

1^{ère} partie :
modélisation

2^{ème} partie :
microcontrôleurs

3^{ème} partie :
API

Function Block Diagram – FBD

1 – NOTION DE BLOC

2 – BLOCS STANDARD

3 – BLOCS PERSONNALISÉS

1 – Notion de bloc

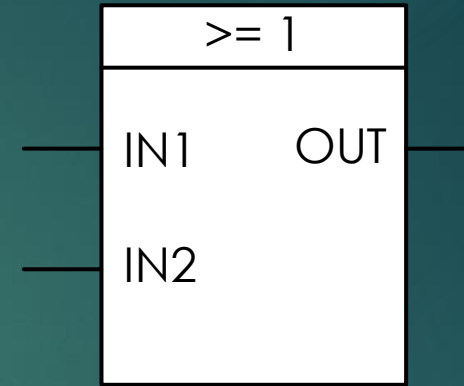
2 – BLOCS STANDARD

3 – BLOCS PERSONNALISÉS

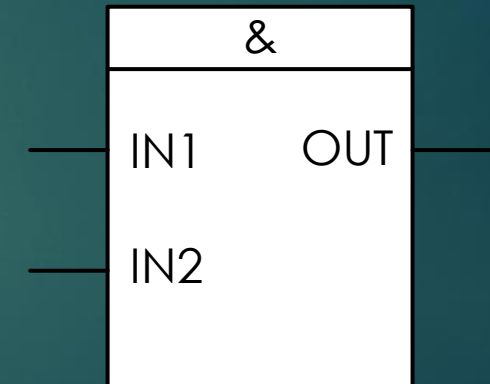
Notion de bloc

3

- ▶ Le langage Function Block Diagram (FBD) est un langage de programmation graphique pour API basé sur des blocs
- ▶ Vous en connaissez déjà les grandes lignes
 - ▶ Fonctions représentées sous forme de rectangles
 - ▶ Les entrées arrivent à gauche
 - ▶ Les sorties sont à droite
 - ▶ Le nom est en haut



Exemple de bloc :
Fonction OU



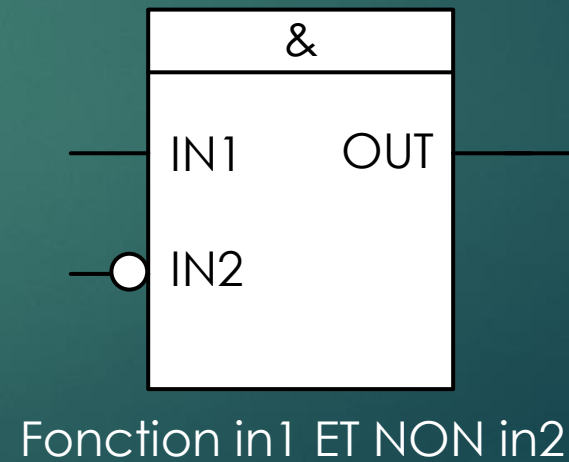
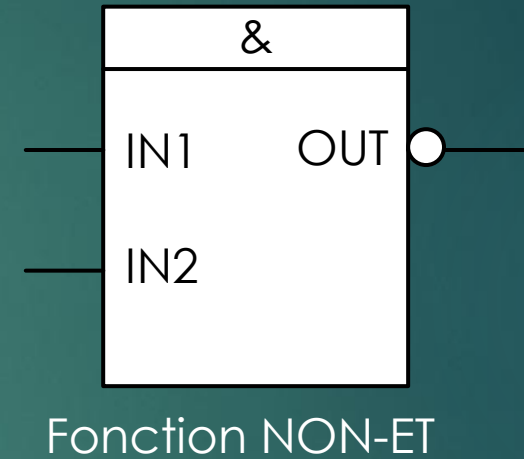
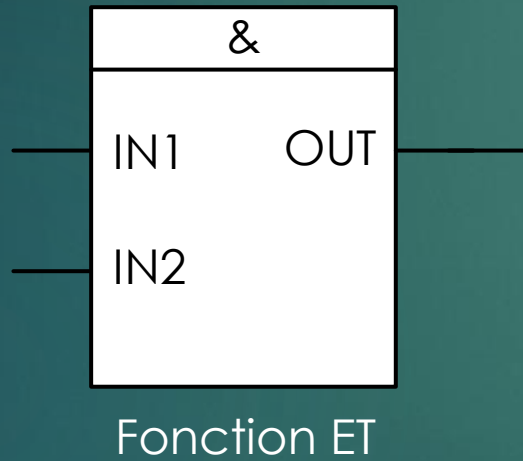
Exemple de bloc :
Fonction ET

Notion de bloc

- ▶ Le langage Function Block Diagram (FBD) est un langage de programmation graphique pour API basé sur des blocs
- ▶ Vous en connaissez déjà les grandes lignes
 - ▶ Fonctions représentées sous forme de rectangles
 - ▶ Les entrées arrivent à gauche
 - ▶ Les sorties sont à droite
 - ▶ Le nom est en haut
- ▶ On peut chaîner plusieurs blocs en branchant une sortie de l'un sur une entrée d'un autre
- ▶ Les liaisons sont orientées de gauche à droite
- ▶ Les entrées/sorties peuvent être munies de cercles indiquant une inversion de la valeur

Exemple : inversion

5



Types de données

6

- ▶ Les données des entrées et des sorties ne sont pas forcément uniquement des bits
 - ▶ Octets
 - ▶ Mots
 - ▶ Réels
 - ▶ Etc.
- ▶ Si l'on relie la sortie d'un bloc à l'entrée d'un autre, les données doivent obligatoirement avoir le même type

Construction d'un bloc

7

- ▶ Un bloc représente une fonction dont on voit l'interface
- ▶ Le contenu de la fonction peut être codé de différentes manières selon l'automate
 - ▶ Assemblage de blocs FBD
 - ▶ Ladder
 - ▶ Code ST
 - ▶ Etc.

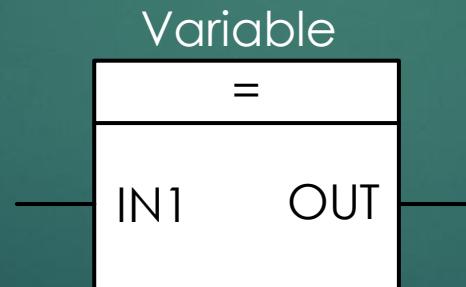
1 – NOTION DE BLOC

2 – Blocs standard

3 – BLOCS PERSONNALISÉS

Blocs standard

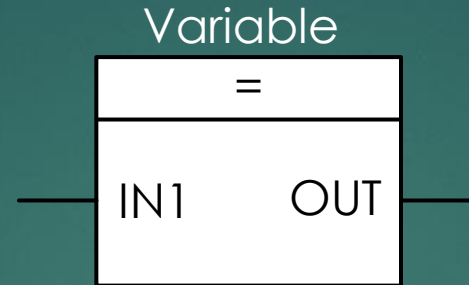
- ▶ De nombreux blocs standard sont définis dans la norme et peuvent être utilisés sans avoir à les redéfinir
- ▶ Un bloc fondamental est le bloc d'affectation, qui permet de donner une valeur à une variable
 - ▶ L'entrée est la valeur à affecter, la sortie recopie la valeur de la variable



Variantes du bloc d'affectation

10

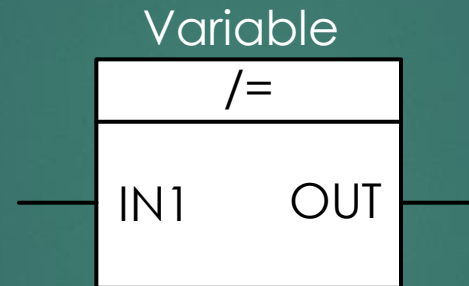
Bloc d'affectation



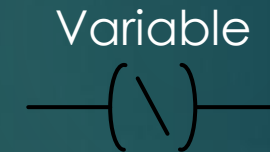
équivalent à



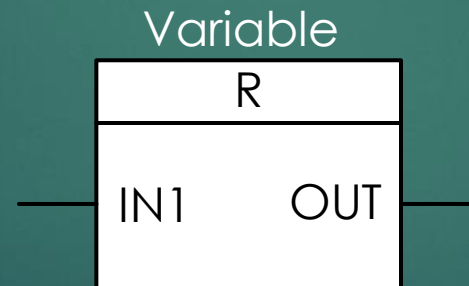
Bloc de négation de l'affectation



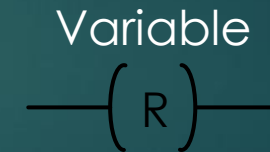
équivalent à



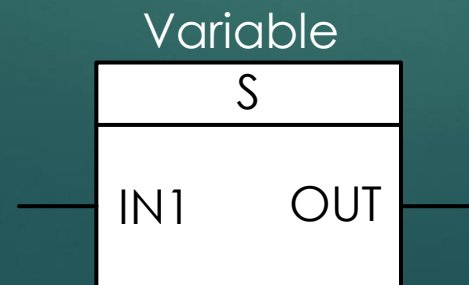
Bloc reset



équivalent à



Bloc set



équivalent à



Quelques blocs standard

11

- ▶ Fonctions logiques
 - ▶ ET
 - ▶ OU
 - ▶ NON
 - ▶ ...
- ▶ Bascules
 - ▶ Bascule RS (reset prioritaire)
 - ▶ Bascule SR (set prioritaire)
 - ▶ ...
- ▶ Détection de front
 - ▶ Front montant
 - ▶ Front descendant
- ▶ Timers
 - ▶ Timer à impulsion
 - ▶ Timers à délai
- ▶ Compteurs
- ▶ ...

1 – NOTION DE BLOC

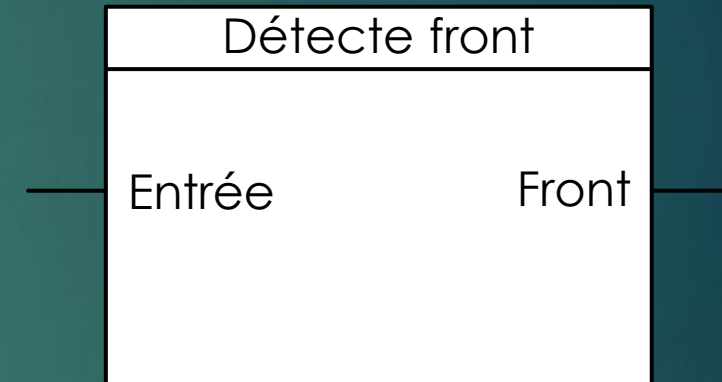
2 – BLOCS STANDARD

3 – Blocs personnalisés

Blocs personnalisés

13

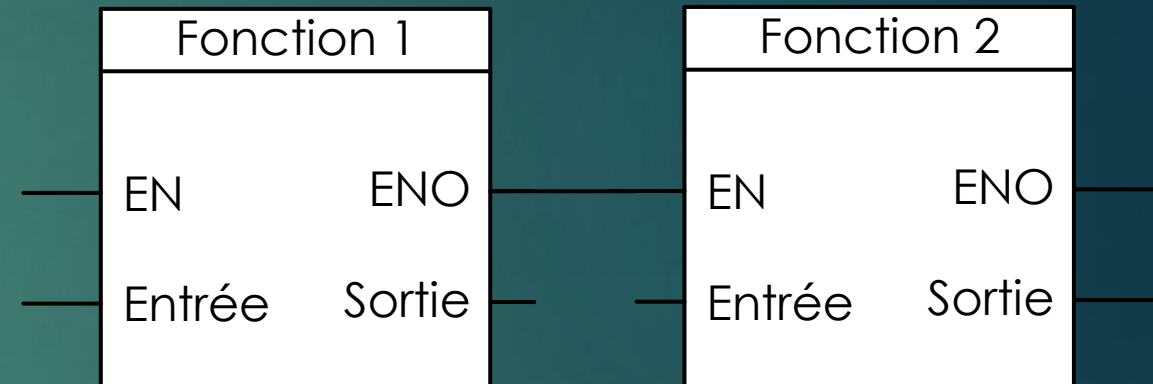
- ▶ Il est ensuite possible de définir ses propres blocs
 - ▶ Définition des entrées
 - ▶ Définition des sorties
 - ▶ Variables internes possibles ou non selon si on utilise une fonction ou un bloc fonctionnel



Blocs personnalisés

14

- ▶ Il est ensuite possible de définir ses propres blocs
 - ▶ Définition des entrées
 - ▶ Définition des sorties
 - ▶ Variables internes possibles ou non selon si on utilise une fonction ou un bloc fonctionnel
- ▶ Des entrées/sorties importantes sont EN et ENO
 - ▶ Permettent de chaîner les blocs



Exercise

Exercice

16

- Convertir la fonction **Feux de carrefour** vue en TD et TP en FBD à l'aide des blocs suivants :

