

Université Mohamed Boudiaf - M'sila

Enseignant : Dr El Oualid ZOUGGAR

Faculté de Technologie

Durée : 1.5 H

Département de génie électrique

2^{ème} Année Master : Electromécanique

Année universitaire 2023-2024

Travaux pratiques d'automates programmables industrielles

TP01:
A la découverte de STEP 7 MicroWin et l'automate programmable
S7-200



Objectif :

Nous voulons vous montrer à l'aide d'exercices pratiques comme il est simple de programmer en CONT avec STAP 7 MicroWin.

Moyens mis en œuvre :

- Le logiciel STEP 7 Micro Win
- Une lampe
- Des relais d'interfaces (24VDC)
- Un moteur asynchrone
- Des contacteurs

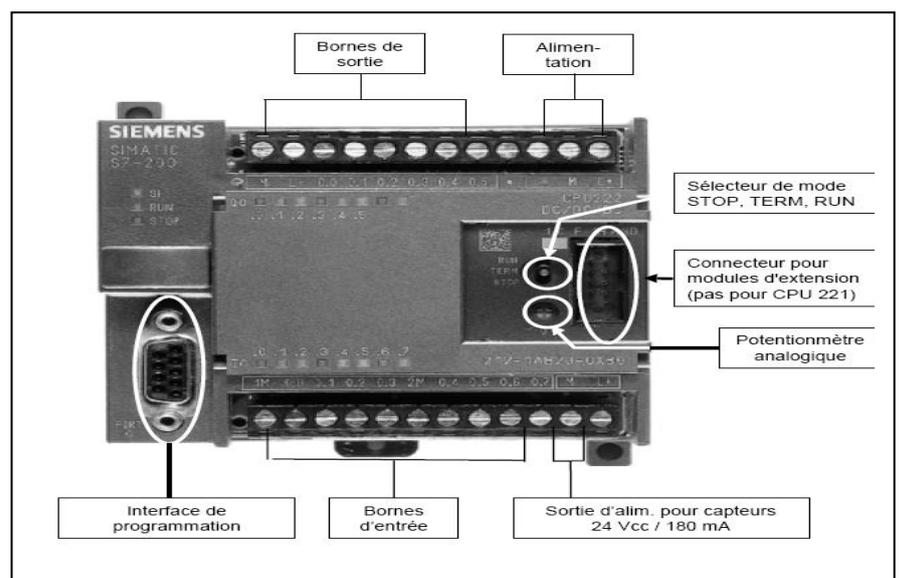
Création d'un programme à l'aide de fonctions binaires

Dans ce TP, vous créez des programmes à l'aide des fonctions binaires permettant l'adressage des entrées et sorties de votre CPU (unité centrale d'un automate programmable).

Structure générale des API :

Les caractéristiques principales d'un automate programmable industriel (API) sont:

- Compact ou modulaire
- Tension d'alimentation
- Taille mémoire
- Sauvegarde (EPROM, EEPROM, pile, ...)
- Nombre d'entrées / sorties
- Modules complémentaires (analogique, communication,..)
- Langage de programmation.

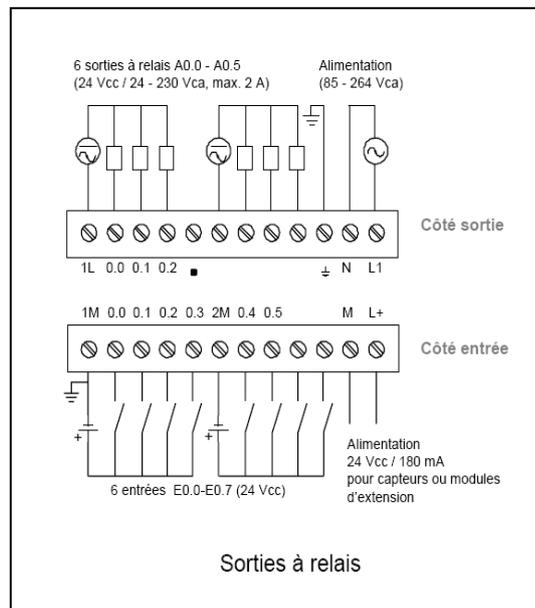


Combinaisons logiques :

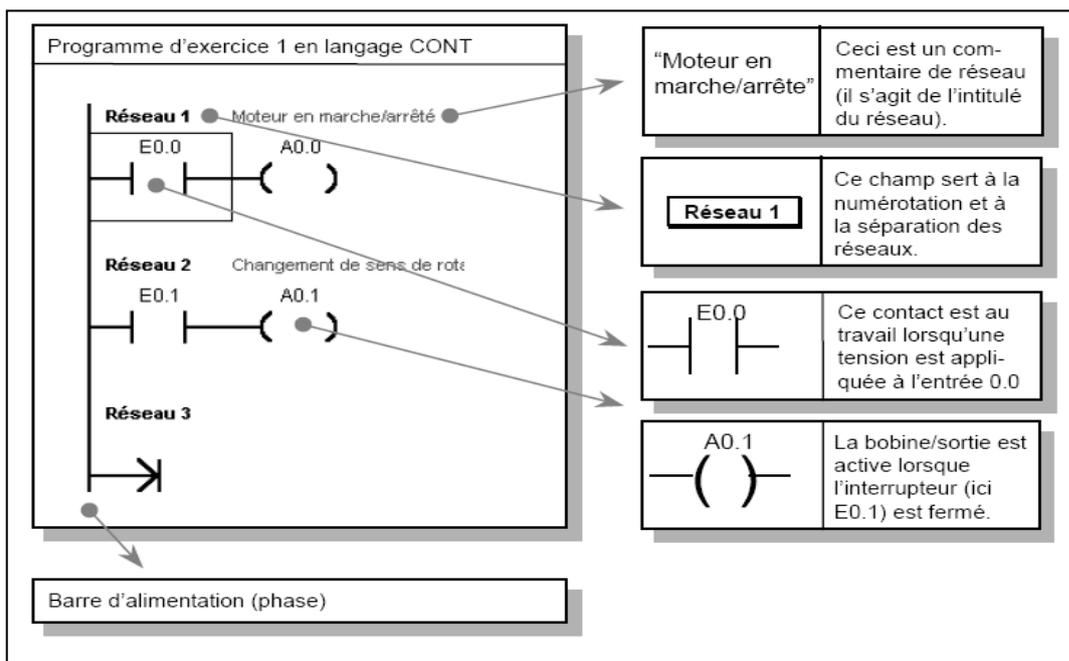
Les opérations combinatoires sur bits utilisent deux chiffres : 1 et 0. Ces deux chiffres sont à la base du système de numération binaire et sont appelés chiffres binaires ou bits. Pour les contacts et les bobines, 1 signifie activé ou excité et 0 signifie désactivé ou déexcité. Les opérations de combinaison sur bits évaluent les états de signal 1 et 0 et les combinent selon la logique booléenne. Le résultat de ces combinaisons est égal à 1 ou 0. Il s'agit du résultat logique (RLG).

Il existe des opérations combinatoires sur bits pour effectuer les fonctions suivantes :

Contact/contacteur	Instruction dans l'API avec la fonction correspondante
	Interrogation : Est-ce qu'il circule du courant ? Si oui, le résultat de l'interrogation est vrai. (interrogation à "1")
	Interrogation : est-ce qu'il ne circule pas de courant ? Si oui (pas de courant), le résultat de l'interrogation est vrai. (interrogation à "0")
	Bobine : si une bobine est alimentée avec la valeur "vraie" (courant), elle est activée (attraction)
	Couplage en série : (combinaison ET). Le premier ET le deuxième contact doivent être fermés pour laisser passer le courant
	Couplage en parallèle (combinaison OU). Le premier OU le deuxième contact doit être fermé pour laisser passer le courant



Les réseaux servent à structurer un programme. On entre chaque circuit dans un réseau.



Déroulement du TP

Vous allez travailler dans ce TP en mode ladder uniquement (langage à contacts). Vous ne devez donc pas concevoir de grafset pour ce TP. Le ladder ressemble par son écriture à la schématisation des contacts électriques et fonctionne de la même façon.

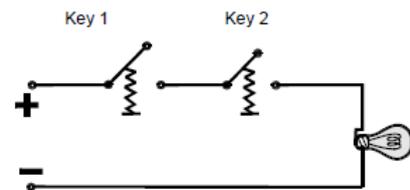
Vous allez réaliser plusieurs programmes en langage ladder.

Travail demandé :

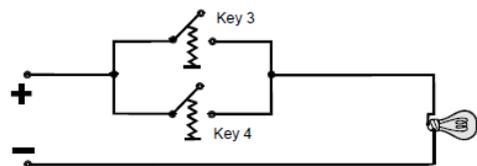
La fonction binaire que vous aurez en première à programmer est la fonction ET. Cette dernière peut être représentée par un circuit électrique à deux commutateurs en série.

L'actionnement du commutateur 1 et 2 faits

Allumer la lampe.



La seconde fonction binaire que nous serons amenés à programmer est la fonction OU. On peut également la représenter par un circuit électrique.

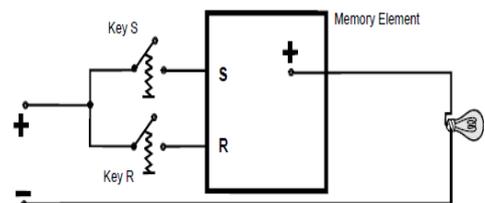


La troisième fonction qui nous occupera est la bascule (fonction SR).

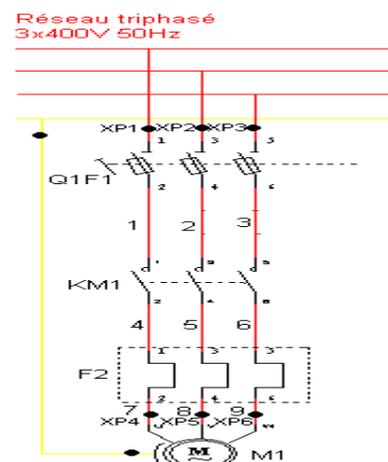
L'actionnement du commutateur S fait s'allumer la

Lampe qui reste allumée jusqu'à l'action du

Commutateur R.



La quatrième expérience c'est le démarrage direct d'un moteur asynchrone



Symbole Ladder :

Les symboles utilisés dans ce mode de programmation sont :

 : variable d'entrée ou contact à fermeture

 : variable d'entrée complétementée ou contact à ouverture,

-()- : variable de sortie (B12 bit interne; O0,2 sortie automate, etc..),

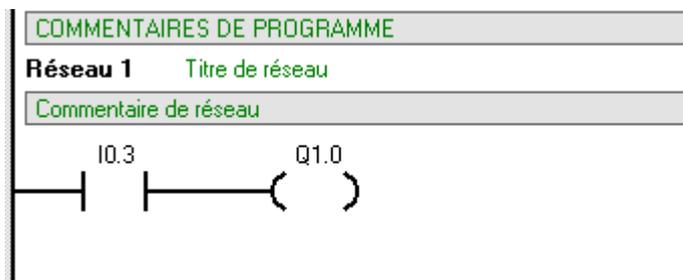
-(S)- : sortie mise à un mémorisée (S = set),

-(R)- : sortie mise à zéro mémorisée (R = reset),

Conseil :

Faites un réseau simple pour commencer avec une entrée puis une sortie, puis vous complétez votre programme au fur et à mesure

Exemple :



Le fonctionnement de ce réseau Ladder est le suivant :

Si j'ai l'entrée I0.3 alors j'active la sortie Q1.0

Démarche à suivre :

- Définir les entrées et sorties de l'API.
- Câbler votre schéma.
- Construire vos réseaux.
- Transférez votre programme dans API.
- Testez votre programme.