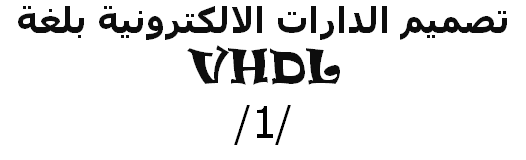
**الجلسة الثانية:**

****

**أولاً - ما هي لغة VHDL ؟**

VHDL : إن مصطلح **VHDL** هي اختصار للعبارة : **V**HSIC **HDL**

**VHSIC** **H**ardware **D**escription **L**anguage

**V**ery **H**igh **S**peed **I**ntegrated **C**ircuits

وهي تعني لغة توصيف الكيان الصلب للدارات المتكاملة ذات السرعات المرتفعة جدا .

تعتبر لغةVHDL لغة برمجة قياسية صممت من قبل وزارة الدفاع الأمريكية , حيث تستعمل VHDL في توصيف سلوك و محاكاة عمل الأنظمة الرقمية (ابتداء من البوابات البسيطة وانتهاء بأعقد الأنظمة الرقمية ), و تصميم ومحاكاة دارات VHSIC .

و تم اتخاذها لغة قياسية معتمدة من قبل IEEE في عام ١٩٨٧, و تمت مراجعتها في الولايات المتحدة الأميركية عام ١٩٩٣.

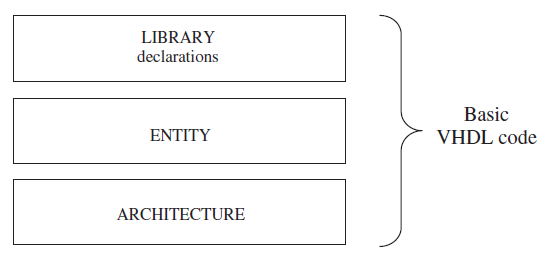
**ثانياً - مراحل توصيف نظام رقمي باستخدام لغة VHDL:**

1- التصريح عن المكتبات : استدعاء مكتبات تحتوي العناصر و التعليمات الأساسية و القياسية المستخدمة في التصميم مثل ieee , std , work .

2- وحدة التوصيف الخارجي (*Entity*): فيها يتم توصيف المداخل و المخارج , و من حيث الشكل يمكن تخيل هذه الوحدة بأنها الغلاف الخارجي للدارة الالكترونية المعبرة عن النظام الذي يحمل كافة الأقطاب .

3- وحدة التوصيف الداخلي (*Architecture*): تخصص هذه الوحدة للتعريف بسلوك أو بنية الدارة أو النظام المطلوب توصيفه انطلاقاً من إشارات الدخل وصولاً إلى إشارات الخرج وفق العلاقات الخاصة بالنظام .

**ملاحظة :** تتم ترجمة نموذج VHDL إلى مجموعة عناصر منطقية ضمن أداة منطقية قابلة للبرمجة مثل FPGA .

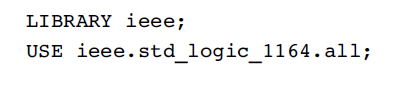


**أولا- استدعاء المكتبات :**

للتصريح عن مكتبة (أي لجعلها مرئية بالنسبة للتصميم ) نكتب ما يلي :

****

**مثال:**

****

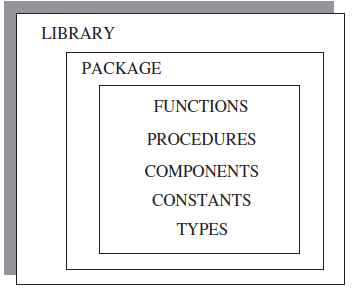
**ملاحظة:**

المكتبات التالية تكون مرئية أمام التصميم افتراضياً default , لذلك لسنا بحاجة للتصريح عنها :

**Std**: مكتبة المصادر و تنسيق النصوص و بعض أنواع المعطيات , وذلك من أجل بيئة تصميم vhdl

**Work** : هي المكتبة التي نخزن فيها تصميمنا (ملف البرنامج) بالاضافة إلى كل الملفات المنشأة بواسطة برنامج الترجمة أو المحاكاة .

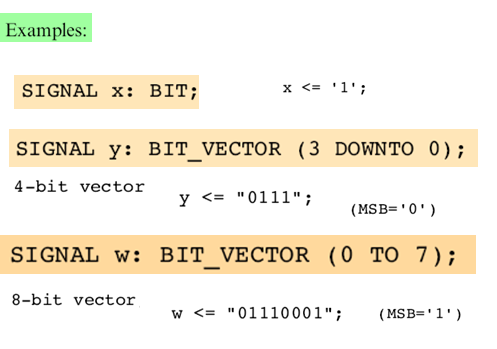
**الأجزاء الأساسية للمكتبة :**

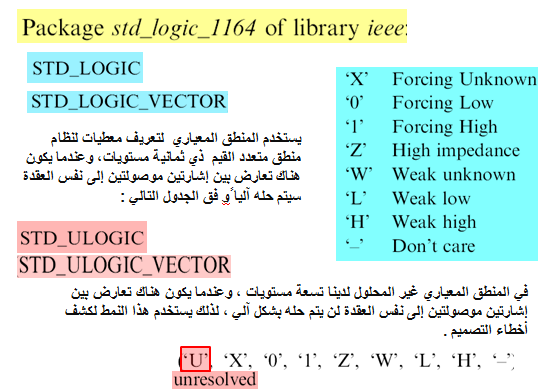


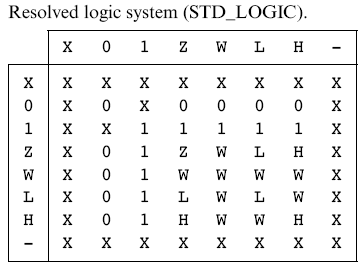
**تصنيف أنواع المعطيات المعرفة في VHDL وفق المكتبات المتوفرة** :

إن لغة VHDL تحتوي عدة أنماط للبيانات مسبقة التعريف و المحددة من خلال المعايير IEEE1076 و IEEE1164 .

هذه الأنواع تم تصنيفها في المكتبات ضمن كتل فرعية packages :







****

****

**-----------------------------------------------------------------------------**

****

الأعداد المؤشرة**** و الأعداد غير المؤشرة**** , يمتلكان مظهر نوع المعطيات std\_logic\_vector و لكنهما يقبلان العمليات الحسابية التي هي بالأساس مطبقة على نوع المعطيات integer , كما يحتوي هذا الجزء من المكتبة على توابع تحويل بعض أنواع المعطيات :

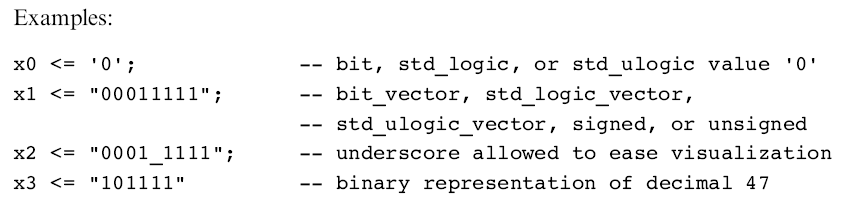
****

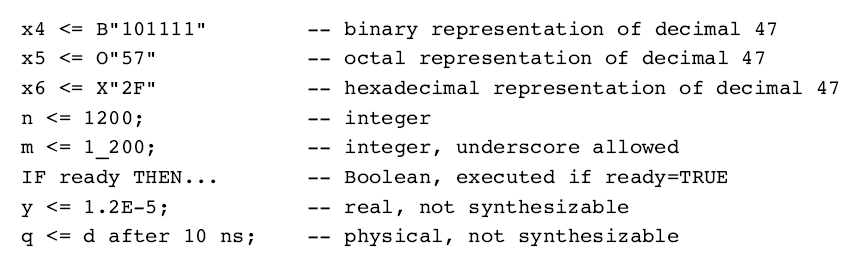
**-------------------------------------------------------------------**

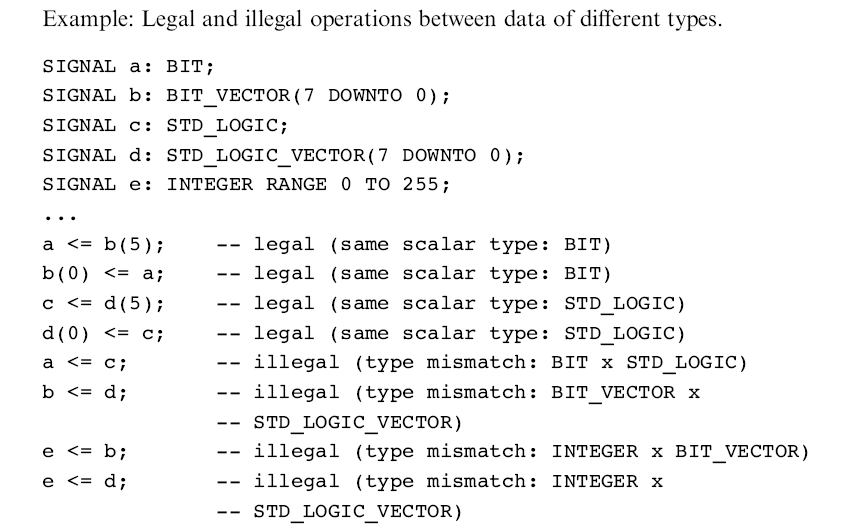
****

يحتويان على توابع تتيح إجراء العمليات على نوع المعطيات 

كما لو أنها كانت من النوع المؤشرة أو غير المؤشرة على الترتيب .

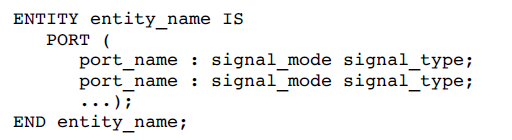
****

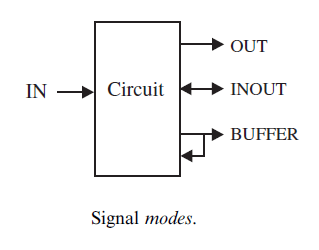
****

****

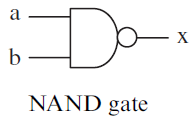
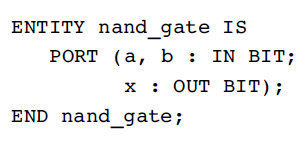
|  |  |
| --- | --- |
| Conversions supported by std\_logic\_1164 package | |
| Conversion | Function |
| std\_ulogic to bit | to\_bit(expression) |
| std\_logic\_vector to bit\_vector | to\_bitvector(expression) |
| std\_ulogic\_vector to bit\_vector | to\_bitvector(expression) |
| bit to std\_ulogic | To\_StdULogic(expression) |
| bit\_vector to std\_logic\_vector | To\_StdLogicVector(expression) |
| bit\_vector to std\_ulogic\_vector | To\_StdUlogicVector(expression) |
| std\_ulogic to std\_logic\_vector | To\_StdLogicVector(expression) |
| std\_logic to std\_ulogic\_vector | To\_StdUlogicVector(expression) |

**ثانياً- وحدة التوصيف الخارجي (*Entity*):**  فيها يتم توصيف المداخل و المخارج وفق النمط التالي :



و أنماط الإشارة موضحة بالشكل التالي : 

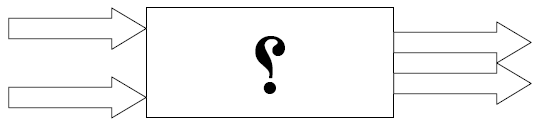
مثال :



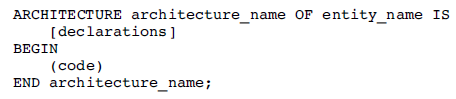
**ثالثاً -وحدة التوصيف الداخلي (*Architecture*):**

هي عبارة عن وصف يعبر عن كيفية عمل الدارة أي توضيح لسلوك الدارة , يمكن أن يتم توصيف النظام داخلياً بعدة طرق :

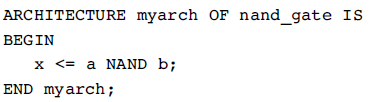
1. توصيف بنيوي **. 2-** توصيف سلوكي **.**

****

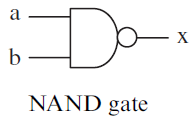
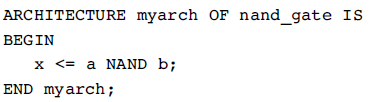
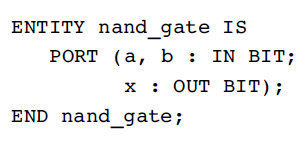
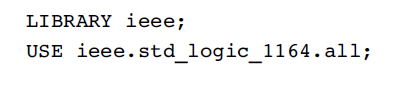
و يتم التصريح عنه بالشكل التالي :

****

مثال :

****

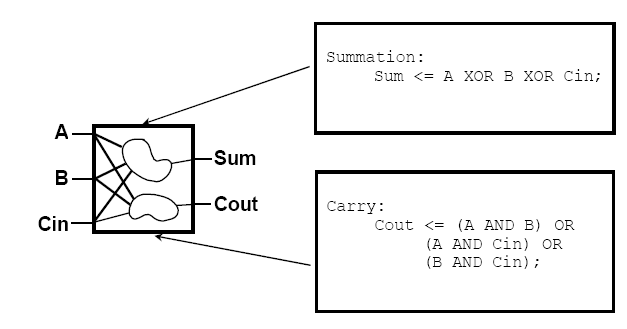
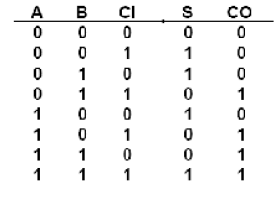
و هكذا يمكننا كتابة التوصيف بشكل كامل للمثال السابق :

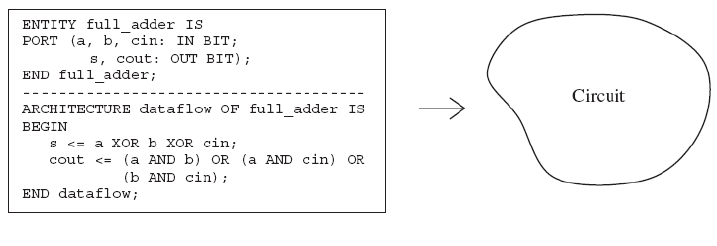
****

**ملاحظات :**

* في لغة VHDL إذا وضع (- - ) قبل الجملة فهي تعليقات أو ملاحظات.
* يجب أن تنتهي جميع العبارات و التصريحات بفاصلة منقوطة . (;)
* إذا كان هناك أكثر من متحول نفصل بينهم بفاصلة (,) .
* إسناد قيمة ما إلى إشارة رمزها (<=) , و إسناد قيمة ما إلى متحول رمزها (=:), و كذلك للتصريح عن قيمة ثابتة و لتعيين قيمة ابتدائية .
* يمكن استخدام الأرقام و الحروف و (\_) في أسماء المتحولات و الإشارات, لكن لا يمكن أن يبدأ الاسم برقم , وكذلك لا يمكن استخدام كلمة محجوزة في الاسم .
* لغة VHDL غير حساسة لحالة الحروف(صغيرة أو كبيرة).
* بالتناقض مع برامج الحاسوب العادية التي تكون متسلسلة عبارات VHDL تنفذ على التفرع لذلك نقول VHDL codeبدلاً من VHDL program.

**مثال : **

****

****

**الوظيفة :**

توصيف الدارات التالية بلغة VHDL :

* دارة Half ADDER .
* دارة . XOR , OR, AND

--------------------------------------------------------------------

في الجلسة القادمة سنتعرف على :

بعض التعليمات الأساسية المتوفرة في VHDL.

بالإضافة إلى حل بعض التمارين و مثال على التوصيف البنيوي .

****