

۱- زبان وریلاگ چیست

۲- پیاده سازی وریلاگ برای چه مسایلی است

۳- مدار جمع کننده را ترسیم کنید

۴- مدار منطقی تابع pl_a را شرح دهید

۵- ایرادهای pl_a را بیان کنید

۶- یک شمارنده با فلیپ فلاپ d طراحی کنید

۷- قوانین برنامه نویسی وریلاگ را بیان کنید

۸- سطوح سیگنال در وریلاگ را بیان کنید

۹- انواع متغیرها در وریلاگ را بیان کنید

۱۰- خواص متغیرهای وریلاگ را بیان کنید

-۱

★ زمان و ریلای : توصیف سخت افزار و کاربرد سخت افزار FPGA

-۲

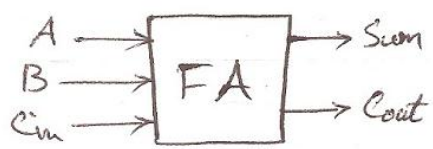
★ کاربرد ها : ۱- کاربرد های نظامی (رزانه ، مکانیزم سنسور تلفنی)

۲- پروژه های شامل چند نوع پردازش مختلف

★ مثال (پردازش تصویر ، ارتباط صافرات و ماهواره ای ، ارتباط سیستم ها ،

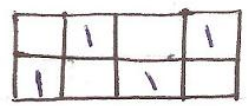
تنظیم سوخت ، سرعت ، فشار خون ، ... محاسباتی که نیاز به یک پردازنده مرکزی دارند :

★ جمع کننده (Full Adder) :



C_{in}	B	A	Sum
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

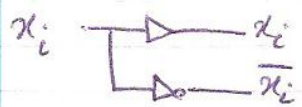
C_{in}	B	A	C_{out}
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



$Sum = A \oplus B \oplus C_{in}$

$C_{out} = AB + BC + AC$

★ برای ساخت اصل SOP :

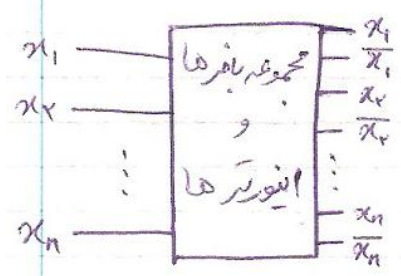


۱- ساخت دردی جا , invert شده آری

۲- ساخت product ها با آرایه AND

۳- ساخت sum ها با آرایه OR

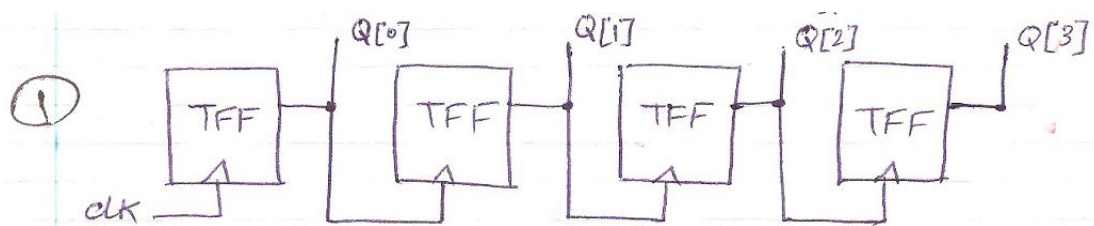
۴- سرچش های قابل برنامه ریزی



-۵

- ★ ابزارهای عمده PLA : ۱- دو طبقه سوئیچ قابل برنامه ریزی (توان مصرفی بالا)
- ۲- تنها مدارهای منطقی ترکیبی را می توان پیاده سازی کرد.
- ۳- سوئیچ ها از جنس فوسر بودند و تنها یکبار program می شدند.
- ۴- خروجی PLA ها قابل استفاده مجدد نبود.

-۶



-۷

- ۱- حروف کوچک و بزرگ دارای معانی متفاوت در ورلداگ هستند.
- ۲- اسم گذاری باید کامل باشد و در تعداد حروف محدودیتی ندارد.
- ۳- نامگذاری خالی در زبان ورلداگ بین $\backslash n$ ، $\backslash t$ ، $\backslash b$ تعدادی وجود ندارد.
- ۴- کامنت های تک خطی (//) ، چند خطی (/* -- */) برای توضیح دادن کد داریم.

-۸

0 ← سطح Low در منطق دیجیتال / تعبیر غلط بودن یک عبارت منطقی
 1 ← سطح High در منطق دیجیتال / تعبیر درست بودن یک عبارت منطقی
 X ← متغیر دارای مقدار است اما مقدار آن برای ما مهم نیست. / به معنای don't care هم هست
 Z ← به معنای high impedance است. / اگر به متغیرهای خارجی wire مقدار داده نشود Z فرض می شود.

-9

net ★ : فقط برای اتصالات فیزیکی بین الیاف فیزیکی استفاده می شود.

باید دارای درایو کننده فیزیکی باشد.
 خروجی گیت منطقی باید چنین wire تعریف شود.

wire - wand - wor - tri - triand - trior

trireg - supply - supply | - tri - tri

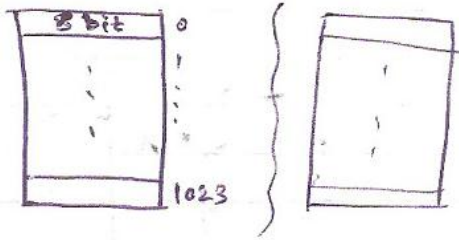
← قابل سنتز نیست.

-10

1- تعریف بصورت برداری `wire [31:0] DataBus;` 

2- قابلیت حفظ اطلاعات را ندارد. در هر لحظه مقدار خود را از یک درایو کننده فیزیکی قبل از خود دریافت می کند.

wire [7:0] Bus [0:1023];



۳- تعریف صورت آرایه

است.