

TD AEV

FICHE 1 Nexys3

Fiche 1 Nexys3

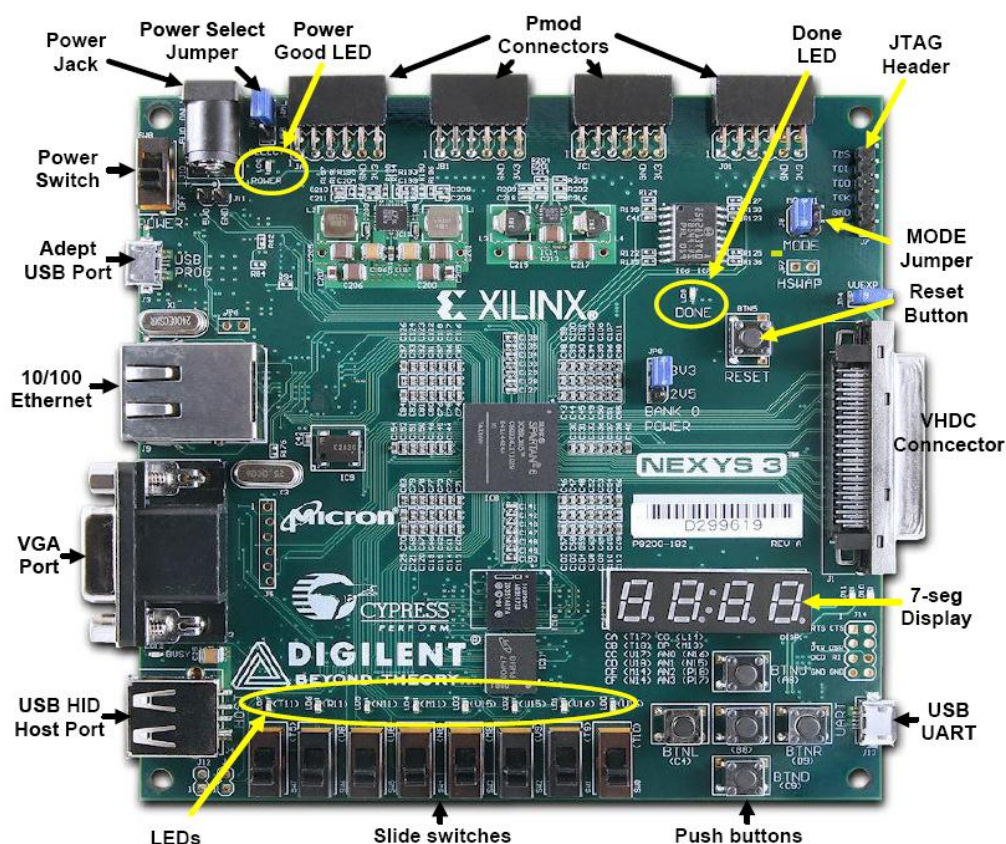
Présentation de la carte Nexys3

The Nexys3 is a complete, ready-to-use digital circuit development platform based on the Xilinx Spartan-6 LX16 FPGA. The Spartan-6 is optimized for high performance logic, and offers more than 50% higher capacity, higher performance, and more resources as compared to the Nexys2's Spartan-3 500E FPGA. Spartan-6 LX16 features include:

- 1 ☐ 2,278 slices each containing four 6-input LUTs and eight flip-flops
- 2 ☐ 576Kbits of fast block RAM
- 3 ☐ two clock tiles (four DCMs & two PLLs)
- 4 ☐ 32 DSP slices
- 5 ☐ 500MHz+ clock speeds

In addition to the Spartan-6 FPGA, the Nexys3 offers an improved collection of peripherals including 32Mbytes of Micron's latest Phase Change nonvolatile memory, a 10/100 Ethernet PHY, 16Mbytes of Cellular RAM, a USB-UART port, a USB host port for mice and keyboards, and an improved high-speed expansion connector. The large FPGA and broad set of peripherals make the Nexys3 board an ideal host for a wide range of digital systems, including embedded processor designs based on Xilinx's MicroBlaze.

Nexys3 is compatible with all Xilinx CAD tools, including ChipScope, EDK, and the free WebPack. The Nexys3 uses Digilent's newest Adept USB2 system that offers FPGA and ROM programming, automated board tests, virtual I/O, and simplified user-data transfer facilities.

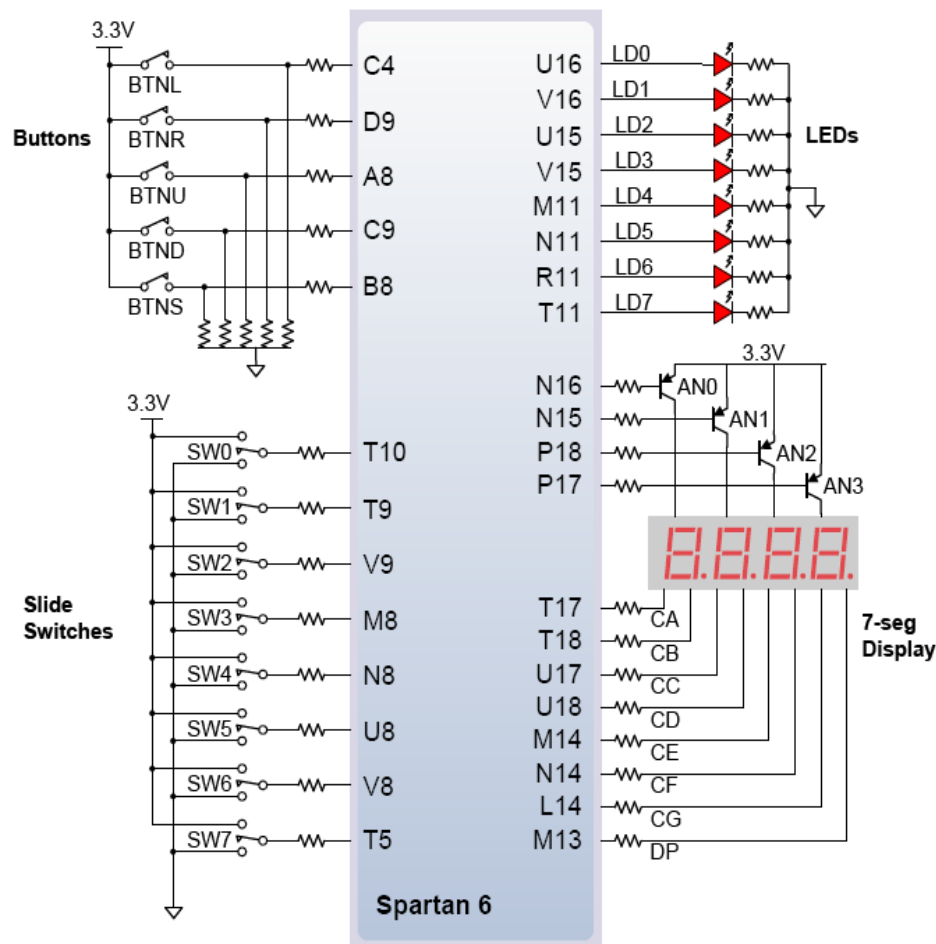


Les entrées sorties de la Naxys2

TD AEV

The Nexys3 board includes eight slide switches, eight push buttons, eight individual LEDs, and a four digit seven-segment display. The pushbuttons and slide switches are connected to the FPGA via series resistors to prevent damage from inadvertent short circuits (a short circuit could occur if an FPGA pin assigned to a pushbutton or slide switch was inadvertently defined as an output). The pushbuttons are "momentary" switches that normally generate a low output when they are at rest, and a high output only when they are pressed. Slide switches generate constant high or low inputs depending on their position.

The eight individual high-efficiency LEDs are anode-connected to the FPGA via 390-ohm resistors, so they will turn on when a logic high voltage is applied to their respective I/O pin. Additional LEDs that are not user-accessible indicate power-on, FPGA programming status, and USB and Ethernet port status.

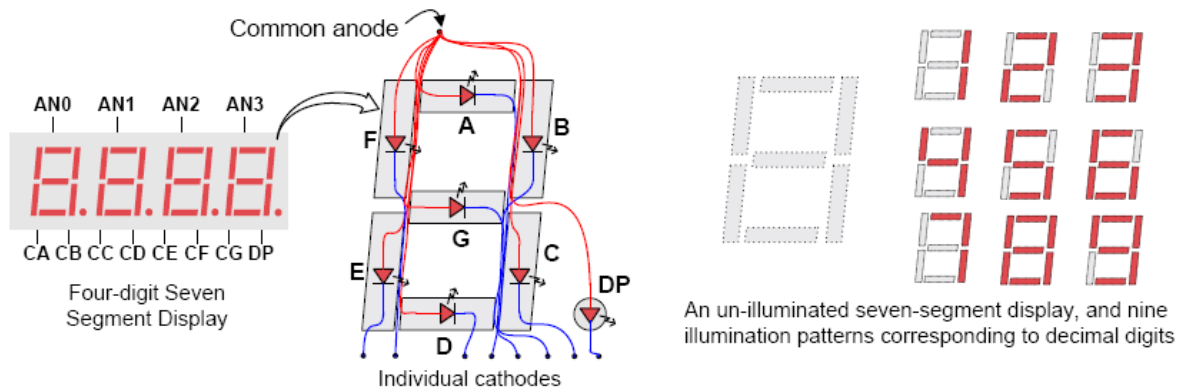


Affichage 7-segments

The Nexys3 board contains a four-digit common anode seven-segment LED display. Each of the four digits is composed of seven segments arranged in the figure pattern, with an LED embedded in each segment. Segment LEDs can be individually illuminated, so any one of 128 patterns can be displayed on a digit by illuminating certain LED segments and leaving the others dark. Of these 128 possible patterns, the ten corresponding to the decimal digits are the most useful.

The anodes of the seven LEDs forming each digit are tied together into one "common anode" circuit node, but the LED cathodes remain separate. The common anode signals are available as four "digit enable" input signals to the 4-digit display. The cathodes of similar segments on all four displays are connected into seven circuit nodes labeled CA through CG (so, for example, the four "D" cathodes from the four digits are grouped together into a single circuit node called "CD"). These seven cathode signals are available as inputs to the 4-digit display. This signal connection scheme creates a multiplexed display, where the cathode signals are common to all digits but they can only illuminate the segments of the digit whose corresponding anode signal is asserted.

TD AEV

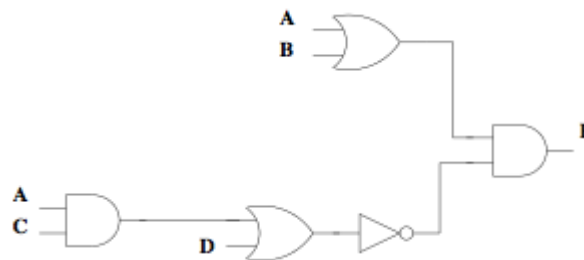


Fichier UCF

Un fichier UCF sert (entre autres) à définir le mapping entre les ports logiques et les ports physiques de la carte (ports du circuit conçu et ports de la nexys 3), pour permettre la gestion des entrées/ sorties. Il s'agit d'un fichier indispensable pour l'implémentation de tout système ayant besoin d'être alimenté par des données et/ ou fournissant des résultats visibles en sortie.

Exemple

Soit le circuit de la figure ci-dessous. Il s'agit d'un circuit prenant entrée 4 bits (A, B, C, D) pour calculer une sortie binaire E.



Pour fournir à un circuit implémenté sur une Nexys3 un nombre binaire, il suffit d'utiliser l'un des switches disponibles. Pour un bit de sortie, une LED (allumée pour 1 et éteinte pour 0) peut être utilisée. Ainsi, le fichier ci-dessous correspond au fichier UCF de l'exemple.

NET A LOC = T10; // switch tout à gauche sur la NEXYS 3

NET B LOC = T9; // switch suivant

NET C LOC = V9; // troisième switch

NET D LOC = M8; // quatrième switch

NET E LOC = U16; // LED au fond à gauche

Exercice 1

Soit le circuit qui réalise une addition 4 bits non signé. On dispose d'un additionneur 4 bits produisant un mot de 4 bits avec une retenue.

Question 1

Proposez une expérimentation de cet additionneur sur la carte nexys 3. Identifier tous les ports devant être connectés aux ES de la Nexys3 (pour chaque port donner la direction, taille et nom), donnez la description en vhdl.

Question 2

Donner le fichier UCF correspondant.

TD AEV

Exercice 2

Soit le fichier UCF suivant

```
Net LED<7> LOC=T11 ;  
Net LED<6> LOC=R11 ;  
Net LED<5> LOC=N11 ;  
Net LED<4> LOC=M11 ;  
Net LED<3> LOC=V15 ;  
Net LED<2> LOC=U15 ;  
Net LED<1> LOC=V16 ;  
Net LED<0> LOC=U16 ;
```

```
Net BTN<0> LOC=D9 ;  
Net BTN<1> LOC=C9 ;  
Net BTN<2> LOC=C4 ;  
Net BTN<3> LOC=A8 ;  
Net BTN<4> LOC=B8 ;
```

```
Net SW<0> LOC=T10 ;  
Net SW<1> LOC=T9 ;  
Net SW<2> LOC=V9 ;  
Net SW<3> LOC=M8 ;  
Net SW<4> LOC=N8 ;  
Net SW<5> LOC=U8 ;  
Net SW<6> LOC=V8 ;  
Net SW<7> LOC=T5 ;
```

```
Net SSEG_CA<0> LOC=T17 ;  
Net SSEG_CA<1> LOC=T18 ;  
Net SSEG_CA<2> LOC=U17 ;  
Net SSEG_CA<3> LOC=U18 ;  
Net SSEG_CA<4> LOC=M14 ;  
Net SSEG_CA<5> LOC=N14 ;  
Net SSEG_CA<6> LOC=L14 ;  
Net SSEG_CA<7> LOC=M13 ;
```

```
Net SSEG_AN<0> LOC=P17 ;  
Net SSEG_AN<1> LOC=P18 ;  
Net SSEG_AN<2> LOC=N15 ;  
Net SSEG_AN<3> LOC=N16 ;
```

Question 1

Donner le schéma représentant la connexion entre le circuit conçu et les ports de la nexys3 (mettre en évidence les nom des ports, leur direction et leur tailles)

Question 2

Est-il possible de

- connecter le même switch à deux entrées différentes dans le circuit (ex les 2 entrées d'une porte OU) ? Justifier
- connecter deux sorties du circuit à une même LED ? Justifier
- connecter deux LEDs à une même sortie du circuit ? Justifier
- connecter deux switches à une même entrée du circuit ? Justifier

Question 3

Ce circuit prend en entrée un nombre binaire sur 8 bits (8 switches). On voudrait concevoir le même circuit mais travaillant sur 16 bit, puis un autre sur 32 bits.

- Comment faire pour alimenter ces circuits en données ? (sachant que la carte ne dispose que de 8 switches)

Exercice 3- Manipulation des afficheurs 7 segments

TD AEV

Décodeur 7 segments simple

Pour permettre de prendre contact avec la plateforme de TP on va commencer par résoudre un problème simple : utiliser 4 interrupteurs en entrée et afficher en hexadécimal, la valeur correspondante sur un des quatre afficheurs sept segments.

Ce problème à résoudre correspond à une entité (entity) VHDL :

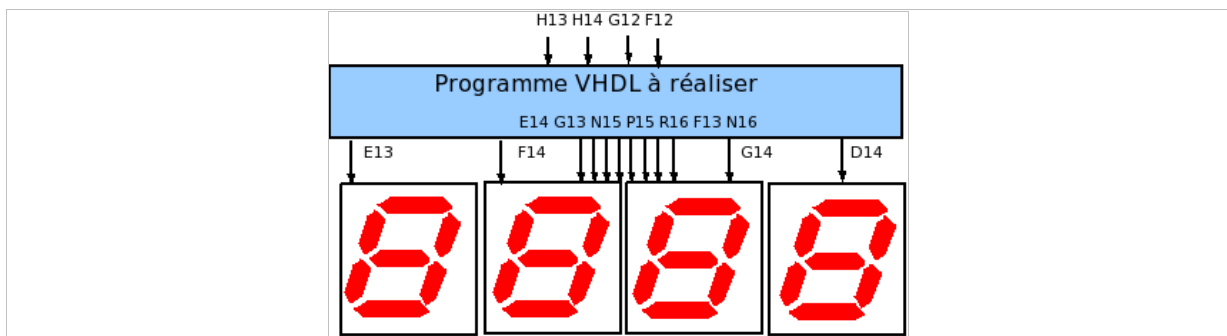
```
entity td1 is
  port (
    entrees : in std_logic_vector(3 downto 0); -- 4 inters en entree
    s7segs : out std_logic_vector(6 downto 0); -- 7 segments en sorties
    aff : out std_logic_vector(3 downto 0)); -- 4 selecteurs d'afficheur
end td1;
```

On peut remarquer que :

- ^ les quatre interrupteurs d'entrées sont regroupés dans une variable que l'on appelle entrees
- ^ les sept segments à afficher sont regroupés dans une variable nommée s7segs
- ^ la sélection des afficheurs est regroupée dans une variable nommée aff

La compréhension de ces choix nécessite une compréhension des afficheurs. Ils sont multiplexés à l'aide d'un signal par afficheur. L'affichage se fait ainsi avec 7 signaux (les segments à allumer) qui sont actifs à l'état bas.

La figure ci-dessous présente le travail à réaliser, il est ici connecté sur une carte nexys2.



Question 1

Expliquer le fonctionnement du système, puis donner le fichier UCF correspondant pour une carte nexys 3.

Décodeur 8 bits vers deux afficheurs

On utilisera que les deux afficheurs poids faibles.

Question 2

On voudrait réaliser un ensemble permettant d'afficher le mot de 8 bits sur deux afficheurs hexadécimal.

- Schématiser le circuit afficheur 8 bits . Peut-on afficher les 8 bits à la fois avec ce circuit ? Sinon combien phases sont nécessaires ?
- Donner un schéma plus détaillé de ce circuit en utilisant des boutons, des multiplexeurs etc...
- Dans ce cas, qu'est ce qui changerait dans le fichier UCF par rapport à la question 1 ?