بسم الله الرحمن الرحیم

ریز پردازنده

کارشناسی سخت افزار ترم ۲

مهدیه نیری

عملگرها

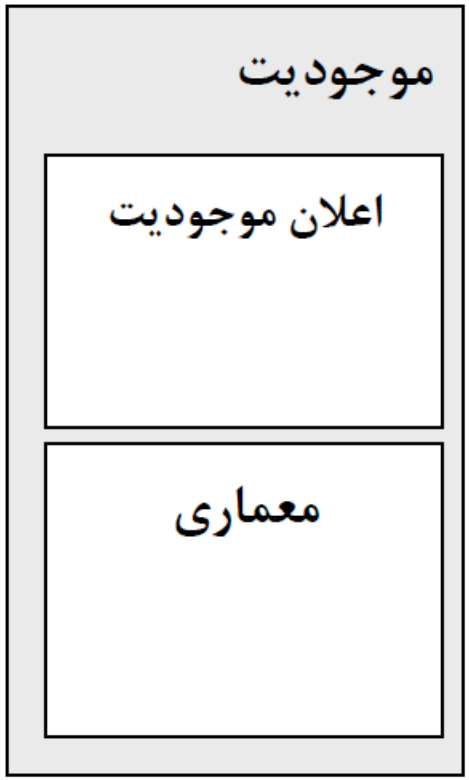
- عملگرهای vhdl در جدول زیر بر اساس تقدم رده بندی شده اند.

	رده عملگر	عملگر
بالاترین تقدم	متفرقه	**, ABS, NOT
	ضرب	*, /, MOD, REM
	علامت	+, -
	جمع	+, -, &
	مقایسه ایی	=, /=, <, <=, >, >=
پایینترین تقدم	منطقی	AND, OR, NAND, NOR, XOR, XNOR

موجودیت (ENTITY) طراحی در VHDL :

به یک مدار یا زیرمداری که توسط کد VHDL توصیف می شود ، موجودیت طراحی یا فقط موجودیت گفته می شود . در شکل ۶-۱ ساختار کلی یک موجودیت را نشان می دهد . این ساختار دارای دو بخش اصلی است : *اعلان* موجودیت که سیگنالهای ورودی و خروجی را برای موجودیت مشخص می کند و *معماری* که جزئیات مدار را می دهد .

ساختار کلی موجودیت در vhdl



شکل ۶-۱: ساختار کلی یک موجودیت در VHDL

اعلان یک موجودیت

- با استفاده از اعلان موجودیت ، سیگنالهای ورودی و خروجی در یک موجودیت مشخص می شود. نام موجودیت می تواند هر نام معتبر در vhdl باشد، کروشه های باز و بسته یک آیتم اختیاری را نشان می دهد. سیگنالهای ورودی و خروجی با استفاده از کلمه کلیدی port مشخص می شود. کلیه مدهای موجود در جدول اسلاید بعدی خلاصه شده اند. اگر مد یک پورت مشخص نشود به طور پیش فرض مد in برای پورت در نظر گرفته می شود.

Entity entity-name is

Port ([signal] signal_name {,signal _name}:[mode] type-name});

END entity_name

جدول انواع مدها برای سیگنالهایی که پورت موجودیت هستند.

مد	مفهوم
IN	برای سیگنالی که یک ورودی برای یک موجودیت است استفاده می شود
OUT	برای سیگنالی که یک خروجی از یک موجودیت است استفاده می شود . این بدین معنا است که در یک دستورالعمل انتساب ، سیگنال فقط در طرف چپ عملگر <= قرار می گیرد
INOUT	برای یک سیگنال که هم ورودی به یک موجودیت و هم یک خروجی از یک موجودیت است ، استفاده می شود
BUFFER	برای سیگنالی که یک خروجی از یک موجودیت است استفاده می شود . مقدار سیگنال می تواند درون موجودیت استفاده شود ، بدین معنا که در یک دستورالعمل انتساب ، سیگنال می تواند هم در طرف چپ و هم در طرف راست عملگر <= قرار گیرد

معماری (Architecture)

- یک معماری جزییات مدار را برای یک موجودیت فراهم می سازد. معماری دارای دو بخش اصلی است. ناحیه اعلان و بدنه معماری.
- ناحیه اعلان قبل از کلمه کلیدی begin واقع است. در این ناحیه سیگنالها ، انواع شمارشی و ثابتها اعلان می شوند.
- اعلان ترکیبات و ویژگیها نیز در این ناحیه صورت می گیرد. عملکرد موجودیت در بدنه معماری و بعد از کلمه کلیدی begin مشخص می شود، این عملکرد توسط دستور العملهایی که توابع منطقی در مدار را به روشهای متنوعی تعریف می کنند، مشخص می شود.

فرم کلی معماری

ARCHITECTURE *ARCHITECTURE_name* OF *ENTITY_name* IS
[SIGNAL *declarations*]
[CONSTANT *declarations*]
[TYPE *declarations*]
[COMPONENT *declarations*]
[ATTREBUTE *speclFications*]
BEGIN
{ COMPONENT *instantiation statement* ;}
{ CONCURRENT ASSIGNMENT *statement* ; }
{ PROCESS *statement* ; }
{ GENERATE *statement* ; }
END [*ARCHITECTURE_name*] ;

مطالب مطرح شده از ابتدای درس تا اینجا

- تا این مرحله یاد گرفتیم که
- چگونه نوع داده ها را قرار بدهیم و هر نوع داده ایی برای چه مواردی کاربرد دارد.
- برای نوشتن هر برنامه دو قسمت را باید کامل کنیم
- ۱- entity یا موجودیت که در آن پورت های ورودی و خروجی را مطرح می کنیم.
- ۲- Architecture یا معماری که عملکرد مدار را در داخل آن می نویسیم

کار های بعدی

- در جلسات آتی با نوشتن برنامه آشنا می شوید . برای یادگیری بهتر از گیت های منطقی ساده شروع می کنیم و به مدارات پیچیده تر می رسیم.
- توصیه می شود دانشجویان مروری بر گیت های منطقی داشته باشند تا بتوانیم راحت تر درس را جلو ببریم.