

به نام خدا

آموزش VHDL

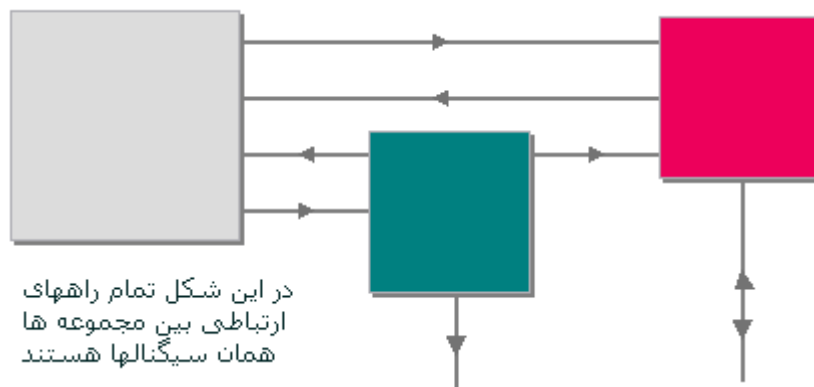
( قسمت دوم )

## Very High Speed Integrated Circuit Hardware Description Language ( Part 2 )

نحوه تعریف انواع سیگنالها و پورتها

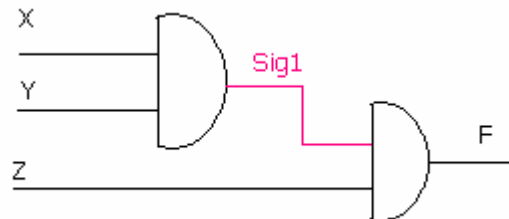
**Signal-۳**

شاید کسانی که درس مدار منطقی را گذراندند تعریف سیگنال را بدانند به هر حال در شکل زیر کاملا واضح است:



به عبارت ساده تر اگر ما دو گیت مثلا AND داشته باشیم که خروجی یکی ، ورودی دیگری باشد آنگاه تکلیف داده ای که از اولین AND خارج می شود چیست بله! این داده را باید به یک متغیر که به صورت سیگنال تعریف شده است نسبت دهیم. (در ضمن عملگر نسبت دادن این است: " a<=b " یعنی مقدار b را درون a قرار بده ، که در مثال زیر واضح تر است:

```
Entity test1 is
Port ( x , y , z :in bit ; f :out bit );
End entity;
Architecture t1 of test1 is
Signal sig1 : bit ;
Begin
Sig1<=x and y ;
F<=sig1 and z ;
End t1;
```



## ۴- انواع سیگنالها

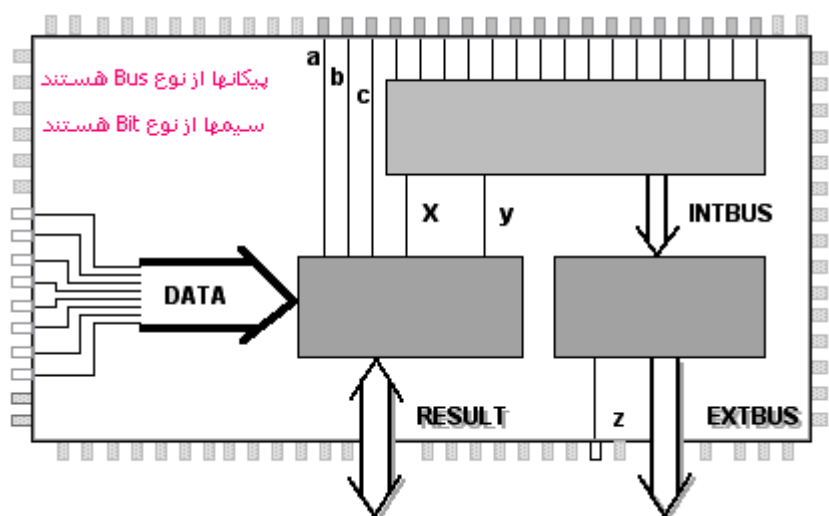
به ورودی و خروجیها هم می توان سیگنال گفت اما انواع مختلفی هستند

۱-سیم یا Wire

۲-گذرگاه یا Bus

که هر کدام از آنها در VHDL با نام مخصوصی عنوان می شود مثلا سیمها همیشه حامل یک بیت داده هستند و Bus ها حامل چند بیت.

که در VHDL هر متغیری را که می خواهیم از نوع سیم یا wire تعریف کنیم باید آن را از نوع Bit در نظر بگیریم همچنین برای متغیرهای از نوع گذرگاه یا Bus باید بنویسیم Bit\_Vector مانند شکل زیر:



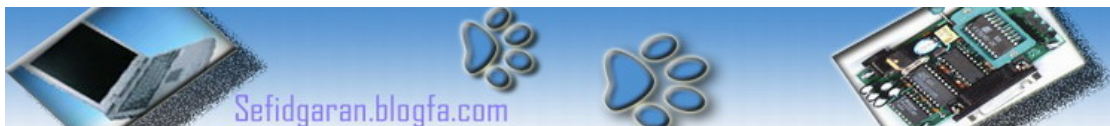
مثلا برای تعریف یک ورودی به نام Data از نوع Bus که حامل ۸ بیت داده باشد و یک ورودی به نام C که از نوع بیت است داریم:

```
Entity test2 is
Port(data : in bit_vector(7 downto 0) ; c : in bit);
End entity;
```

**نکته:** ورودی Data عددی است ۸ بیتی که اولین رقم از سمت راست معادل باینری آن دارای اندیس صفر می باشد اگر بخواهیم اندیسها از چپ به راست شروع شوند باید اینگونه بنویسیم:

```
Entity test2 is
Port(data : in bit_vector(0 to 7) ; c : in bit);
End entity;
```

**یادآوری مهم:** همیشه مکان تعریف ورودی و خروجی ها در Entity و قسمت Port میباشد و Signal ها همیشه در Architecture و قبل از Begin تعریف می شوند. مثلا در شکل قبل فقط آن پیکانی که بین دو مستطیل قرار گرفته و مقابله عبارت INTBUS نوشته شده از نوع Signal است این ادعا در مورد دو Wire به نامهای X و Y هم صادق است.



به مثال زیر که در مورد شکل قبل طرح شده توجه فرمایید:

**Entity Test3 is**

**Port**(a , b , c : in bit ; Data : in Bit\_Vector(7 downto 0) ;

**Z : in bit ; EXTBUS : out Bit\_Vector(4 downto 0) ;**

**RESULT : inout Bit\_Vector (0 to 7) );**

**End entity;**

**Architecture RTL of Test3 is**

**Signal : x , y : bit ;**

**Signal : INTBUS : bit\_vector(4 downto 0);**

**Begin**

**End RTL;**

**IN** : داده از نوع ورودی می باشد.

**OUT** : داده از نوع خروجی می باشد.

**INOUT** : داده می تواند هم خروجی و هم ورودی باشد.

درس اول را با ذکر دو نکته زیر به پایان می بریم:

۱-اسامی که شما برای Entity و Architecture هایتان انتخاب می کنید حتما باید با حروف الفبا آغاز شوند.

۲-اگر در میان خطوط برنامه خواستید توضیحات (Comment) اضافه کنید قبل از آن باید دو خط فاصله (همان علامت تفریق) قرار دهیم .

امید وارم این درس بدون هیچ ابهامی از نظر شما گذشته باشد.  
تا درس آینده ...

پایان قسمت دوم

نگارنده : فرشید سفیدگران  
کارشناسی کامپیوتر سخت افزار  
خرداد ۱۳۸۲

[Sefidgaran@gmail.com](mailto:Sefidgaran@gmail.com)  
<http://Sefidgaran.blogfa.com>