

Un microcontrôleurMBED au service d'une interface LabVIEW



Ou comment utiliser un microcontrôleur programmé en C/C++ compilé sous MBED couplé à une interface labview.

Journée AlpesVIEW Grenoble au GE2LAB le 10 octobre 2019



David Charalampous
Laboratoire ISMO (Orsay – Région parisienne)
CNRS – UPSud

Plan de présentation

- Présentation du Contexte : Application, Description
- Mbed : les microcontrôleurs
- Mbed : le compilateur
- Interfaçage avec l'ordinateur
- Interfaçage avec Labview
- Pour aller plus loin...

Présentation du contexte : Application

2013/2014 au laboratoire MSC-UMR 7057

Réalisation pompe hydrostatique automatisée :

- Une seringue sur une translation verticale fait varier une hauteur d'eau
- ⇒ Exercice d'une pression/dépression sur un échantillon de cellules sous microscope

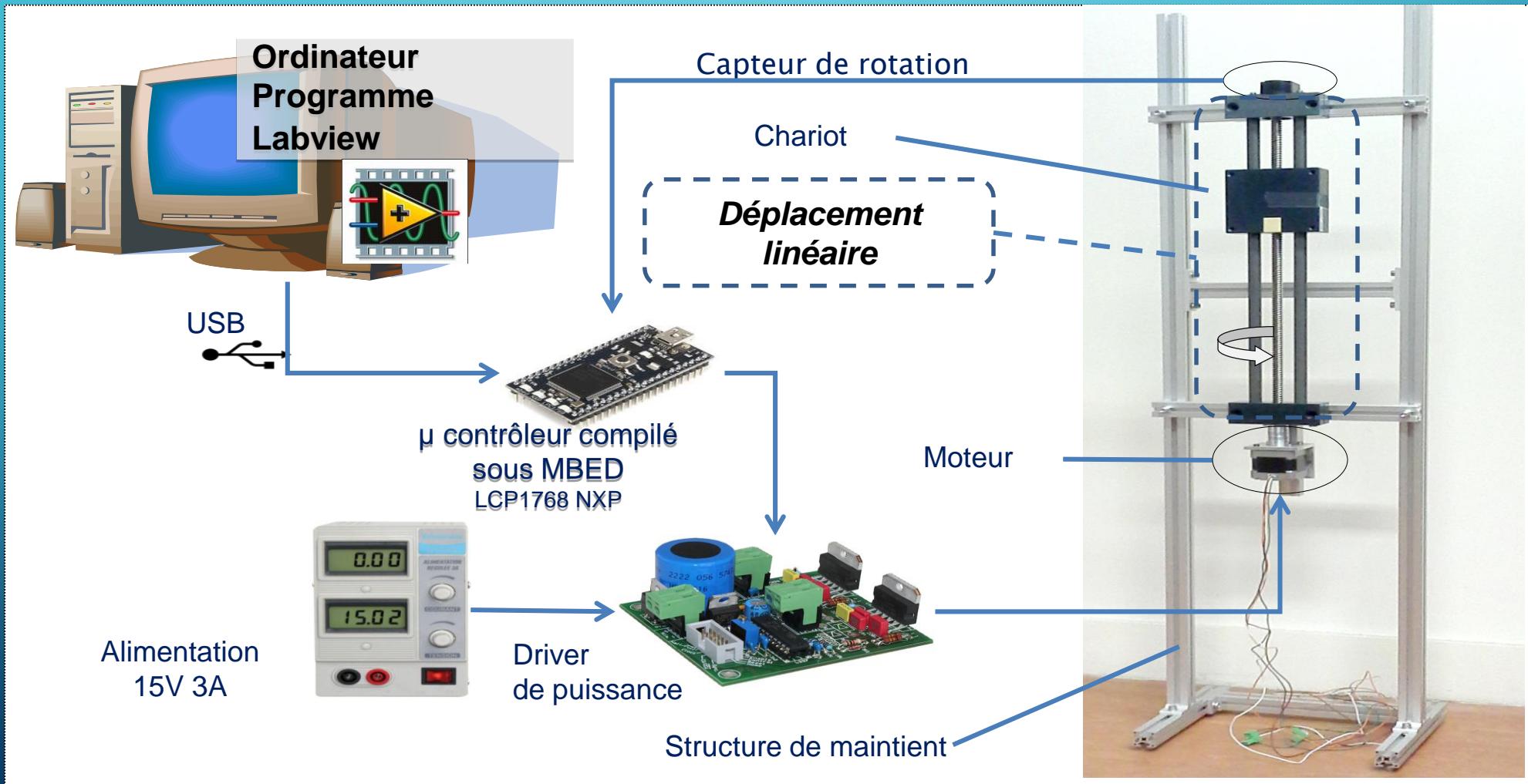
Demandeurs : François Graner et Sham Tlili



Spécifications :

- Déplacement linéaire de 30 cm
- Moteur pas à pas de 200 pas/tour
- Encodeur 200 traits => résolution en quadrature de 1/800^e de tour
- Choix du microcontrôleur : LCP1768 avec compilation MBED
- Communication via port COM virtuel 460 800 Bauds/s

Présentation du Contexte : Description



Mbed : les microcontrôleurs

- ✓ Environ 173 cartes de microcontrôleurs **ARM 32 bits** compatibles MBED, fabriquées surtout par ST Micro électronique et NXP.
- ✓ Langage écrit en C/C++ contenant de l'objet
- ✓ Fonctions de bases de haut niveau : timer, front, PWM, E/S...
- ✓ Une bibliothèque riche faites aussi par les utilisateurs : drivers de caméras, écrans capteurs, actionneurs, ...
- ✓ Pour programmer : se créer un compte en ligne, connecter sa carte par USB, programmer, compiler et c'est parti!

DAC I2C

PWM ADC

Gestion de carte micro-SD

SPI
Ecrans tactile intégré

Ethernet

Caméra

.....



NXP : LPC1768



NXP : FRDM-K64F

MBed : le compilateur en ligne

The screenshot shows the mbed online compiler interface. On the left, there's a sidebar titled "Program Workspace" containing a list of various mbed programs. In the center, there's a main workspace with a code editor showing C/C++ code for a motor driver. A red dashed box highlights the "Compile" button at the top of the editor. A red arrow points from the text "Touche pour compiler le programme" to this button. Below the code editor, a green dashed box highlights the sidebar area, with the text "Barre où sont tous vos programmes" pointing to it. At the bottom, there's a "Compile output" section.

Touche pour compiler le programme

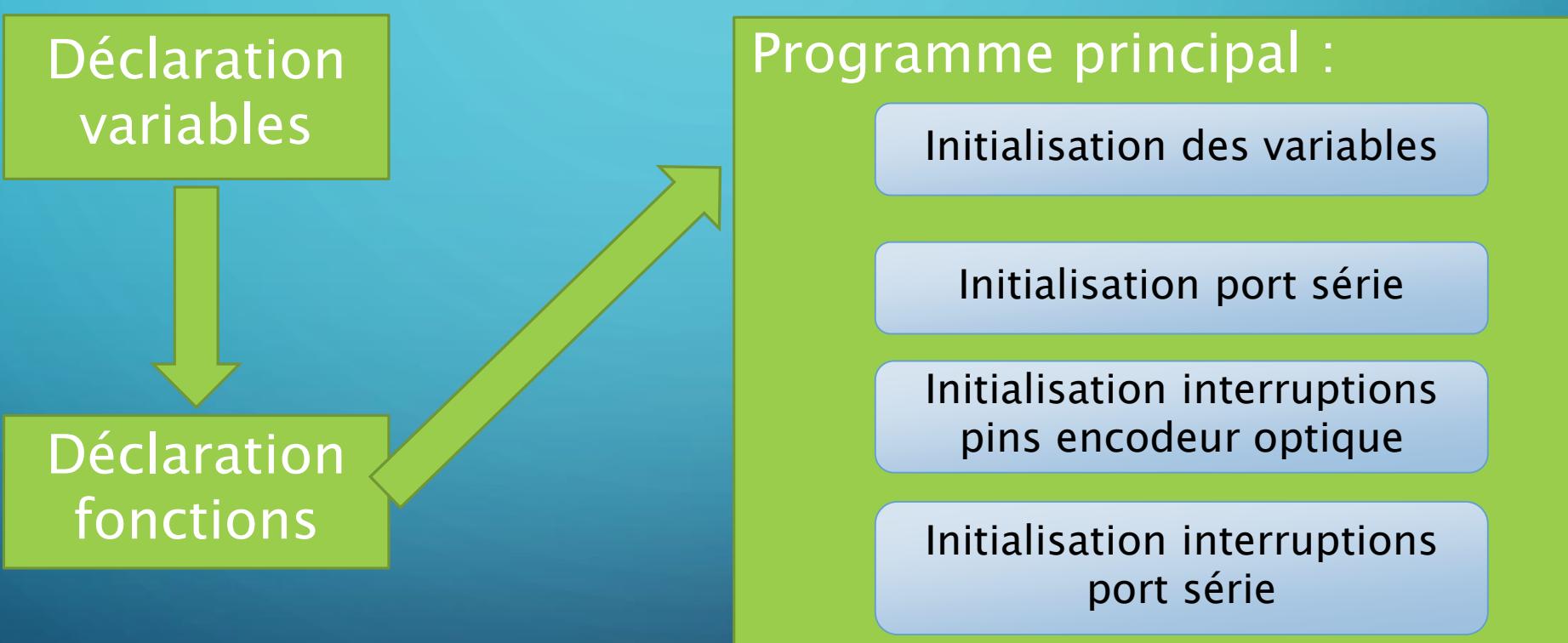
Barre où sont tous vos programmes

- Interface web : le code est conservé en ligne
- code écrit en C/C++
- Possibilité d'importer des librairies et de partager son code
- le C++ permet d'avoir/créer des objets

Interfaçage avec l'ordinateur

- Bibliothèque « Serial » : création d'un port COM virtuel transitant par le port USB du Microcontrôleur.
- Sous windows il faut installer un driver.
- Pour certaines cartes il est parfois nécessaire d'installer un firmware sur la carte en mode « bootloader ».
- Attention à la compatibilité avec window 10

Programme sur le microcontrôleur



Interfaçage avec Labview : un langage à écrire

Bout de code dans programme MBED :

Serial pc(USBTX,USBRX); //tx, rx : Ici l'objet « pc » est relié au port usb du microcontrôleur.

Dans le code C/C++ j'ai créé les instructions qui sont principalement :

POW <x> : x='0' power off / x ='1' power on

PAS <p> : p ='0' : pas entier / p ='1' : demi-pas

VIT <f> : f = fréquence de pulsation des pas ou demi-pas

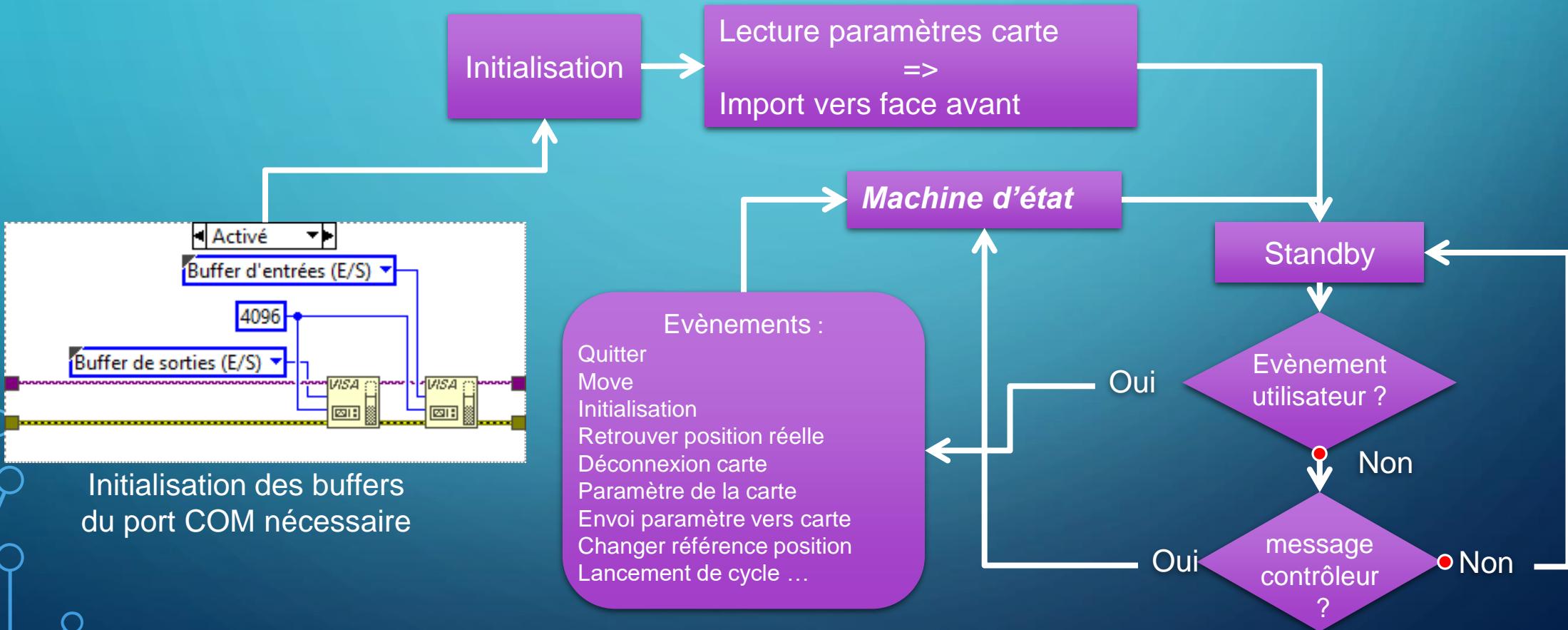
DRCT <d> : d ='0' ou '1' : donne la direction du chariot

ZERO : remet à zéro le compteur

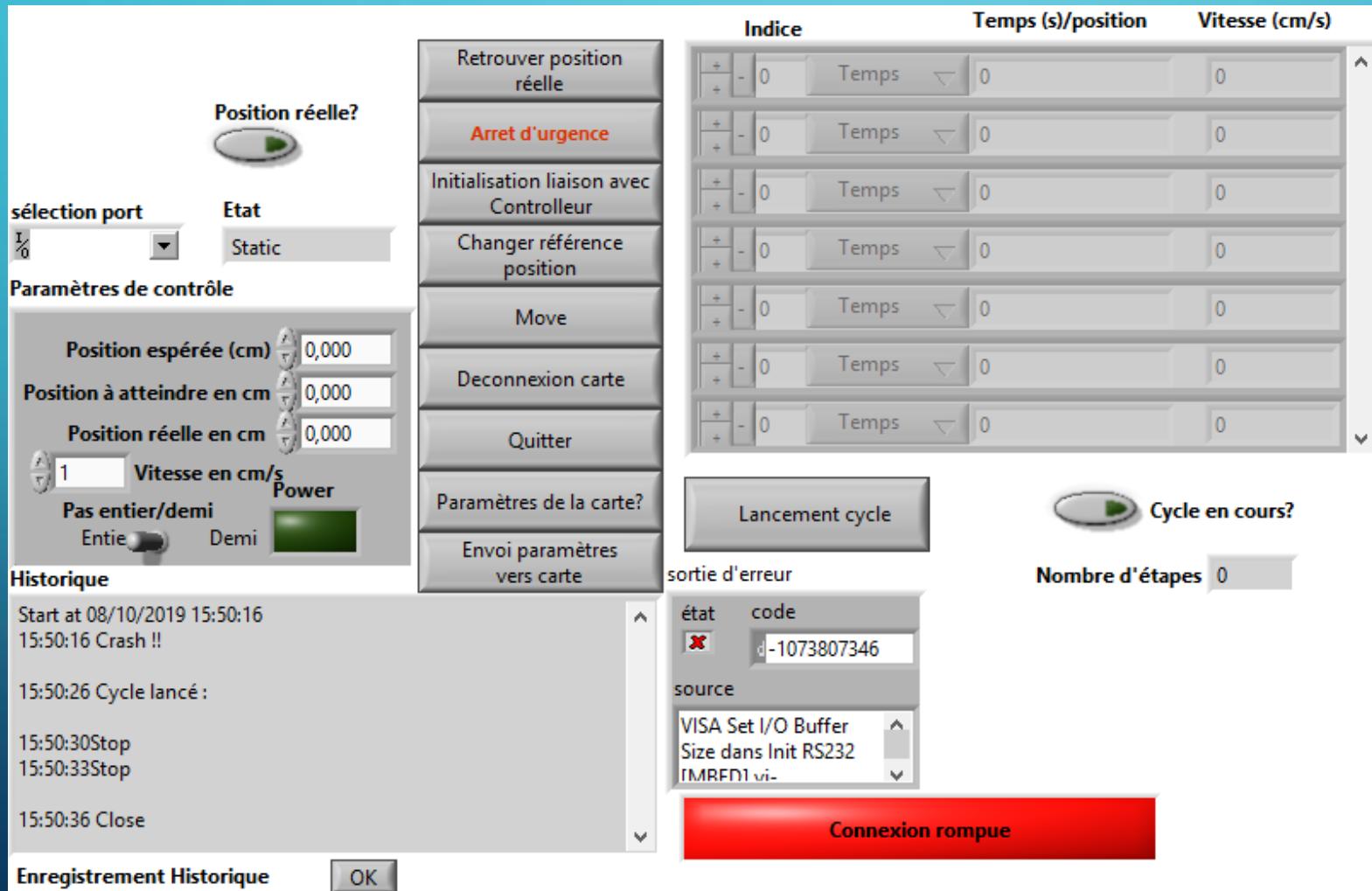
POS? : interroge la position relative

MOVE <dx> : déplacement vers dx

Diagramme du code Labview



Interface utilisateur Labview



- Paramètres de contrôles
- Gestion de tâches via boutons booléens -> déclenche un évènement
- Possibilité de se reconnecter à la carte : gestion de l'erreur timeout
- Lancement de cycle avec plusieurs positions puis temps mort
- Historique : suivi des évènements horodaté

Bilan et prospectives

- Difficultés :
 - il faut tout faire de A-Z
 - L'utilisation de codes LabVIEW RPC limités : one shot
 - Acquérir une certaine maîtrise du C/C++
 - Parfois il faut écrire ses propres drivers
 - Pas de debugger
- Intérêts :
 - Outil une fois maîtrisé -> facilement interfaçable
 - Plein de bibliothèques disponibles
 - Couteau-suisse instrumental très utile pour une 1ère phase de prototypage
 - Langage C/C++ de haut niveau simplifiant l'écriture du code
 - Intérêt pour mon nouveau poste 100% LabVIEW
 - Coût modique en particulier pour les motorisation via moteurs pas à pas

Prospectives :

- Connexion Ethernet possible alternative plus intéressante que le port RS232
- Communication en mode RPC => utilisation de programmation objet sur le micro le code « sort » du microcontrôleur (existe aussi avec d'autres familles notamment les PICs)
- Possibilité de créer des objets LabVIEW pour encapsuler la gestion des drivers MBD-LabVIEW

Lien web du code Mbed partagé sur ce projet :

http://mbed.org/users/dcharala/code/Dplct_Lin_1_3_2/

- Merci de votre attention...