**INTRODUCTION GENERALE**

La robotique est un domaine essentiellement pluridisciplinaire. Ses réalisations conjuguent naturellement l’automatique, l’informatique, la mécanique et l’électronique. Ses liens plus récents avec les sciences de la vie, les sciences de la matière, ou les sciences cognitives ont donné lieu à des expérimentations très originales et ouverts de nouvelles perspectives. Du cout cette science permet la conception et la réalisation des machines automatiques (les robots) qui sont aujourd’hui la technologie du jour et importante, qui sont initialement été créé pour décharger l’homme des tâches fastidieuses. Par exemple, les tâches de manutention sont souvent fatigantes.

Dans ce rapport on a travaillé sur le robot pro-bot 128 pour lequel il faut réaliser des programmes qui lui fait déplacer de trois façons différentes, donc on a pensé à suivre un plan détaillé comme suit :

Le travail est partagé en deux chapitres. Dans le premier, nous focaliserons sur les définitions et les généralités d’où on a parlé de la robotique en général avec son historique ainsi ses différentes domaine d’utilisation qui est vraiment vaste. Effectivement, la définition des robots pour passé par la suite à une présentation détaillé de notre robot utilisé dans ce travaille. Différents aspects seront abordés tel que les cartes utilisées, ses composantes. Nous expliquerons ensuite les deux étapes de conception et de réalisation dans le chapitre qui suit ou on va parler des mouvement demandé et les différents outils utilisés le langage C-CONTROL, ainsi les étapes de son installation. On fini le travaille avec une conclusion général.

* **Introduction**

En effet, la robotique est très rependue dans l'industrie, où elle effectue sans relâche des tâches répétitives et avec rigueur. Dans les chaînes de montage de l'industrie automobile, ils y remplacent les ouvriers dans les tâches pénibles et dangereuses comme la peinture ou la soudure. Les robots industriels sont maintenant munis de systèmes de vision qui leur procurent une souplesse d'exécution et des moyens de vérifier la qualité des produits fabriqués.

**I.1. Historique**

Au cours de l'histoire on peut distinguer 3 types de robots correspondant en quelques sorte à l'évolution de cette "espèce" créée par l'Homme.

Le **premier type** de machine que l'on peut appeler robot correspond aux "**Automates**". Ceux-ci sont généralement programmés à l'avance et permettent d'effectuer des actions répétitives dans un environnement donné.

Le **second type** de robot correspond à ceux qui sont équipés de capteurs (en fait les sens du robot). On trouve des capteurs de température, photo électronique, à ultrasons pour par exemple éviter les obstacles et/ou suivre une trajectoire. Ces capteurs vont permettent au robot une relative adaptation à son environnement afin de prendre en compte des paramètres aléatoires qui n'aurait pu être envisagés lors de leur programmation initiale. Ces robots sont donc bien plus autonomes que les automates mais nécessitent un investissement en temps de conception et en argent plus conséquent.

Enfin le **dernier type** de robot existant correspond à ceux disposant d'une intelligence dite "artificielle" et reposant sur des modèles mathématiques complexes tels que les réseaux de neurones. En plus de capteurs physiques comme leurs prédécesseurs, ces robots peuvent prendre des décisions beaucoup plus complexes et s'appuient également sur un apprentissage de leurs erreurs comme peut le faire l'être humain. Bien sûr il faudra attendre encore longtemps avant que le plus "intelligent" des robots ne soit égal, tant par sa faculté d'adaptation que par sa prise de décisions, à l'Homme.[2]

**I.2. Robotique**

la robotique est la science qu' intéresse aux robots . en fait, il s'agit d'un domaine multidisciplinaire : on y trouve des aspects concernant la mécanique, l'informatique, l'électronique...[1]

**I.3. Les domaines d’utilisation**

**-** Domaine de l’exploration.

- Domaine du Médical.

- Domaine du Service.

- Domaine industriel.

**I.4. Définition d'un Robot**

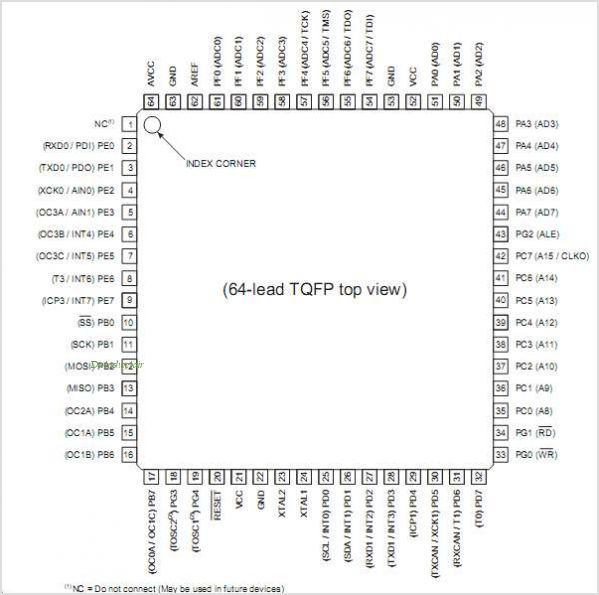
Un robot est une machine équipée de capacités de perception, d’action et de décision qui lui permet d’agir de manière autonome dans son environnement en fonction de la perception qu’il en a et de ses objectifs.[4]

**I.4.1. Présentation de Robot PRO-128**

Le robot Pro-Bot 128 est conçu pour fonctionner avec un microcontrôleur « CCONTROL Pro 128» basé sur un « AT90CAN » de la société ATMEL

* **Présentation de la carte C-CONTROL Pro 128**
* **Pinout du microcontrôleur AT90CAN**

Les I/O sont :



**Port A (PA7..PA0)**

**Port B (PB7..PB0)**

**Port C (PC7..PC0)**

**Port D (PD7..PD0)**

**Port E (PE7..PE0)**

8-bit bi-directional I/O port

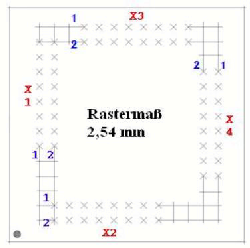
**PE0**  et **PE1** servent à

communiquer avec l’ordinateur

* **Pinout du boitier C-CONTROL vs microcontrôleur AT90CAN**

|  |  |  |
| --- | --- | --- |
| **Module** | **M128** | **Port** |
| **X1\_16** | **2** | **PE0** |
| **X1\_15** | **3** | **PE1** |
| **X1\_14** | **4** | **PE2** |
| **X1\_13** | **5** | **PE3** |
| **X1\_12** | **6** | **PE4** |
| **X1\_11** | **7** | **PE5** |
| **X1\_10** | **8** | **PE6** |
| **X1\_09** | **9** | **PE7** |
| **X1\_08** | **10** | **PB0** |
| **X1\_07** | **11** | **PB1** |
| **X1\_06** | **12** | **PB2** |
| **X1\_05** | **13** | **PB3** |
| **X1\_04** | **14** | **PB4** |
| **X1\_03** | **15** | **PB5** |
| **X1\_02** | **16** | **PB6** |
| **X1\_01** | **17** | **PB7** |

|  |  |  |
| --- | --- | --- |
| **Module** | **M128** | **Port** |
| **X2\_5** | **18** | **PG3** |
| **X2\_6** | **19** | **PG4** |
| **X2\_3** | **20** |  |
| **X2\_9** | **25** | **PD0** |
| **X2\_10** | **26** | **PD1** |
| **X2\_11** | **27** | **PD2** |
| **X2\_12** | **28** | **PD3** |
| **X2\_13** | **29** | **PD4** |
| **X2\_14** | **30** | **PD5** |
| **X2\_15** | **31** | **PD6** |
| **X2\_16** | **32** | **PD7** |
| **X2\_7** | **33** | **PG0** |
| **X2\_8** | **34** | **PG1** |
| **X2\_4** | **43** | **PG2** |



|  |  |  |
| --- | --- | --- |
| **Module** | **M128** | **Port** |
|  | **1** |  |
|  | **23** |  |
|  | **24** |  |

|  |  |  |
| --- | --- | --- |
| **Module** | **M128** | **Port** |
| **X3\_16** | **44** | **PA7** |
| **X3\_15** | **45** | **PA6** |
| **X3\_14** | **46** | **PA5** |
| **X3\_13** | **47** | **PA4** |
| **X3\_12** | **48** | **PA3** |
| **X3\_11** | **49** | **PA2** |
| **X3\_10** | **50** | **PA1** |
| **X3\_09** | **51** | **PA0** |
| **X3\_08** | **54** | **PF7** |
| **X3\_07** | **55** | **PF6** |
| **X3\_06** | **56** | **PF5** |
| **X3\_05** | **57** | **PF4** |
| **X3\_04** | **58** | **PF3** |
| **X3\_03** | **59** | **PF2** |
| **X3\_02** | **60** | **PF1** |
| **X3\_01** | **61** | **PF0** |

|  |  |  |
| --- | --- | --- |
| **Module** | **M128** | **Port** |
| **X4\_10** | **21** |  |
| **X4\_12** | **22** |  |
| **X4\_8** | **35** | **PC0** |
| **X4\_7** | **36** | **PC1** |
| **X4\_6** | **37** | **PC2** |
| **X4\_5** | **38** | **PC3** |
| **X4\_4** | **39** | **PC4** |
| **X4\_3** | **40** | **PC5** |
| **X4\_2** | **41** | **PC6** |
| **X4\_1** | **42** | **PC7** |
| **X4\_10** | **52** |  |
| **X4\_12** | **53** |  |
| **X4\_11** | **62** |  |
| **X4\_12** | **63** |  |
| X4\_9 | **64** |  |

* **Présentation de la carte 1**

Circuit intégré

Alimentation Extérieure

Condensateurs

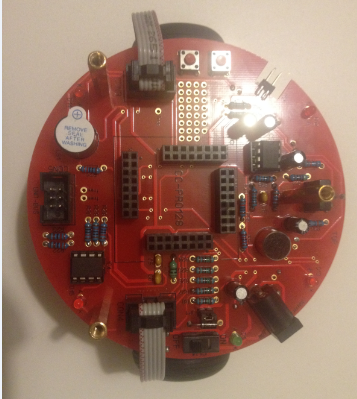
Port USB

Transducteur acoustique

Boutons

Interrupteur

Réglette à douilles pour C-Control PRO



**Figure 1: Présentation de la carte 1**

* **Présentation de la carte 2**

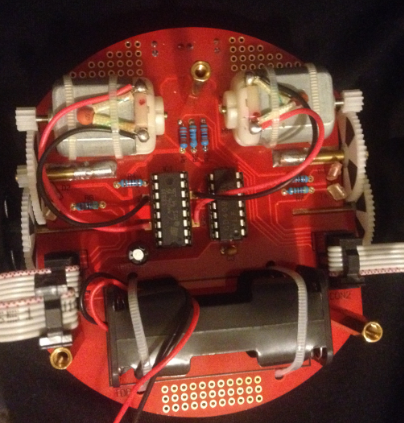
Capteurs de roue

Pignon moteur

Entretoise pour platine, support de piles et prise de charge

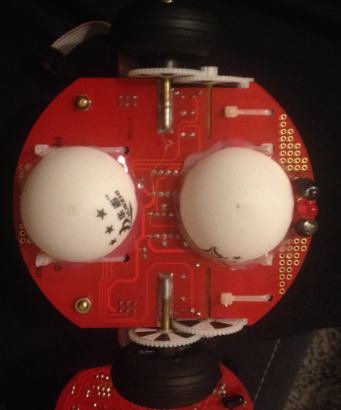
les câbles

de connexion



**Figure 2: Présentation de la carte 2**

* **Présentation du deuxième coté de la carte 2**



Phototransistors et LED IR

Balle de ping-pong

**Figure 3: Présentation du deuxième coté de la carte 2.**

**I.4.2.Les éléments du robot [3]**

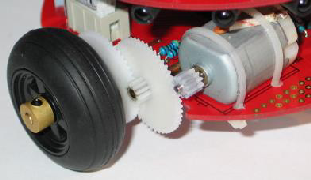
* **Alimentation:**
* Le robot est conçu pour fonctionner avec des accumulateurs (accu) car ils permettent de délivrer plus de courant. Dans ce cas le « jumper » (JP1) doit être en position fermé ce qui court-circuite la diode D4.
* Ces accumulateurs peuvent être rechargés sans être enlevés en basculant le commutateur (SW1) sur OFF (à vérifier)
* Si on utilise des piles alors le « jumper » (JP1) doit être enlevé (ouvert) pour qu’aucun courant ne puisse entrer dans les piles.



* **Connexion vers l’ordinateur**
* Ces 2 connecteurs ne seront pas utilisés avec la X1 de la carte car ils correspondent à la prise USB.
* **Résistances dépendantes de la lumière (LDR)**
* Les résistances dont la valeur change en fonction de l’intensité lumineuse sont ce qu’on appelle des LDR (Light X3 Dependent Resistor)
* A connecter sur des entrées analogiques



* **Les moteurs**
* Le control des moteurs s’effectue avec 2 CI : le CD4093 (Quad 2-Input NAND Schmitt Trigger) et le L293D (PUSH-PULL FOUR CHANNEL DRIVER WITH DIODES)



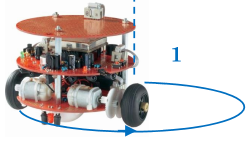
**conclusion**

A la fin de ce chapitre on a arrivé à définir les déférentes notion de la robotique, puis on a détaillé notre robot "C-Control Pro128 ".

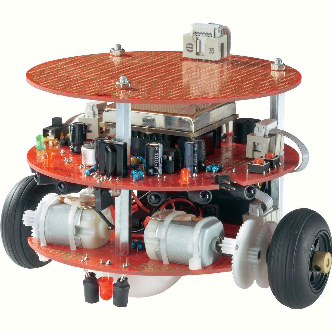
**II.1. Conception**

Le robot va faire trois mouvements différents :

* **mouvement cercle**

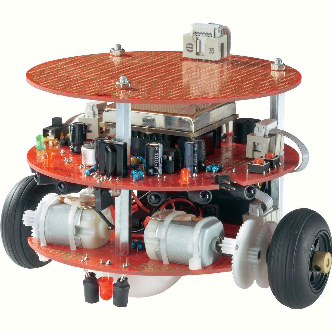


* **le code source 1**
* **mouvement ligne droit**



**2**

* **code source 2**
* **mouvement carré**



**3**

* **code source 3**

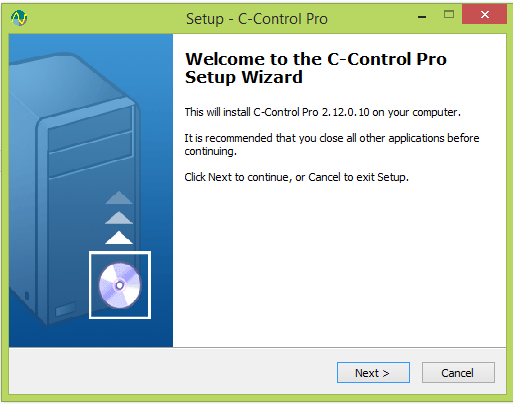
**II.2. Réalisation**

* **Le logiciels utilisé**

**C-Control Pro** la série C-Control a été élargie pour inclure le C-Control PRO famille de 2005. Ces unités ont par rapport à la famille C-Control-I significativement plus de ressources de mémoire et de matériel.

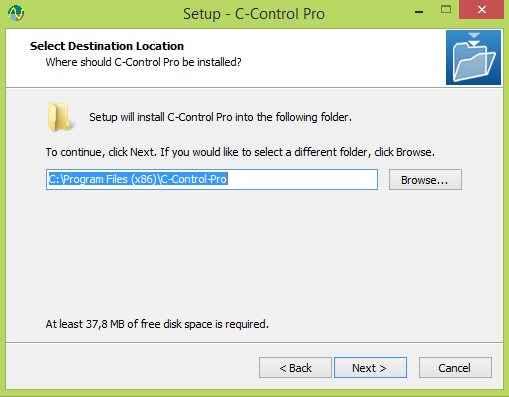
 Le C-Control PRO est fondée sur les mêmes microcontrôleurs RISC de [la famille AVR](https://de.wikipedia.org/wiki/Atmel_AVR) d'Atmel. Le MEGA128 offre beaucoup plus d'E / S de ressources que tout autre publié précédemment C-Control quand il n'y a pas d'expansion de RAM externe est utilisé. Le C-Control Pro offre multithreading, similaire à la C-Control II. La programmation peut être faite en C compact ou de base. Les deux langues peuvent être mélangés, comme une partie se composent de compact C et l'autre partie du logiciel de base. Comme il est un système de programmation modulaire, et les grands projets peuvent être relativement faciles à gérer. Modules universels doivent être réutilisé comme il l'a dans le C-Control II et au C-Control I sous Basic ++, peut être programmé qu'une seule fois et peut de cette façon. Cependant, le C-Control-IDE ne contient pas un simulateur comme dans C-Control I et C-Control II, mais a plutôt un matériel débogueur.

On passe par les étapes suivantes pour installer C-control Pro:



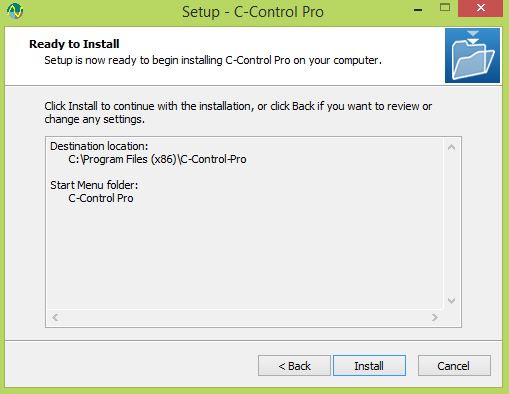
**Figure1: installation de C-Control Pro**

De là, sélectionnez "Installer à partir d'une liste ou une autre source" et cliquez sur "Next" ..

****

**Figure 2 : deuxième étapes d'installation.**

Ensuite, tapez le chemin vers le répertoire du pilote. Si le logiciel a été installé dans "C: \ Programmes" il sera chemin "C: \ Programmes Files \ C-Control-Pro ".

****

**Figure 3: chemin d'installation (suite).**

* **Langage de programmation**

  disponibles pour la programmation du C-Control C-Control II CD du logiciel de [CONRAD](https://de.wikipedia.org/wiki/Conrad_Electronic) implique [la programmation](https://de.wikipedia.org/wiki/Programmiersprache) CCBASIC et BASIC ++ (comme une composante majeure de l'environnement de développement Workbench ++). Sinon peut être utilisé pour la version C-Control I 1.1 également MBasic, le [C, comme le langage](https://de.wikipedia.org/wiki/C_(Programmiersprache))[CCCCC](https://de.wikipedia.org/w/index.php?title=CCCCC&action=edit&redlink=1) développé ou pour le projet Open Control, l'utilisation rétro compatible OCBASIC.

L'je la version C-Control 1.1 peut également investir dans [assembleur](https://de.wikipedia.org/wiki/Assemblersprache) peut être programmé. Cependant, une assemblée spéciale appelée CCASM est nécessaire parce qu'après avoir reçu CCASM démarre programmes signés sur le nouveau C-commandes pour l'I M-Unit C-Control 2.0 et C-Control Micro.

Exemples des fonctions qu' on utilises :

Main () 'Ceci est un commentaire

Fonction main ()

DEFINE As Byte i

Définir comme STRING \* 10 myString

LCD.INIT

LCD.CLEAR

POUR i = 1 à 10

Mon String = "valeur de i =" & STR (i)

LCD.PRINT myString

I PROCHAINE

LCD.OFF

FIN FONCTION

* **quelques captures du Robot (en cours...)**
* **Les perspectives**
* Un programme qui permet le robot de ce déplacé dans la ligne droite
* Un programme qui permet le robot de faire un trajectoire sous forme d'un cercle.
* Un programme qui permet le robot de faire un trajectoire sous fourme d'un carré.
* notre robot va détecté le mouvement de la main pour changer la direction.

**Conclusion Générale**

on va essayé de attendre les perspectives mentionner précédemment.

**BIBLIOGRAHIE**

[1] : <http://www.jeunesse-et-science.be/sites/default/files/document/esc_84_3_2.pdf>

[2] : <http://www-igm.univ-mlv.fr/~dr/XPOSE2002/robotique/chapitres/Historique.htm>

[3] :<http://users.polytech.unice.fr/~pmasson/Enseignement/Arduino%20Pro_Bot128%20-%20MASSON%20V11.pdf>

[4] : <http://liris.cnrs.fr/laetitia.matignon/index/coursL1robotique.pdf>

[5] : <http://de.wikipedia.org/wiki/C-Control>