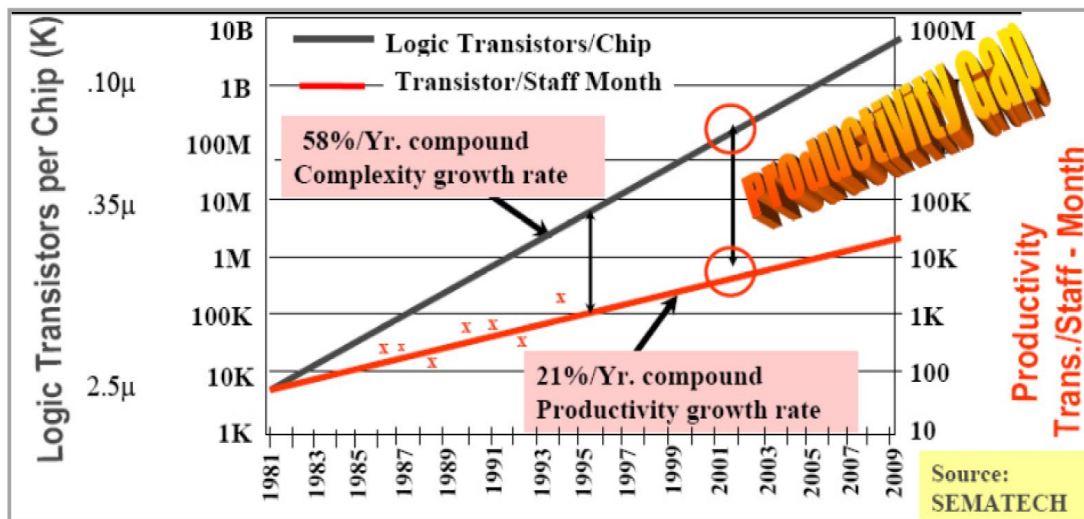
□ قانون مور: هر ۱۸ تا ۲۴ ماه: مقدار سخت افزار در واحد سطح: دو برابر

• توانایی طراحان:

• محورهای عمودی: لگاریتمی

□ سرعت پیشرفت کمتر

# Productivity Gap



• اکنون:

- ☐ ۳ میلیارد ترانزیستور
- ☐ ۱۵۰۰ نفر برای ۴۰ ماه
- ☐ بهره‌وری: ۵۰,۰۰۰ ترانزیستور بر نفر-ماه

• اوایل دهه ۱۹۸۰:

- ☐ ۱۰,۰۰۰ ترانزیستور
- ☐ ۱۰۰ نفر-ماه
- ☐ بهره‌وری: ۱۰۰ ترانزیستور بر نفر-ماه

- پیچیدگی تراشه: ۳۰۰,۰۰۰ برابر
- بهره‌وری: ۵۰۰ برابر

# Productivity Gap

- منحنی واقعی تر: زیگزاگ

