

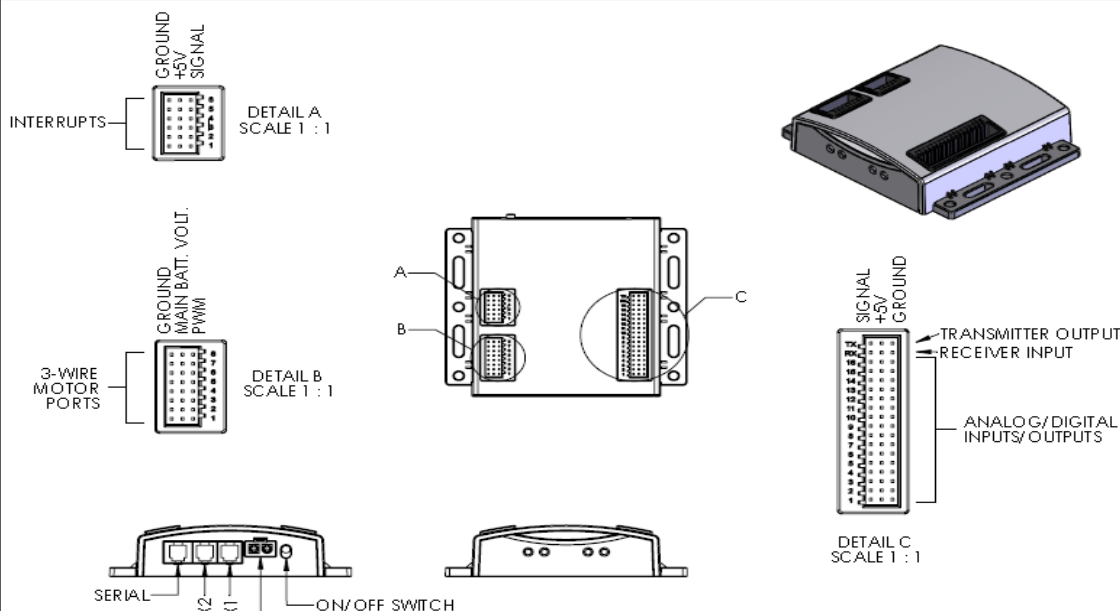
2017

TEST DES DIFFÉRENTS COMPOSANTS DU ROBOT DE TYPE VEX

Activités pratiques guidées

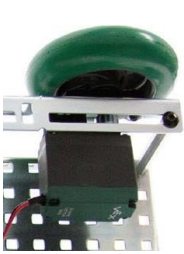
Objectifs d'apprentissage :

- Connaître programmer la plupart des composants du robot VEX
- Connaître les différents composants du robot VEX

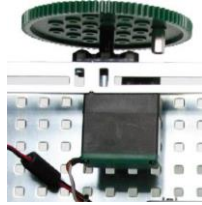


Pratique #1 : Faire tourner les moteurs

1. Le moteur et servomoteur



Moteur



Servomoteur



Batterie



CORTEX + Câble



PIC + Câble

Avant de commencer, vous devez :

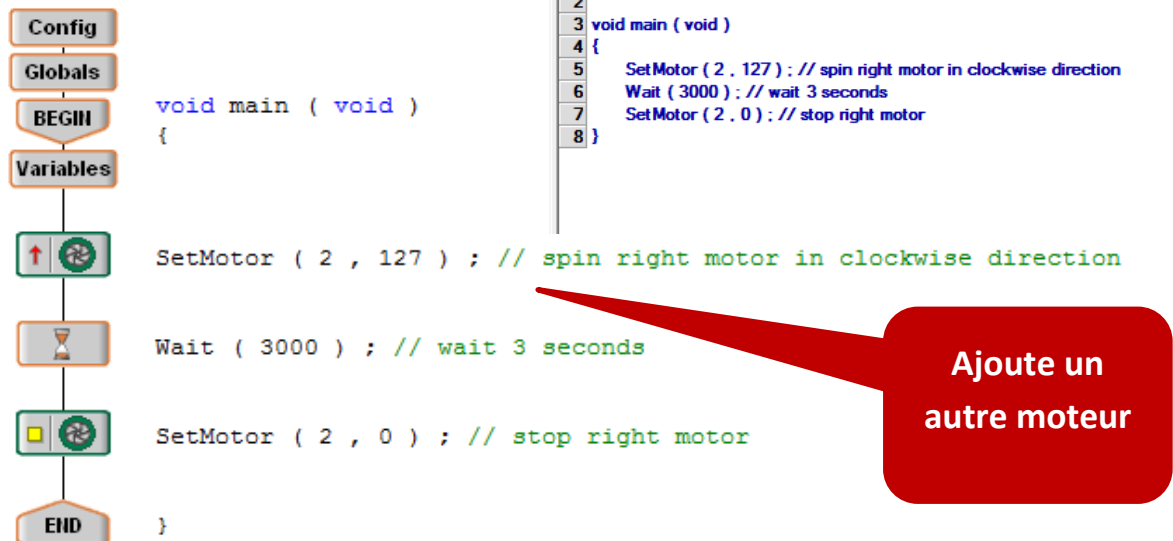
- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter les moteurs aux **ports 2 et 3** du PIC/Cortex indiqués dans le code ci-dessous.
- C. Connecter le servomoteur au **port 6** du PIC/Cortex indiqué dans le code ci-dessous.
- D. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB.

Après les connexions des différents composants et les vérifications, vous devez :

- A. Démarrer de votre ordinateur le programme « EasyC V5 »
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »

B. Programmer les moteurs et le servomoteur en déplaçant et en plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme. Puis enregistrer votre programme.

1) Programmation des Moteurs :



2) Programmation du Servomoteur :

Ajoute le servomoteur

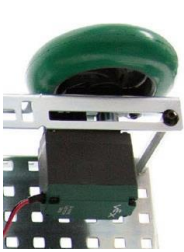
C. Tester votre programme et faites la documentation interne

D. Ajouter votre touche personnelle à ce code (ex, faites tourner les moteurs dans le même sens vers l'avant, vers l'arrière et dans les sens contraires)

2. Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.

Pratique #2 : Bumper fait tourner les moteurs

1. Le moteur et bumpers :



Moteur



Bumpers



Batterie



CORTEX + Câble



PIC + Câble

Avant de commencer, vous devez :

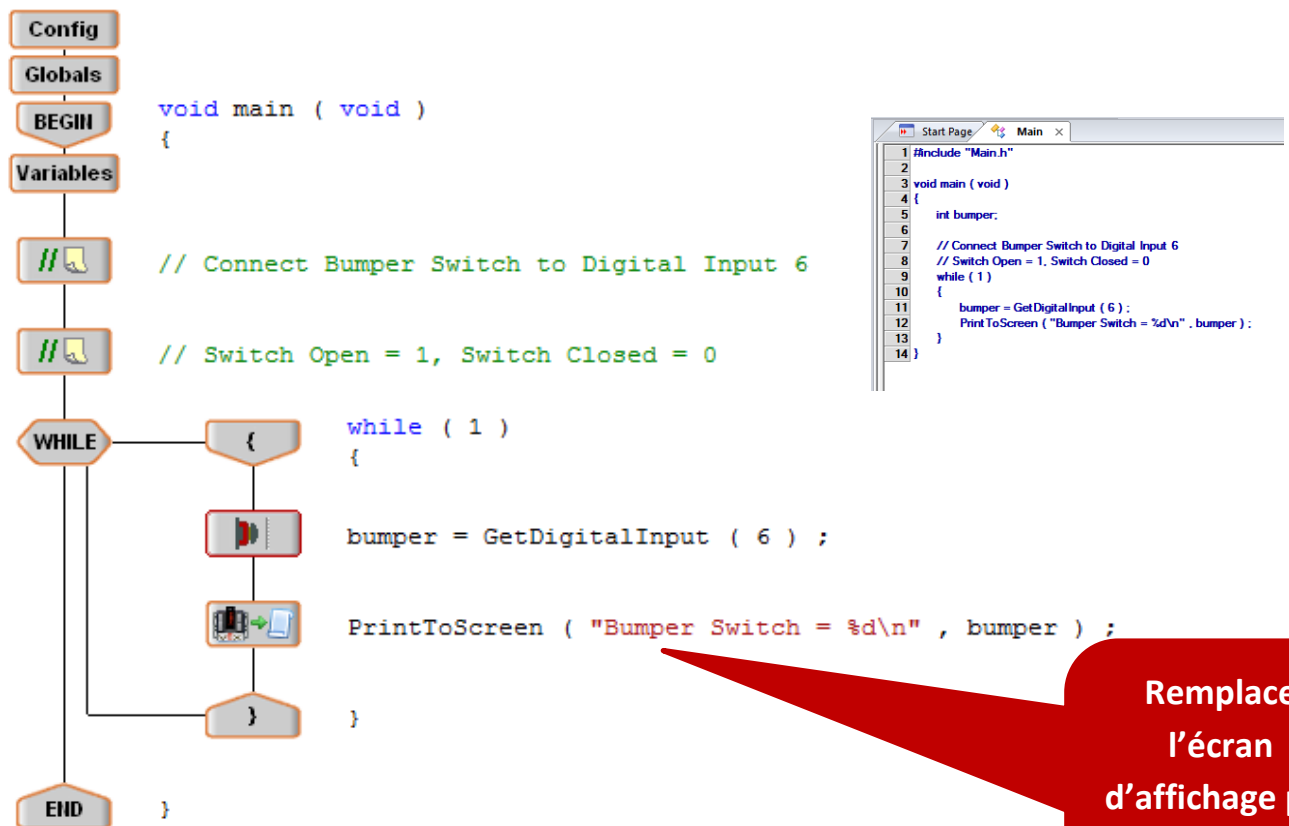
- A. Vous s'assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- C. Connecter les bumpers au **port 6 (DIGITAL)** du PIC/Cortex indiqué dans le code ci-dessous.
- D. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB

Après les connexions des différents composants et les vérifications, vous devez :

- A. Démarrer de votre ordinateur le programme « EasyC V5 »
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »

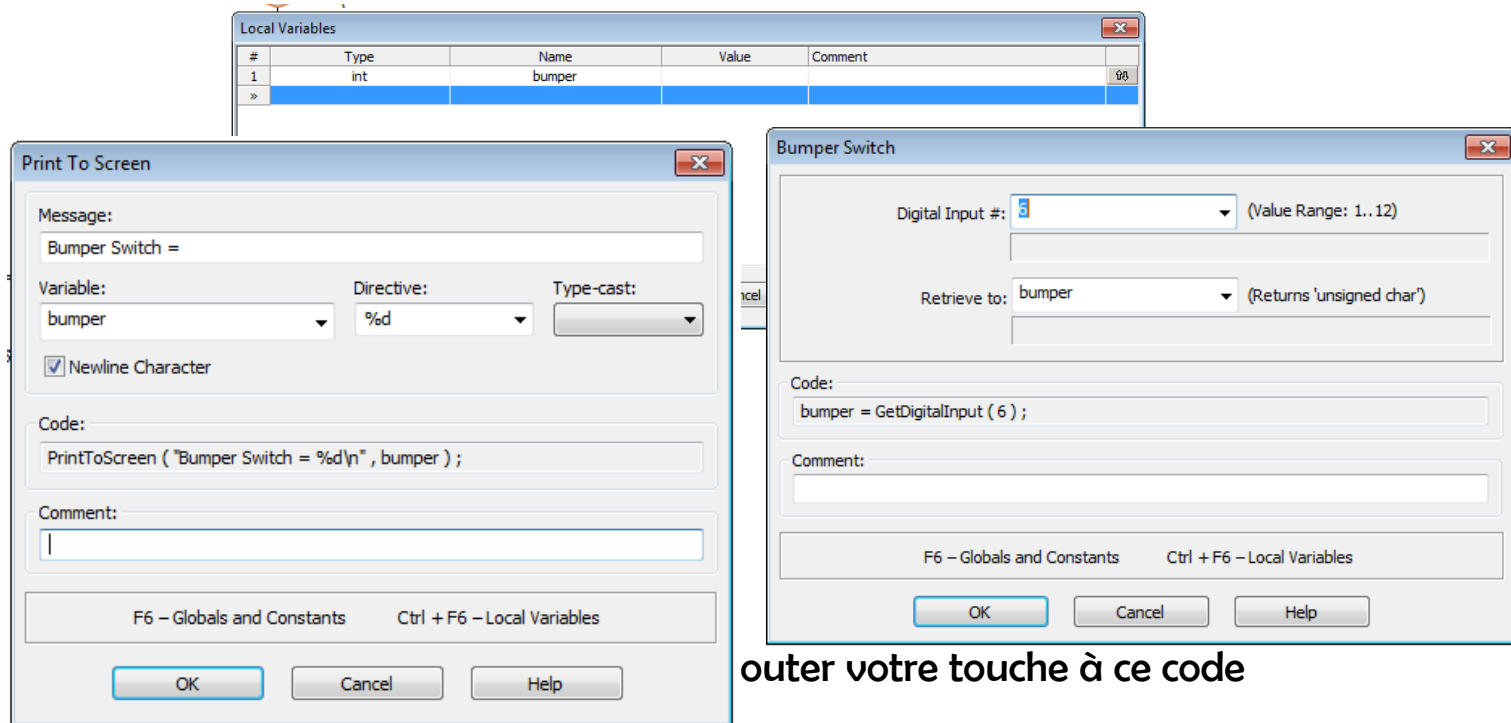
B. Cliquer sur « Macros and constants » et dans la section « Global Variables » déclarer une variable et la nommer « bumper » de type « int ».

C. Programmer les moteurs et le bumpers en déplaçant et en plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme. Puis enregistrer votre programme.



Utilise les images ci-dessous pour configurer les bumpers

Remplace
l'écran
d'affichage par
les moteurs
dans une
structure de
condition if



outer votre touche à ce code

2. Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.

Pratique #3 : capteur de limite fait tourner les moteurs

1. Le moteur et capteur de limite:



Moteur



Limit Switch



Batterie



CORTEX + Câble



PIC + Câble

Avant de commencer, vous devez :

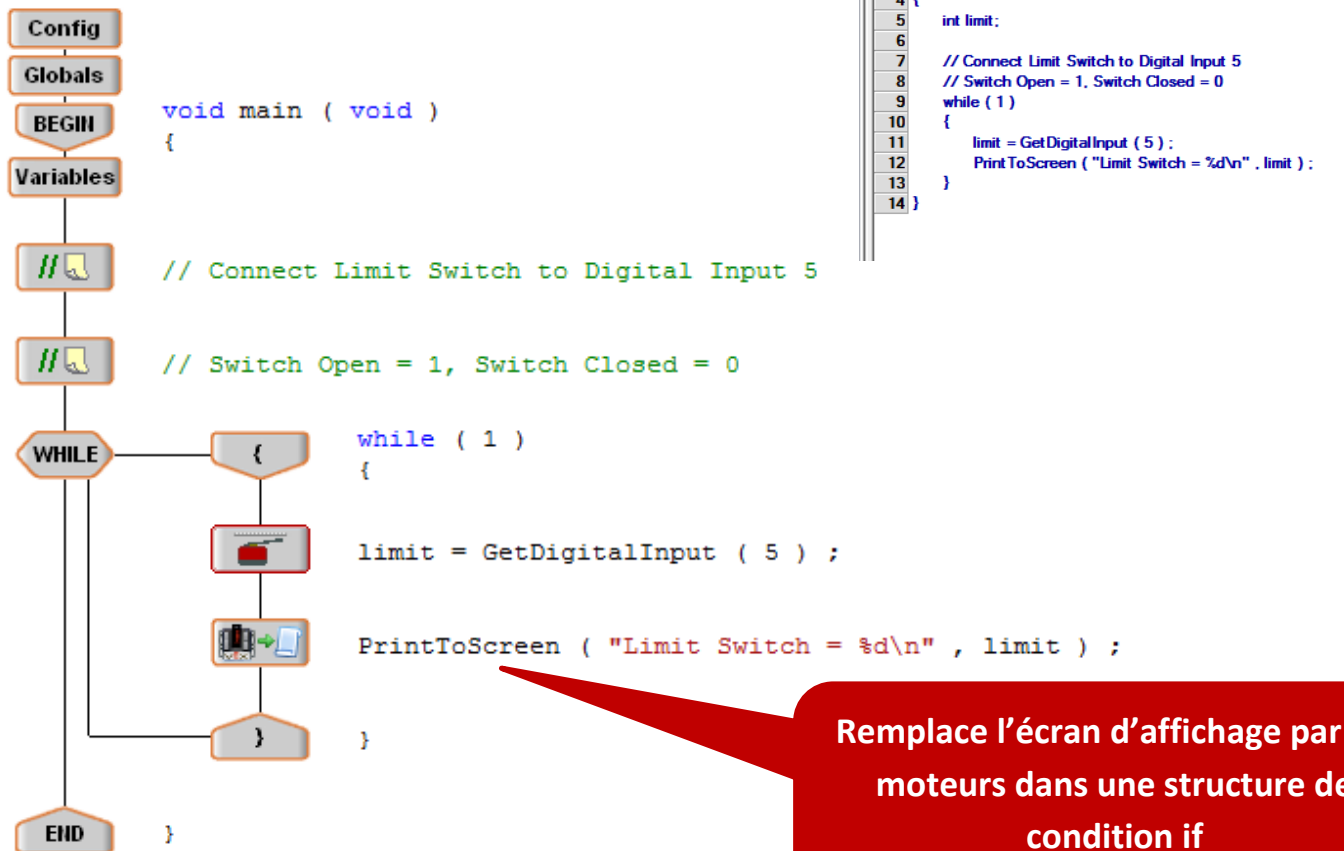
- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- C. Connecter le capteur de limite au **port 5 (DIGITAL)** du PIC/Cortex indiqué dans le code ci-dessous.
- D. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB.

Après les connexions des différents composants et les vérifications, vous devez :

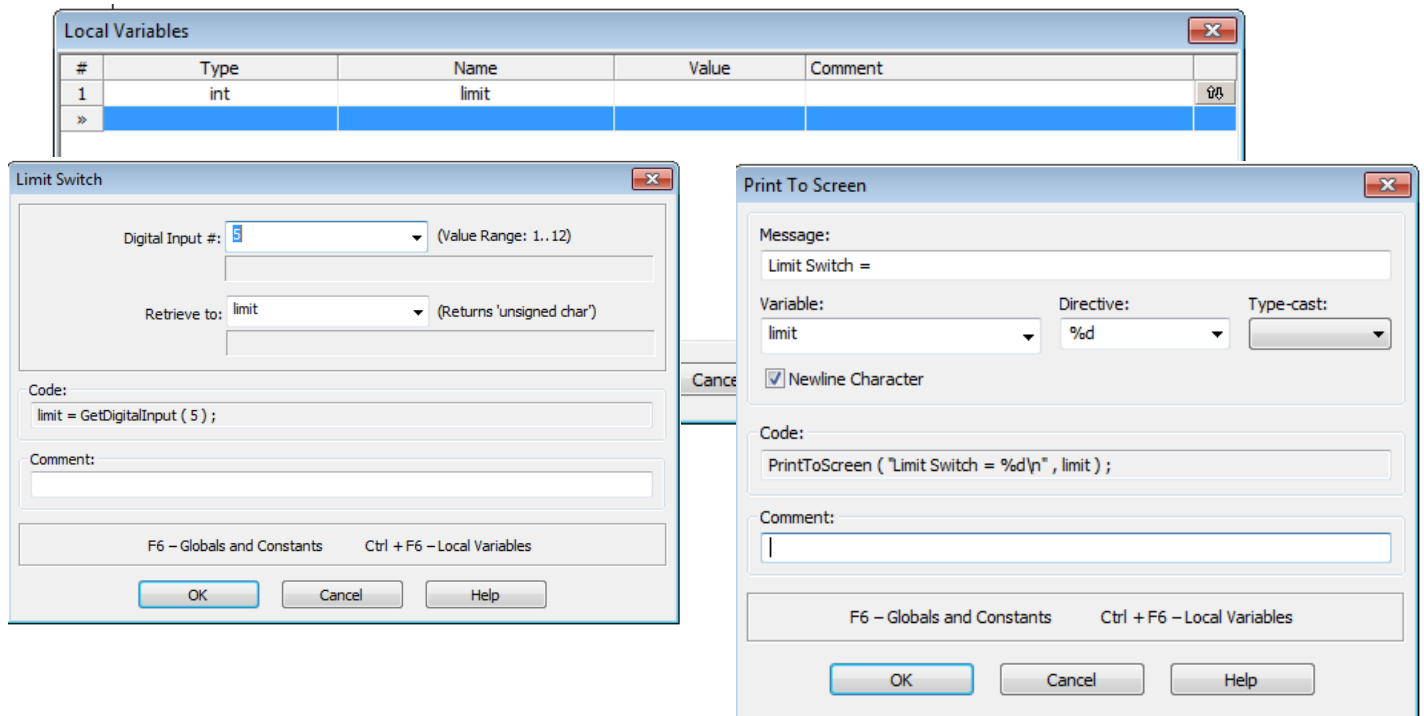
- A. Démarrer de votre ordinateur le programme « EasyC V5 »
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »

B. Cliquer sur « Macros and constants » et dans la section « Global Variables » déclarer une variable qui se nomme « limite » de type « int ».

C. Programmer les moteurs et le capteur de limite en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Utilise les images ci-dessous pour configurer le capteur de limite



Tester votre programme et ajouter votre touche à ce code

2.Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.

Pratique #4 : Capteur ultrasonique fait tourner les moteurs

1. Le moteur et capteur ultrasonique:



Moteur



Capteur ultrasonique



Batterie



CORTEX + Câble



PIC + Câble

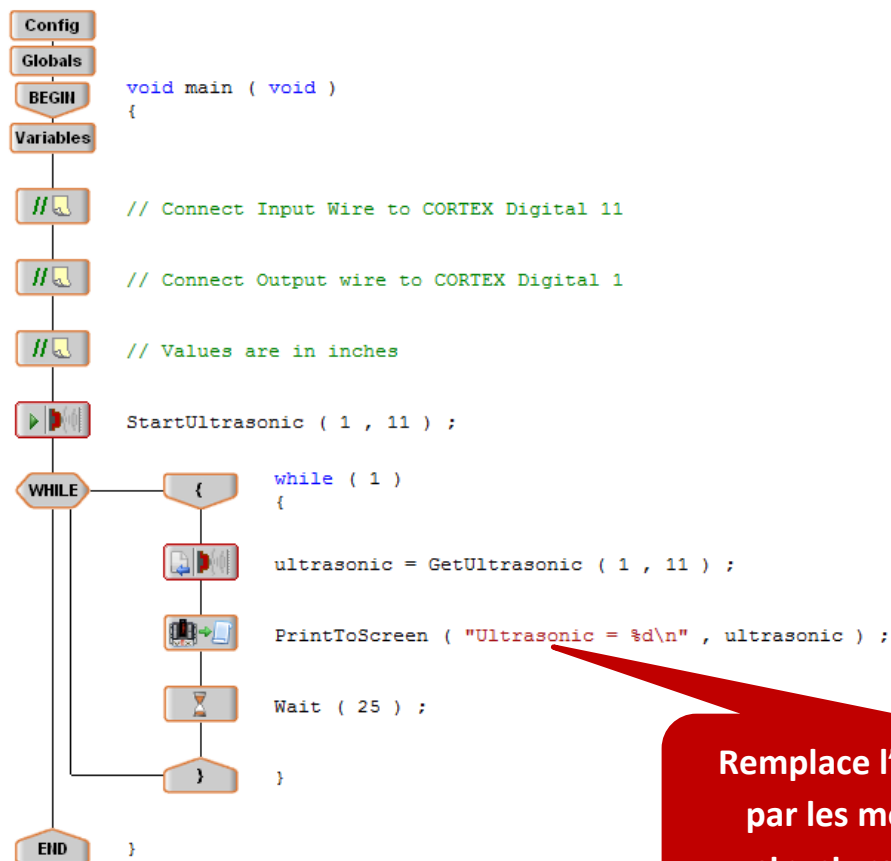
Avant de commencer, vous devez :

- A. Vous s'assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- C. Connecter **le capteur ultrasonique - Input** au **port 11 (DIGITAL)** et **le capteur ultrasonique - Output** au **port 1 (DIGITAL)** du PIC/Cortex indiqués dans le code ci-dessous.
- D. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB

Après les connexions des différents composants et les vérifications, vous devez :

- A. Démarrer de votre ordinateur le programme « EasyC V5 »
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »

- B. Cliquer sur « Macros and constants » et dans la section « Global Variables » déclarer une variable qui se nomme « ultrasonic » du type « int ».
- C. Programmer les moteurs et le capteur ultrasonique en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Utilise les images ci-dessous pour configurer le capteur ultrasonique

capteur

Local Variables

#	Type	Name	Value	Comment
1	int	ultrasonic		
>>				

OK Delete Cancel Help

Ultrasonic Sensor

Select command:

☒ Start
☐ Get (inch)
☐ Get (cm)
☐ Stop

Interrupt Port #: 1 (Value Range: 1..12)

Output Port #: 11 (Value Range: 1..12)

Code:
StartUltrasonic (1, 11);

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Ultrasonic Sensor

Select command:

☐ Start
☒ Get (inch)
☐ Get (cm)
☐ Stop

Interrupt Port #: 1 (Value Range: 1..12)

Output Port #: 11 (Value Range: 1..12)

Retrieve to: ultrasonic (Returns 'unsigned int')

Code:
ultrasonic = GetUltrasonic (1, 11);

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Print To Screen

Message:
Ultrasonic =

Variable: ultrasonic Directive: %d Type-cast:

☒ Newline Character

Code:
PrintToScreen ("Ultrasonic = %d\n", ultrasonic);

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Wait

Wait [msec]:
Wait (25);

Add Variable: Add Operator:

Code:
Wait (25);

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

- D. Tester votre programme et ajouter votre touche à ce code**
- 2. Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.**

Pratique #5 : Line Follower fait tourner les moteurs

1. Le moteur et Line Follower:



Moteur



Line Follower



Batterie



CORTEX + Câble



PIC + Câble

Avant de commencer, vous devez :

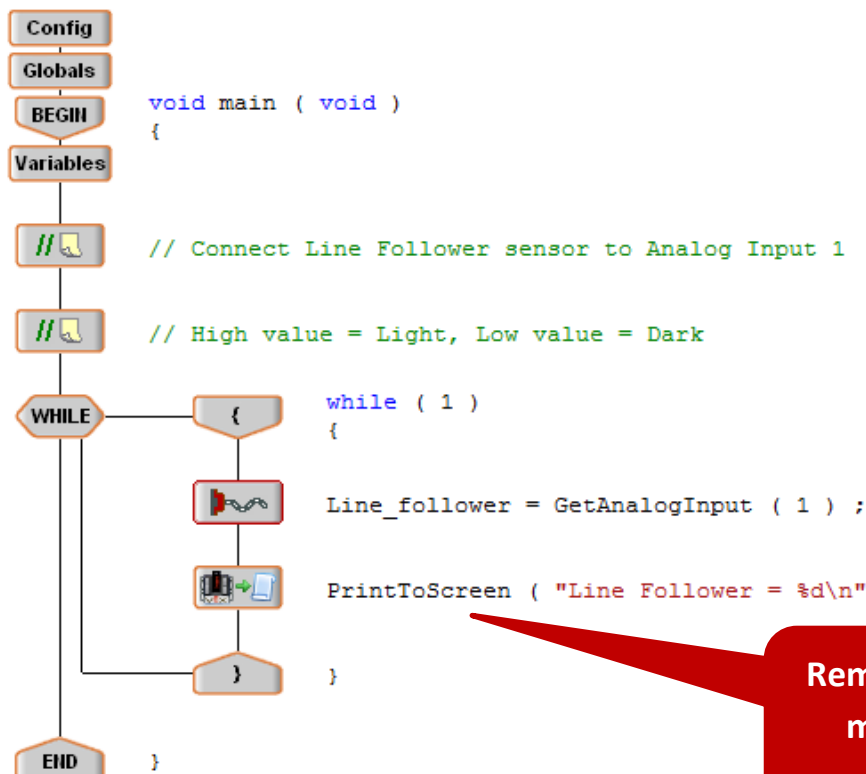
- A. Vous s'assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- C. Connecter le **Line Follower** au **port 1** (ANALOGUE) du PIC/Cortex indiqués dans le code ci-dessous.
- D. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB

Après les connexions des différents composants et les vérifications, vous devez :

- A. Démarrer de votre ordinateur le programme « EasyC V5 »
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »

B. Cliquer sur « Macros and constants » et dans la section « Global Variables » déclarer une variable que se nomme « line_follower » du type « int ».

C. Programmer les moteurs et le Line Follower en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Remplace l'écran d'affichage par les moteurs dans une structure de condition if

Utilise les images ci-dessous pour configurer **line Follower**

#	Type	Name	Value	Comment
1	int	Line_follower		
»				

OK Delete Cancel Help

Line Follower

Analog Input #: (Value Range: 1..8)

Retrieve to: (Returns 'unsigned int')

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

Print To Screen

Message:

Variable: Directive: Type-cast:

☒ Newline Character

Code:

Comment:

D. Tester votre programme et ajouter votre touche à ce code

2. Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.

Pratique #6 : Capteur de lumière fait tourner le moteur

1. Le moteur et capteur de lumière :



Moteur



Capteur de lumière



Batterie



CORTEX + Câble



PIC + Câble

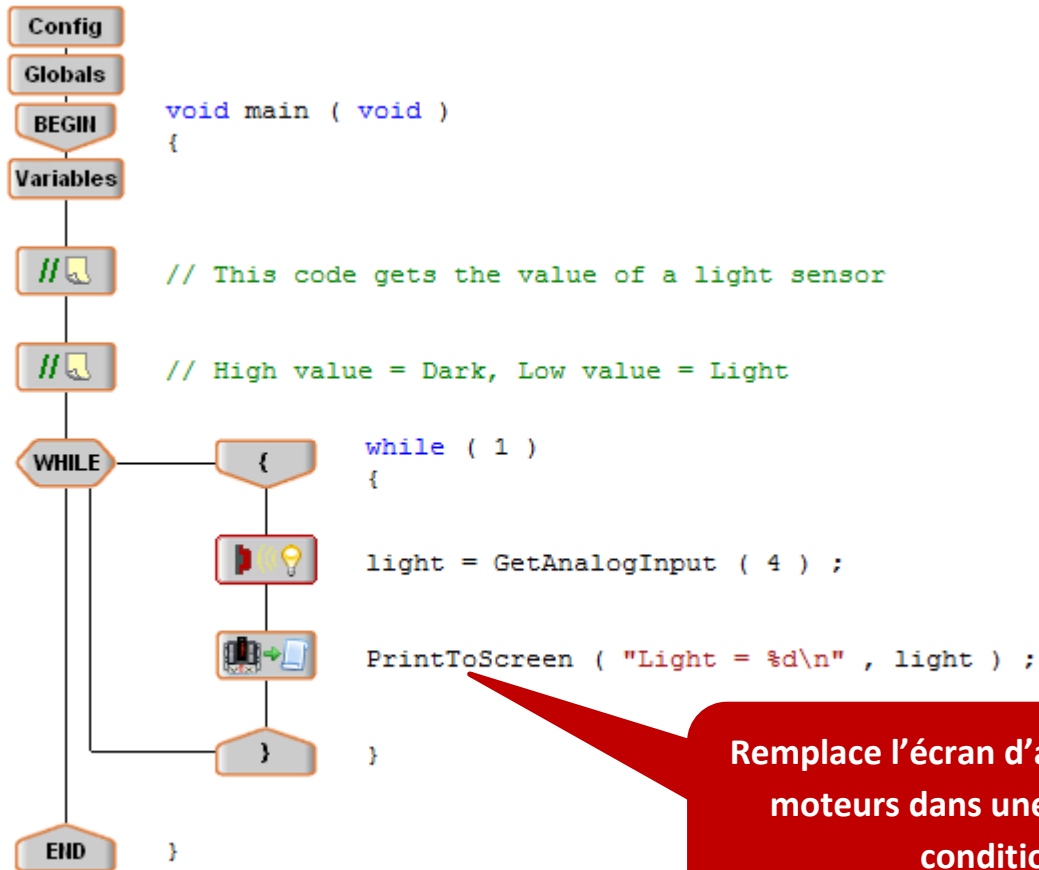
Avant de commencer, vous devez :

- Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- Connecter le capteur de lumière au **port 2** (DIGITAL) du PIC/Cortex indiqué dans le code ci-dessous.
- Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB

Après les connexions des différents composants et les vérifications, vous devez :

- Démarrer de votre ordinateur le programme « EasyC V5 »
- Cliquer sur « Macros and constants » et dans la section « Global Variables » déclarer une variable que se nomme « lumière » du type « int ».

C. Programmer les moteurs et le capteur de lumière en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Light Sensor ✕

Analog Input #: (Value Range: 1..8)

Retrieve to: (Returns 'unsigned int')

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

Print To Screen ✕

Message:

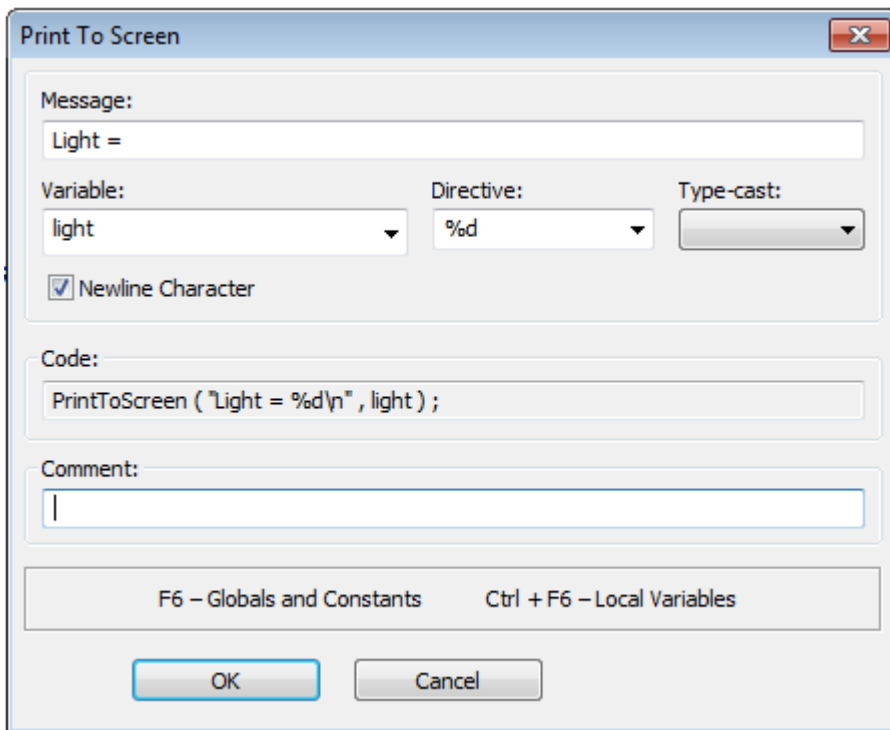
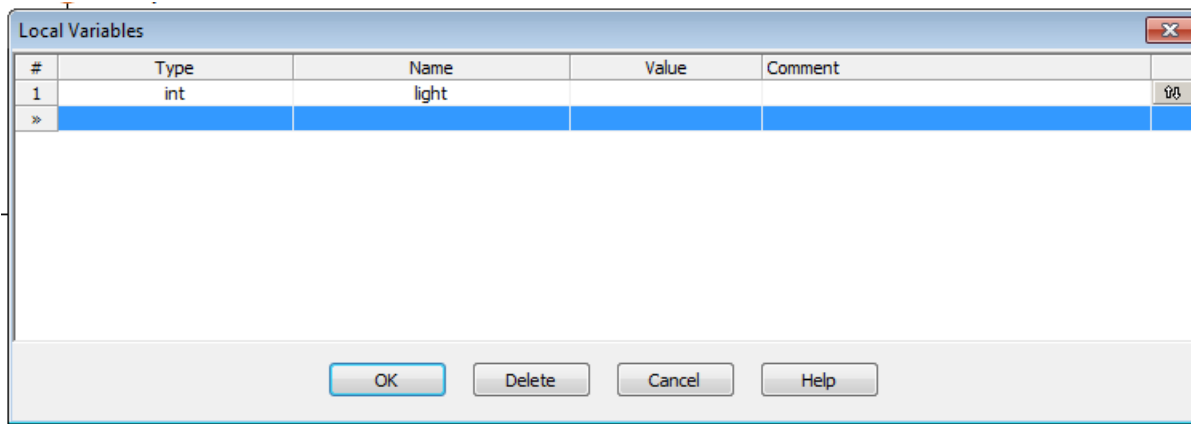
Variable: Directive: Type-cast:

☒ Newline Character

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

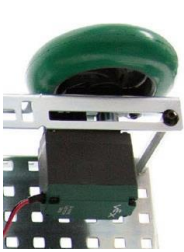


D. Tester votre programme et ajouter votre touche à ce code

2. Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.

Pratique #7 : Potentiomètre fait tourner les moteurs

1. Le moteur et Potentiomètre :



Moteur



Potentiomètre



Batterie



CORTEX + Câble



PIC + Câble

Avant de commencer, vous devez :

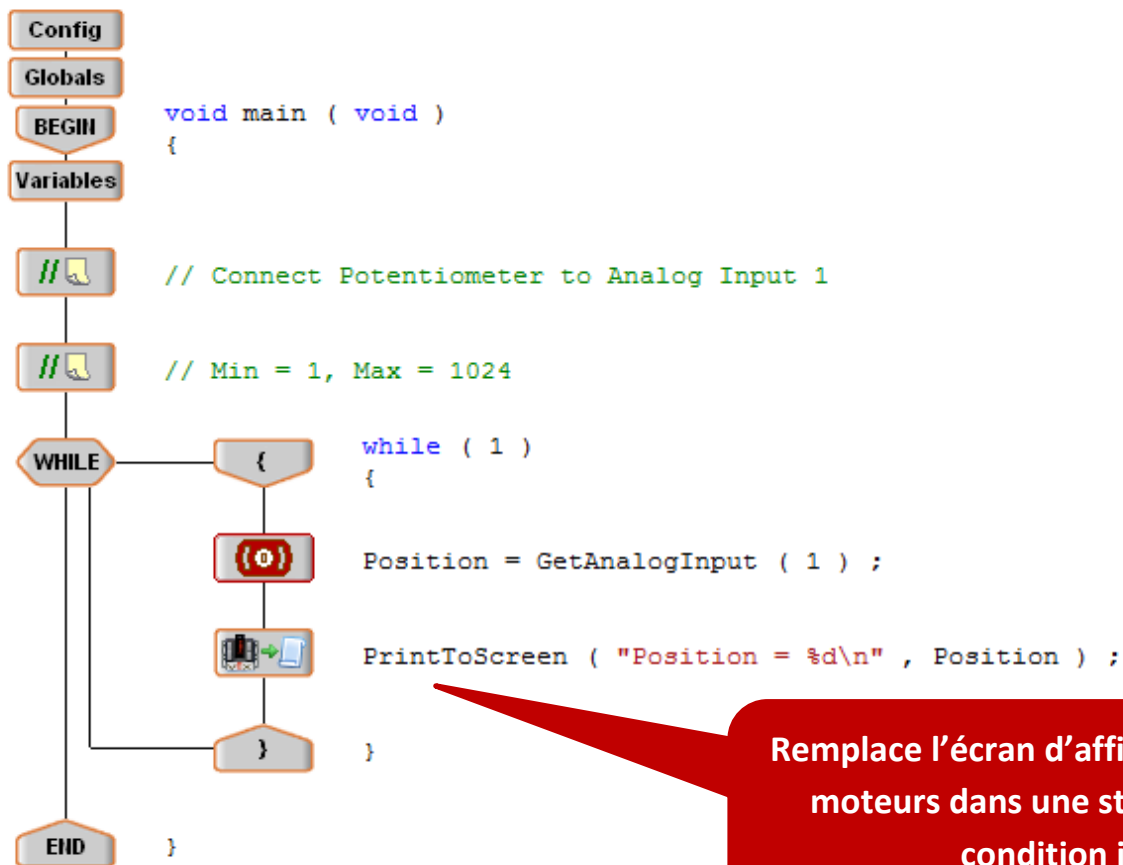
- A. Vous s'assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- C. Connecter le Potentiomètre au **port 1 (ANALOGUE)** du PIC/Cortex indiqué dans le code ci-dessous.
- D. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB

Après les connexions des différents composants et les vérifications, vous devez :

- A. Démarrer de votre ordinateur le programme « EasyC V5 »
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »

B. Cliquer sur « Macros and constants » et dans la section « Global Variables » déclarer une variable que se nomme « Position » du type « int ».

C. Programmer les moteurs et le Potentiomètre en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Local Variables

#	Type	Name	Value	Comment
1	int	Position		
»				

OK Delete Cancel Help

Local Variables

#	Type	Name	Value	Comment
1	int	Position		
»				

OK Delete Cancel Help

Potentiometer

Analog Input #: (Value Range: 1..8)

Retrieve to: (Returns 'unsigned int')

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Print To Screen

Message:
Position =

Variable: Position Directive: %d Type-cast:

☒ Newline Character

Code:
PrintToScreen ("Position = %d\n" , Position) ;

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

```
1 #include "Main.h"
2
3 void main ( void )
4 {
5     int Position;
6
7     // Connect Potentiometer to Analog Input 1
8     // Min = 1, Max = 1024
9     while ( 1 )
10    {
11        Position = GetAnalogInput ( 1 ) ;
12        PrintToScreen ( "Position = %d\n" , Position ) ;
13    }
14 }
```

D. Tester votre programme et ajouter votre touche à ce code

2. Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.

Pratique #8 : mesurer la vitesse et le sens de rotation

1. Le moteur et codeur optique



Moteur



Codeur optique



Batterie



CORTEX + Câble



PIC + Câble

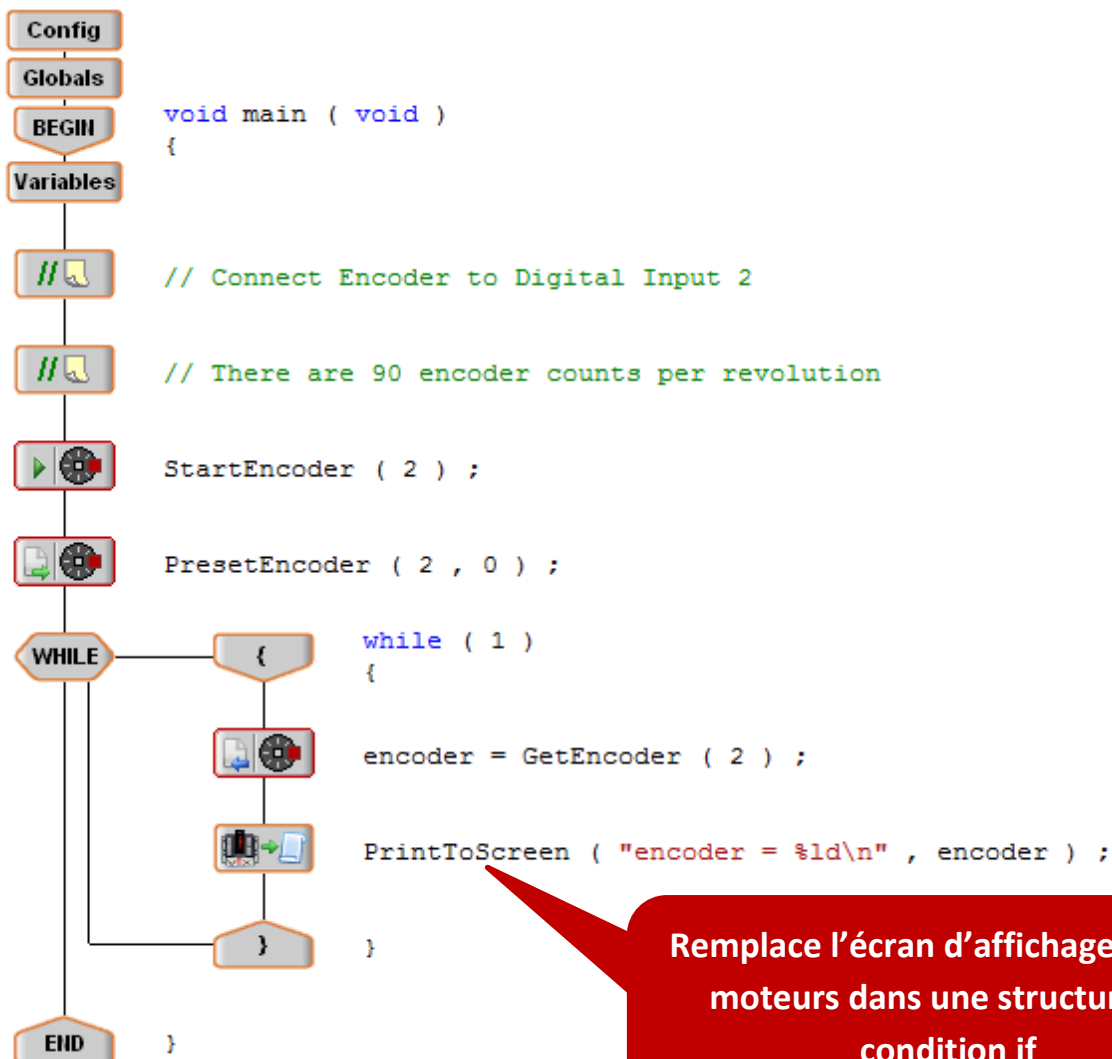
Avant de commencer, vous devez :

- A. Vous s'assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- C. Connecter le codeur optique **aux ports 1 et 2 (DIGITAL)** du PIC/Cortex indiqués dans le code ci-dessous.
- D. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB

Après les connexions des différents composants et les vérifications, vous devez :

- A. Démarrer de votre ordinateur le programme « EasyC V5 ».
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »

- B. Cliquer sur « Macros and constants » et dans la section « Global Variables » déclarer une variable que se nomme « encoder » du type « unsigned long ».
- C. Programmer les moteurs, le codeur optique et l'écran LCD en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Local Variables				
#	Type	Name	Value	Comment
1	unsigned long	encoder	0	
»				

Optical Encoder

Select command:

☒ Start
☐ Preset
☐ Get
☐ Stop

Interrupt Port #: 2 (Value Range: 1..12)

Code:
StartEncoder (2) ;

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Optical Encoder

Select command:

☐ Start
☒ Preset
☐ Get
☐ Stop

Interrupt Port #: 2 (Value Range: 1..12)

Preset to: 0

Code:
PresetEncoder (2 , 0) ;

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Optical Encoder

Select command:

☐ Start
☐ Preset
☒ Get
☐ Stop

Interrupt Port #: 2 (Value Range: 1..12)

Retrieve to: encoder (Returns 'unsigned long')

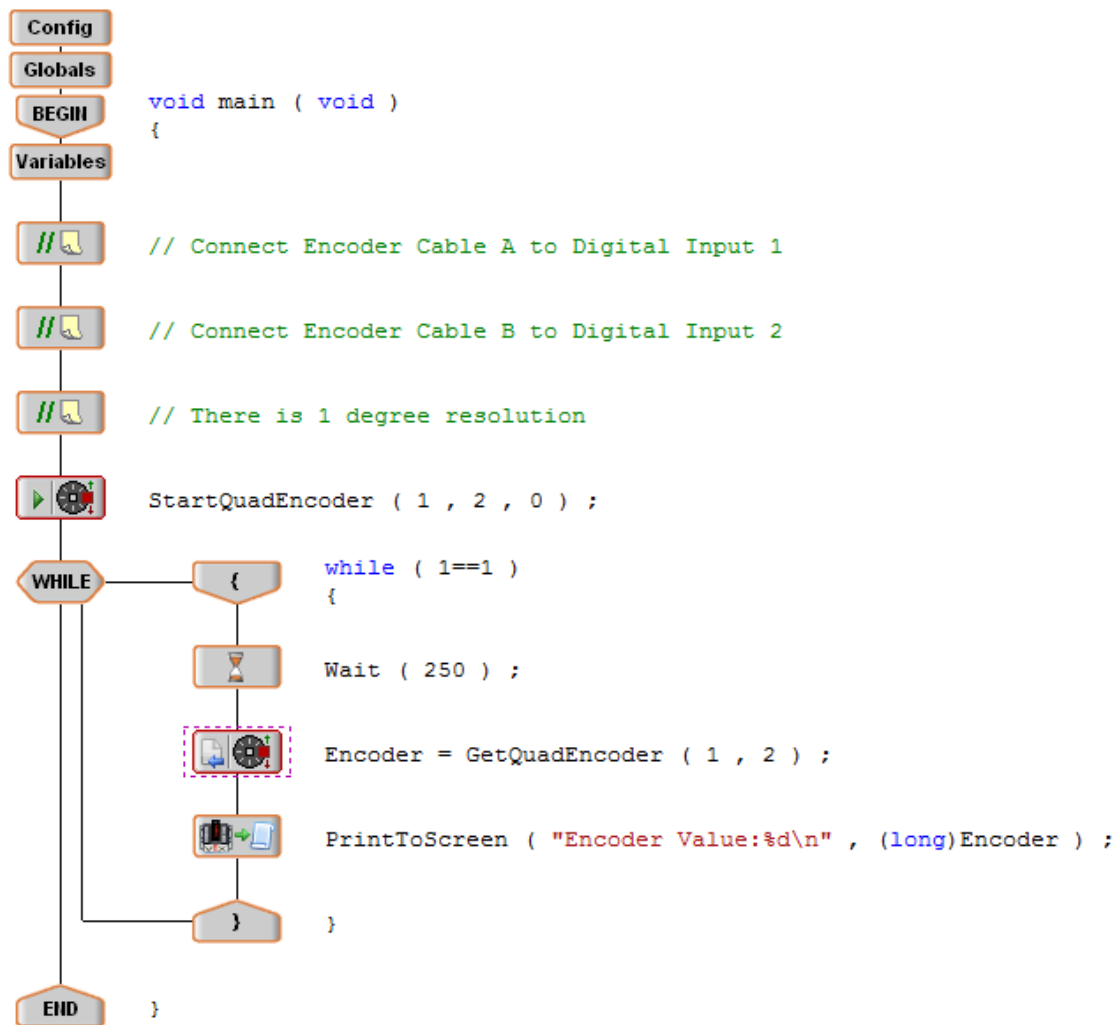
Code:
encoder = GetEncoder (2) ;

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Tester ceci

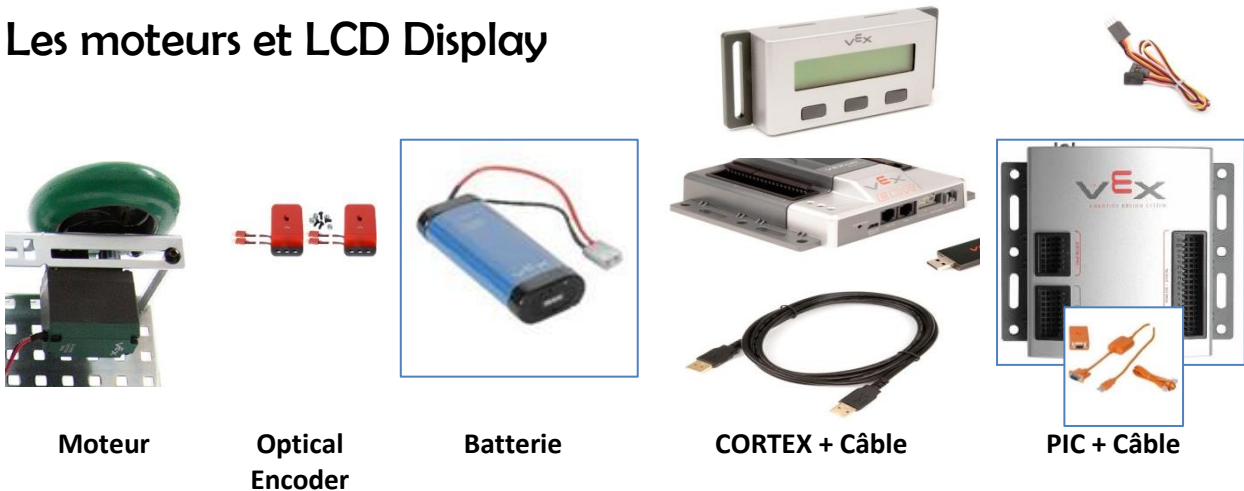


D. Tester votre programme et ajouter votre touche à ce code

2. Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.

Pratique #10 : Affiche la vitesse et le sens de rotation

1. Les moteurs et LCD Display



Avant de commencer, vous devez :

- A. Vous s'assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- C. Connecter le LCD Display (RX et TX) et l'Encoder aux **ports 3 (INTERRUPT), 11 (ANALOGUE/DIGITAL)** du PIC/Cortex indiqué dans le code ci-dessous.
- D. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB

Après les connexions des différents composants et les vérifications, vous devez :

- A. Démarrer de votre ordinateur le programme « EasyC V5 ».
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »

B. Cliquer sur « Macros and constants » et dans la section « Global Variables » déclarer une variable que se nomme « Encoder » du type « unsigned long ».

C. Programmer les moteurs, l'Encoder et le LCD Display en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.

1. Afficher un chiffre à l'écran:

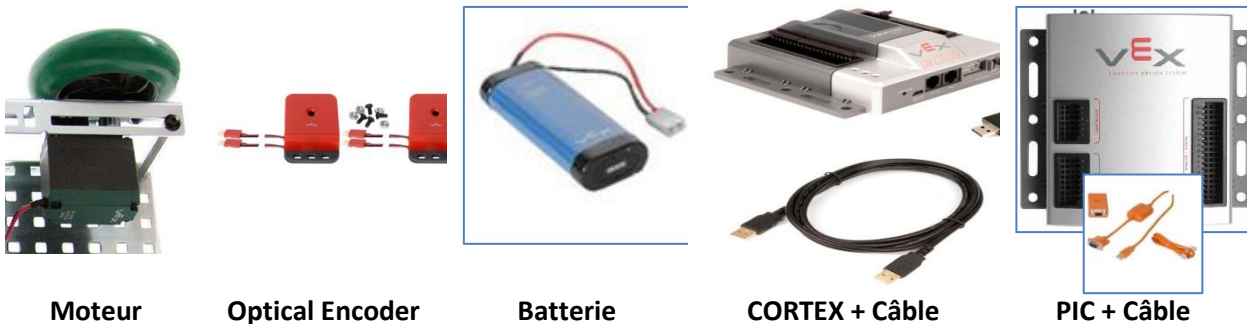
2. Afficher la vitesse du moteur et le sens de rotation du moteur :

D. Tester votre programme

2. Obtenir l'approbation de l'enseignant avant de passer à l'activité suivante.

Pratique #8 : Accéléromètre

3. Le moteur et Optical Encoder

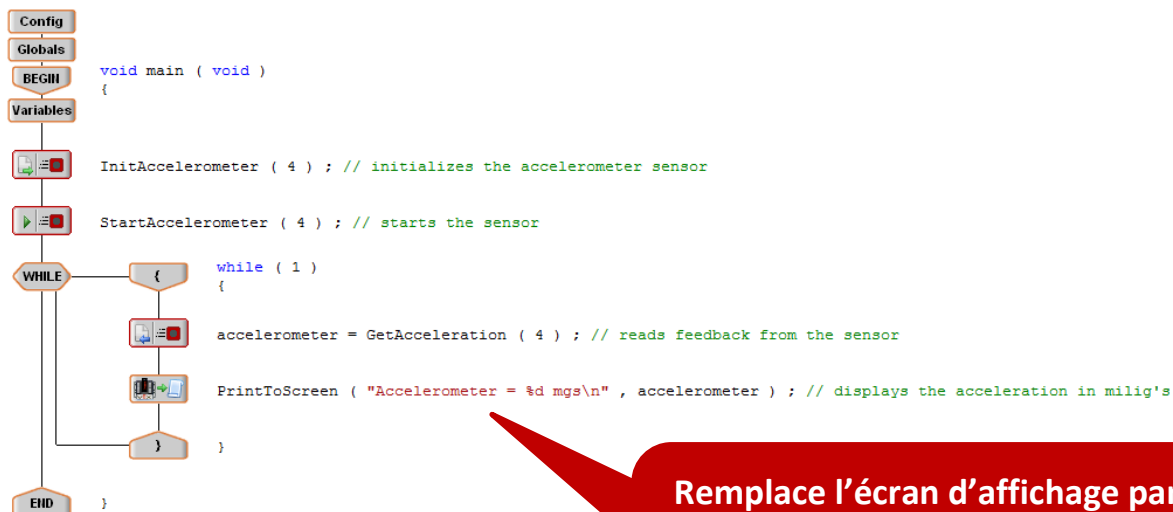


Avant de commencer, vous devez :

- Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- Connecter les moteurs aux ports du PIC/Cortex comme indiqué dans la pratique #1.
- Connecter l'Optical Encoder au **port 4 (DIGITAL)** du PIC/Cortex indiqué dans le code ci-dessous.
- Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB

Après les connexions des différents composants et les vérifications, vous devez :

- Démarrer de votre ordinateur le programme « EasyC V5 »
 - Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »
- Programmer les moteurs, l'Optical Encoder et LCD display en déplaçant et plaçant les blocs de codes utilisés dans le



Remplace l'écran d'affichage par les moteurs dans une structure de condition if

Print To Screen

Message:

Variable: Directive: Type-cast:

☒ Newline Character

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Local Variables

#	Type	Name	Value	Comment
1	int	accelerometer		
»				

OK Delete Cancel Help

Accelerometer

Select command:

☐ Initialize

☐ Start

☒ Get

☐ Stop

Analog Input #:

4

(Value Range: 1..8)

// accelerometer

Retrieve to:

accelerometer

(Returns 'int')

Code:

accelerometer = GetAcceleration (4) ;

Comment:

reads feedback from the sensor

F6 – Globals and Constants

Ctrl + F6 – Local Variables

OK

Cancel

Help

Accelerometer

Select command:

☒ Initialize

☐ Start

☐ Get

☐ Stop

Analog Input #:

4

(Value Range: 1..8)

// accelerometer

Code:

InitAccelerometer (4) ;

Comment:

initializes the accelerometer sensor

F6 – Globals and Constants

Ctrl + F6 – Local Variables

OK

Cancel

Help

Accelerometer

Select command:

☐ Initialize

☒ Start

☐ Get

☐ Stop

Analog Input #:

4

(Value Range: 1..8)

// accelerometer

Code:

StartAccelerometer (4) ;

Comment:

starts the sensor

F6 – Globals and Constants

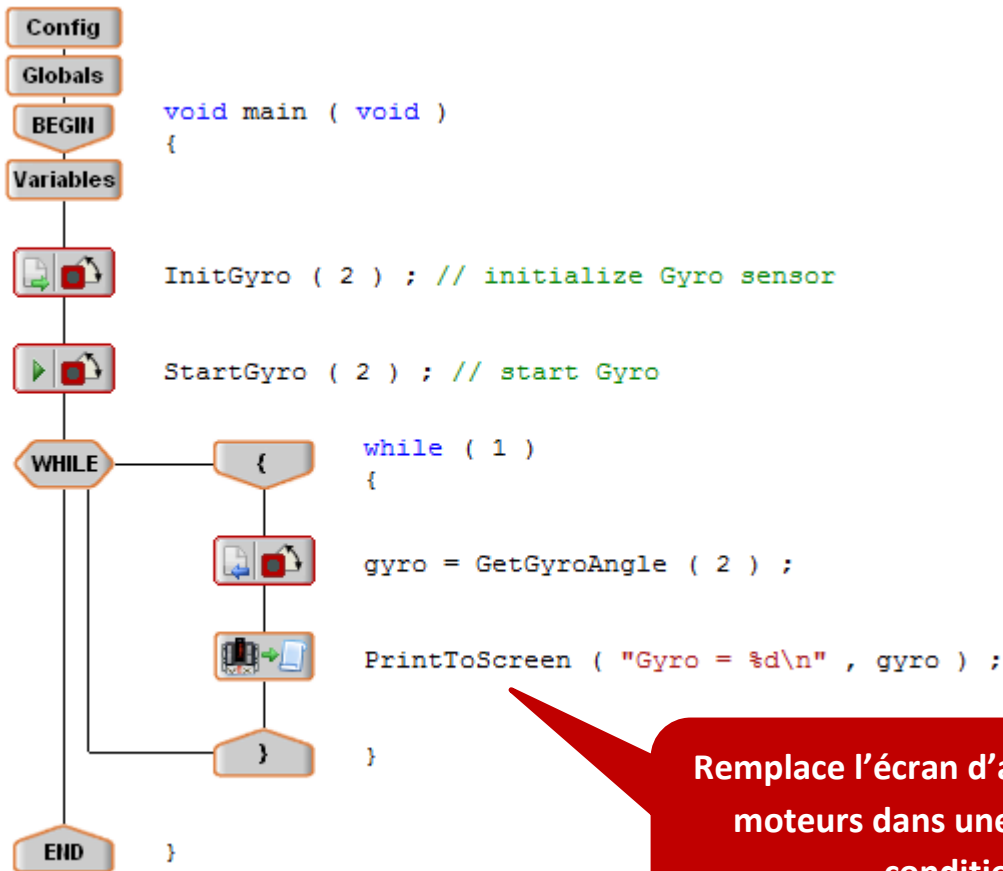
Ctrl + F6 – Local Variables

OK

Cancel

Help

PRATIQUE #9 : Gyro test



Remplace l'écran d'affichage par les moteurs dans une structure de condition if

```
1 #include "Main.h"
2
3 void main ( void )
4 {
5     int gyro;
6
7     InitGyro ( 2 ) ; // initialize Gyro sensor
8     StartGyro ( 2 ) ; // start Gyro
9     while ( 1 )
10    {
11        gyro = GetGyroAngle ( 2 ) ;
12        Print ToScreen ( "Gyro = %d\n" , gyro ) ;
13    }
14 }
```

Local Variables

#	Type	Name	Value	Comment
1	int	gyro		
»				

OK Delete Cancel Help

Gyro Sensor

Select command:

☒ Initialize
 ☐ Set Type
 ☐ Start
 ☐ Get
 ☐ Stop

Analog Input #: 2 (Value Range: 1..8)

Code:

InitGyro (2);

Comment:

initialize Gyro sensor

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Gyro Sensor

Select command:

☐ Initialize
 ☐ Set Type
 ☒ Start
 ☐ Set Deadband
 ☐ Get
 ☐ Stop

2 (Value Range: 1..8)

Code:

StartGyro (2);

Comment:

start Gyro

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Gyro Sensor [X]

Select command:

☐ Initialize ☐ Set Type
☐ Start ☐ Set Deadband
☒ Get
☐ Stop

Analog Input #: (Value Range: 1..8)

Retrieve to: (Returns 'int')

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables


```

1 #include "Main.h"
2
3 void main ( void )
4 {
5     int gyro;
6
7     InitGyro ( 2 ) ; // initialize Gyro sensor
8     StartGyro ( 2 ) ; // start Gyro
9     while ( 1 )
10    {
11        gyro = GetGyroAngle ( 2 ) ;
12        PrintToScreen ( "Gyro = %d\n" , gyro ) ;
13    }
14 }
  
```

Print To Screen [X]

Message:

Variable: Directive: Type-cast:

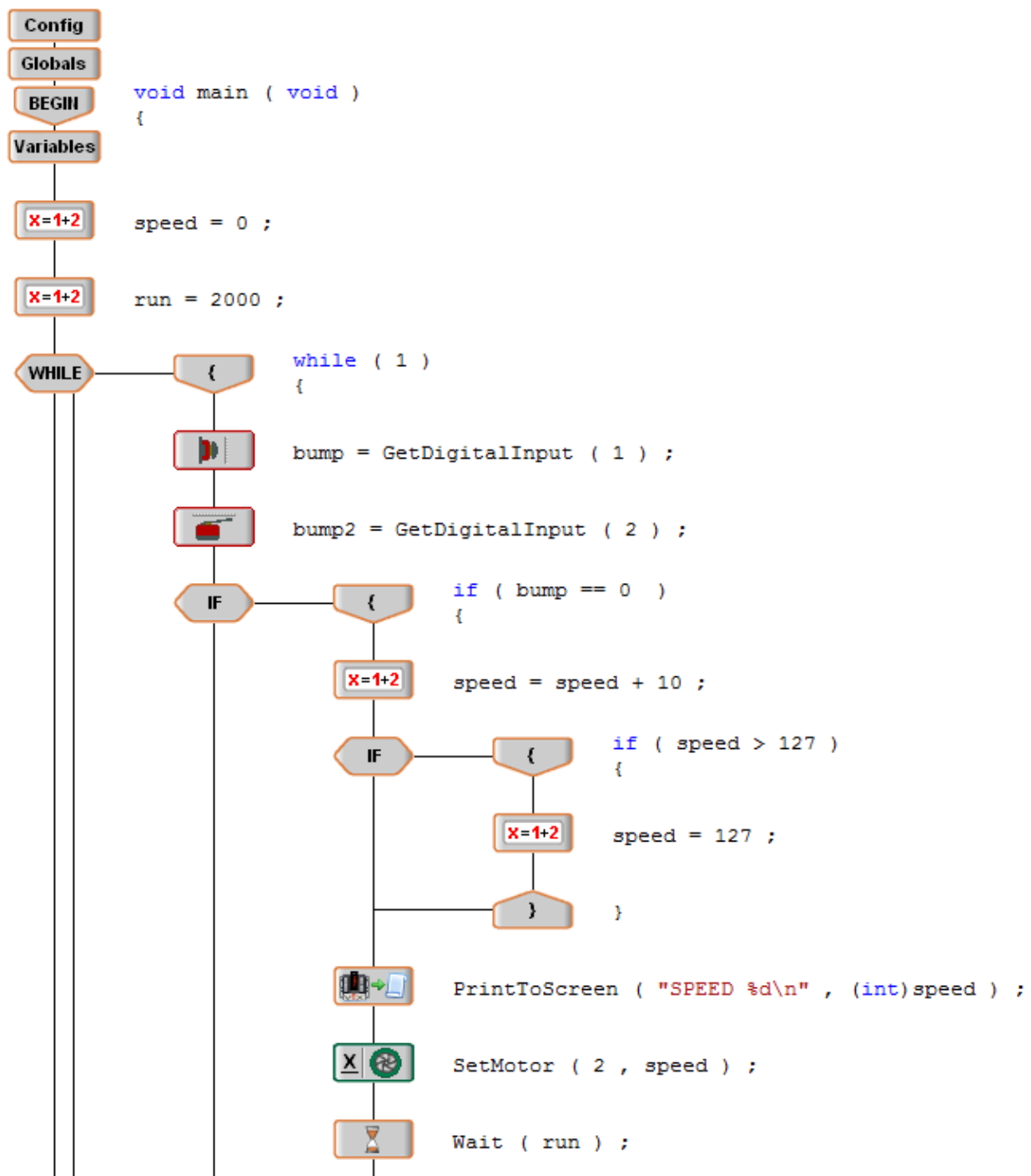
☒ Newline Character

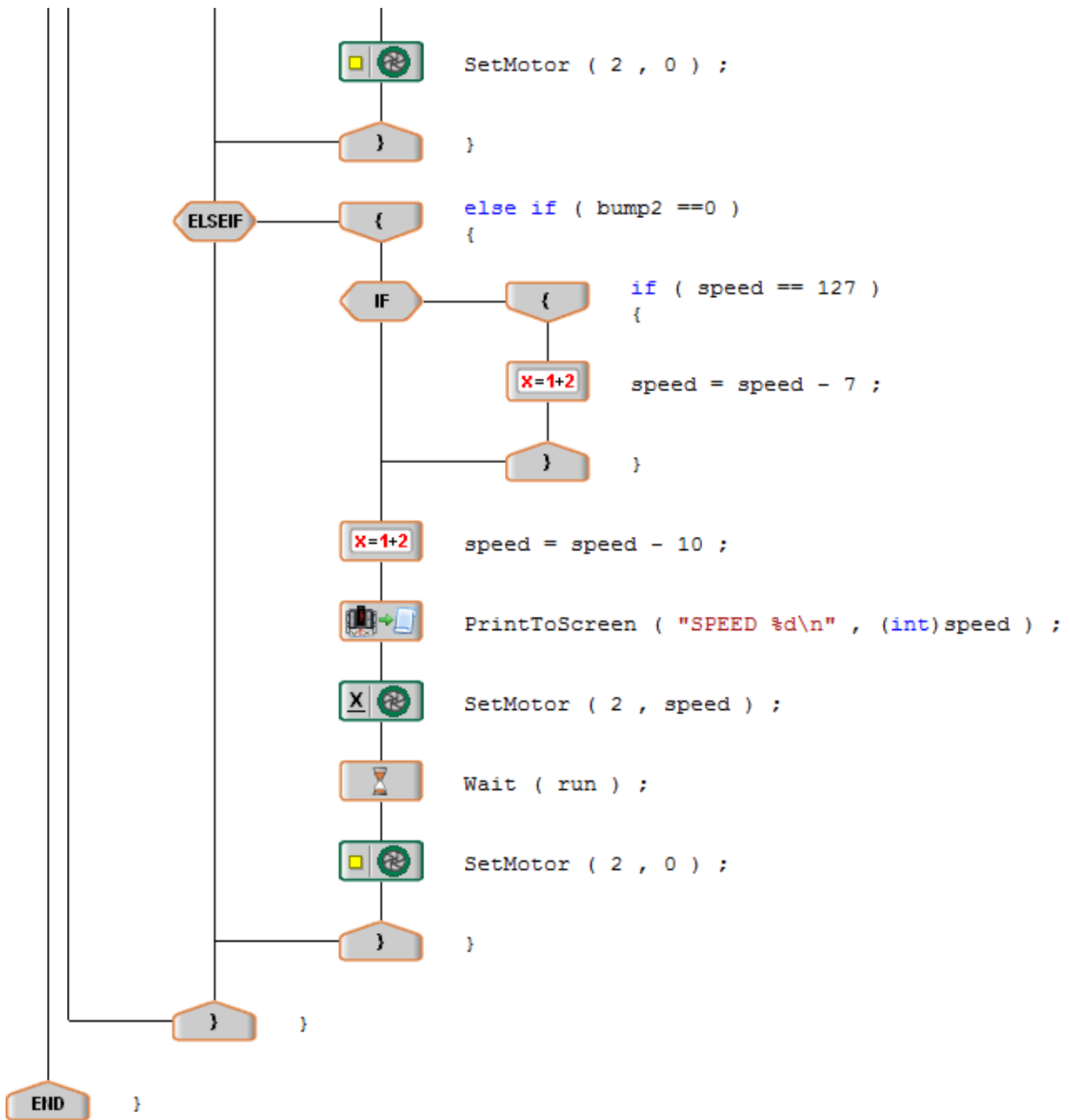
Code:

Comment:

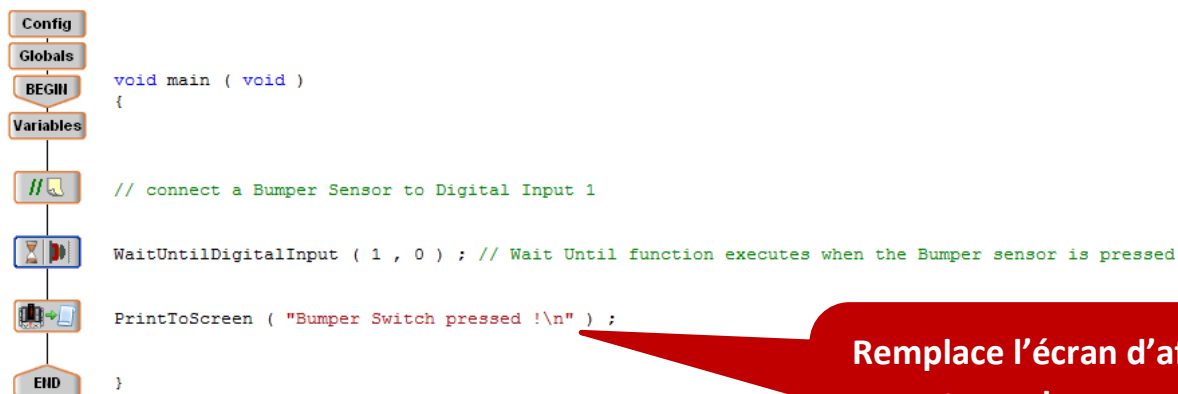
F6 – Globals and Constants Ctrl + F6 – Local Variables

PRATIQUE #10 : Torque test

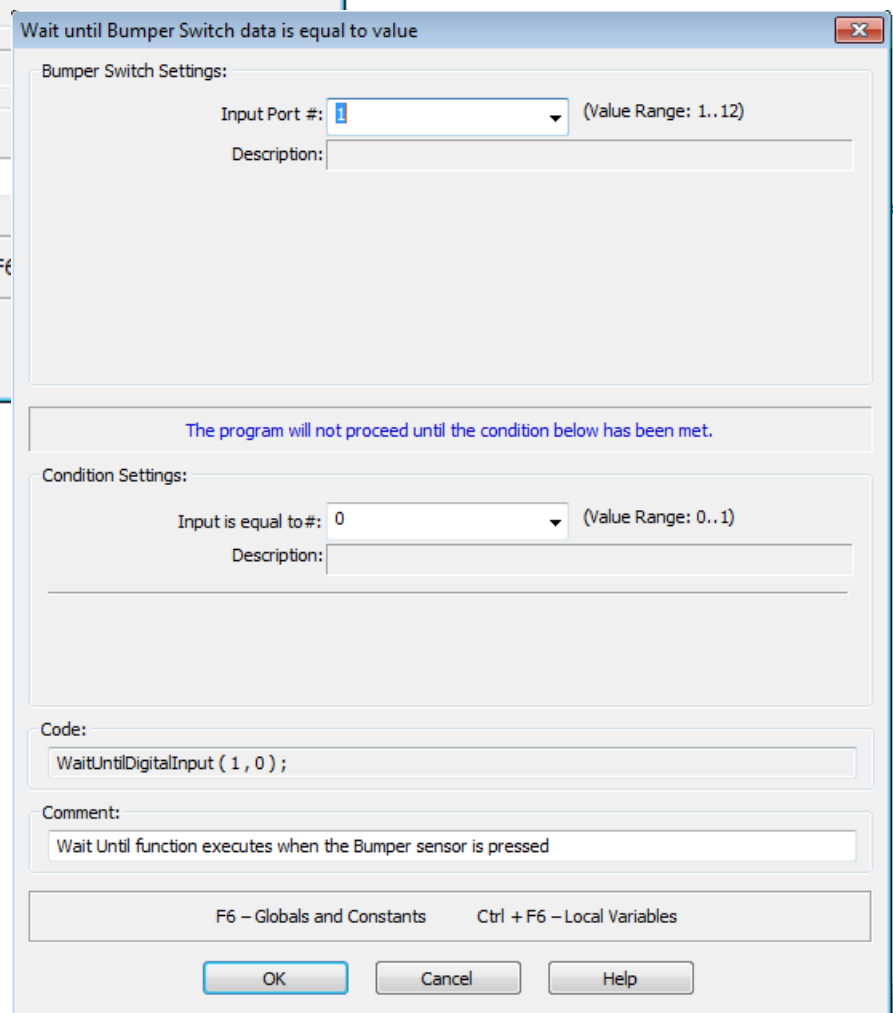
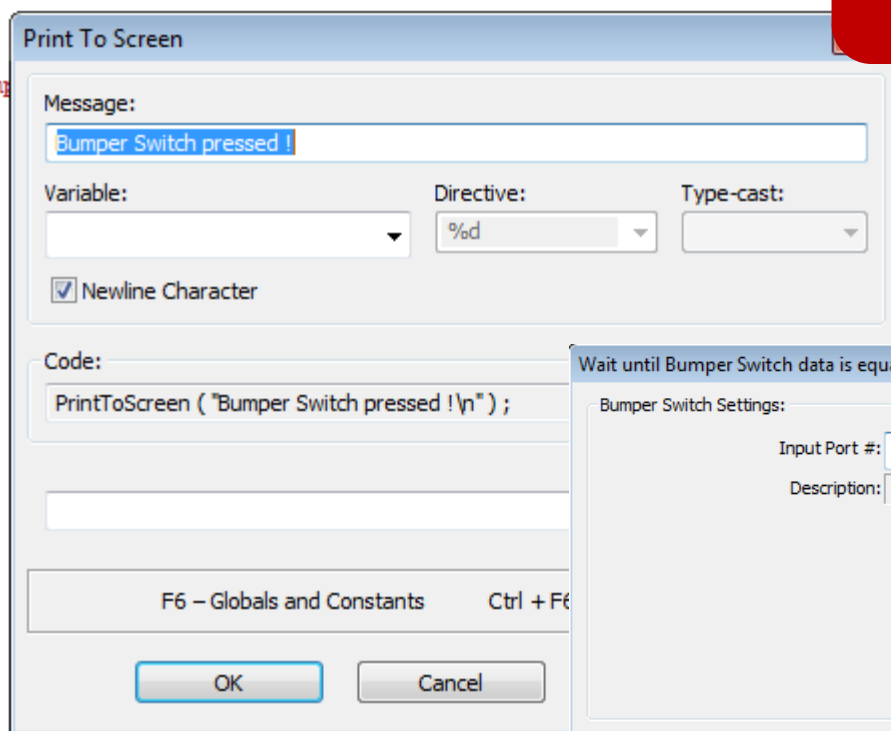




PRATIQUE #11 : wait bumper



Remplace l'écran d'affichage par les moteurs dans une structure de condition if



PRATIQUE #12 : Télécommande



Batterie



CORTEX + Câble



Télécommande



USB



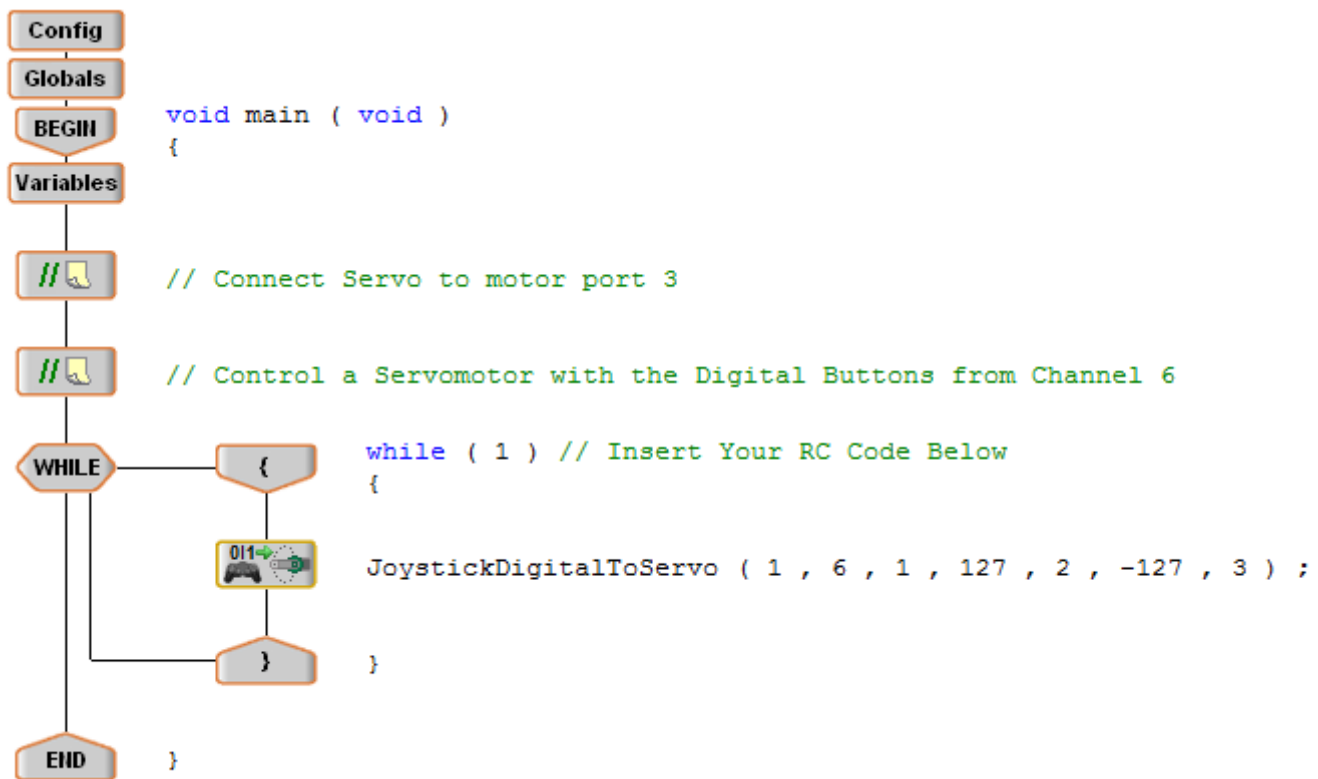
Moteur

Avant de commencer, vous devez :

- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter le moteur au **port 3** du PIC/Cortex indiqué dans le code ci-dessous
- C. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB pour télécharger le programme.

Après les connexions des différents composants et les vérifications, vous devez :

- E. Démarrer de votre ordinateur le programme « EasyC V5 ».
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** ».
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** ».
- F. Programmer le moteur et la chaîne de télécommande en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.
- G. Déconnecter le câble USB de l'ordinateur et connecter la clé USB pour que la télécommande puisse communiquer avec le Cortex. Attendre environ 10 secondes pour qu'ils se connectent.



Joystick Digital to Servo

Joystick #: 1 (Value Range: 1..2)

Channel #: 5 (Value Range: 5..8)

Forward Direction:

Button #: 1 (Down = 1, Up = 2)

Servo Value: 127 (Value Range: -127..127)

Reverse Direction:

Button #: 2 (Down = 1, Up = 2)

Servo Value: -127 (Value Range: -127..127)

Servo #: 3 (Value Range: 1..12)

Code:

```
JoystickDigitalToServo ( 1 , 5 , 1 , 127 , 2 , -127 , 3 ) ;
```

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

PRATIQUE 13 : potentiomètre



Batterie



CORTEX + Câble



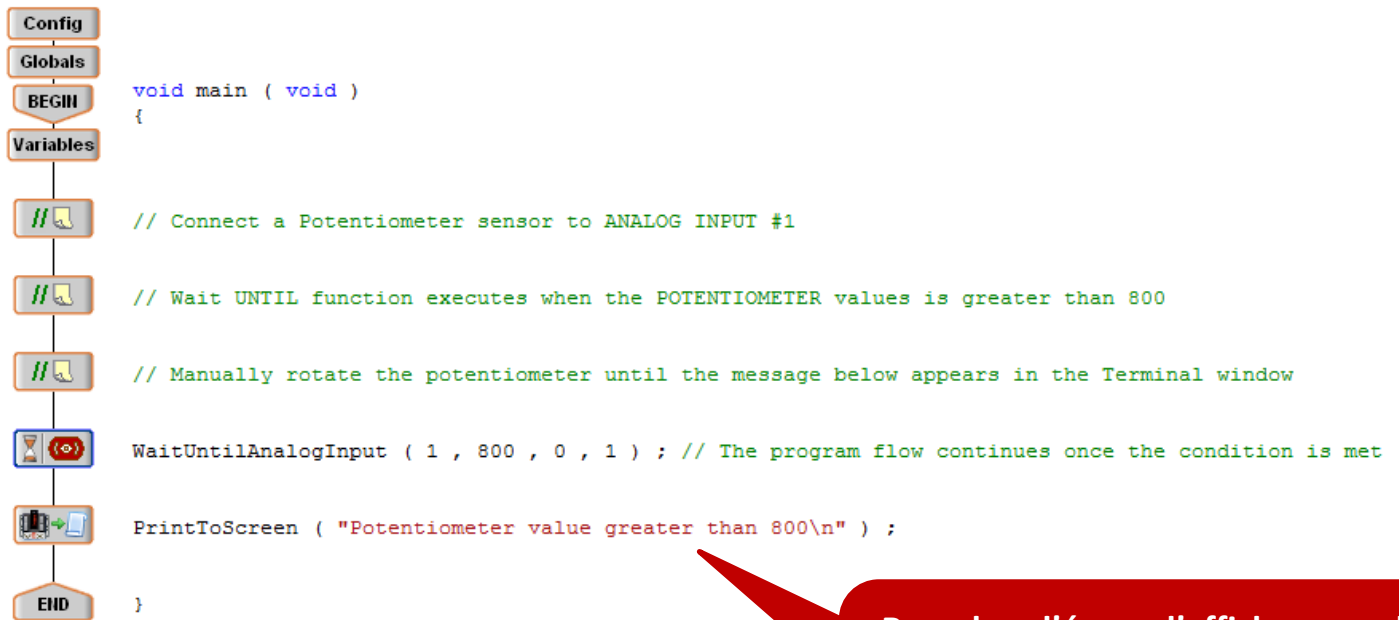
Potentiomètre

Avant de commencer, vous devez :

- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter le potentiomètre au **port 1 (ANALOG)** du PIC/Cortex indiqué dans le code ci-dessous.
- C. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB pour télécharger le programme.

Après les connexions des différents composants et les vérifications, vous devez :

- D. Démarrer de votre ordinateur le programme « EasyC V5 ».
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** ».
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »
- E. Programmer le potentiomètre et la chaîne de télécommande en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Remplace l'écran d'affichage par les moteurs dans une structure de condition if

Wait until Potentiometer data is greater or less than value

Potentiometer Settings:

Input Port #: (Value Range: 1..8)
Description:

The program will not proceed until the condition below has been met.

Condition Settings:

☒ Input is greater than: (Value Range: 0..1023)
Description:

☐ Input is less than:
Description:

Code:
WaitUntilAnalogInput (1 , 800 , 0 , 1) ;

Comment:
The program flow continues once the condition is met

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel

Print To Screen

Message:

Variable: Directive: Type-cast:

☒ Newline Character

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

PRATIQUE #14 : « Wait » Capteur ultrasonique



CORTEX + Câble



Télécommande



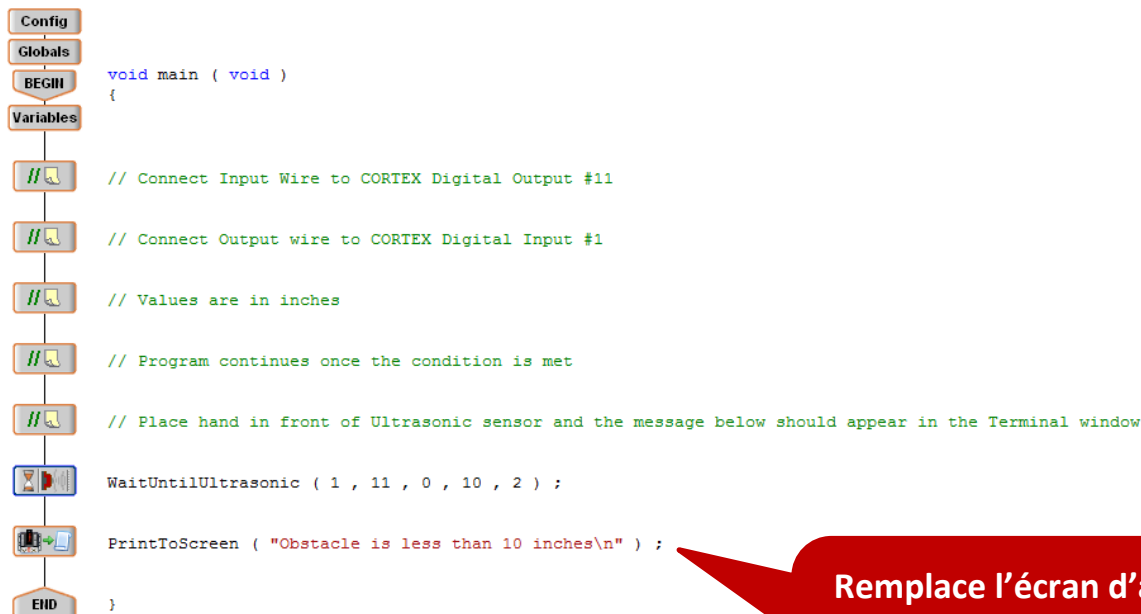
Capteur
ultrasonique

Avant de commencer, vous devez :

- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter le capteur ultrasonique (INPUT) au **port 11** (ANALOG) et (OUTPUT) au **port 1** (ANALOG) du PIC/Cortex indiqué dans le code ci-dessous.
- C. Connecter le PIC ou le Cortex à l'ordinateur à l'aide du câble USB pour télécharger le programme.

Après les connexions des différents composants et les vérifications, vous devez :

- D. Démarrer de votre ordinateur le programme « EasyC V5 ».
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** »
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »
- E. Programmer le capteur ultrasonique en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Remplace l'écran d'affichage par les moteurs dans une structure de condition if

Wait until Ultrasonic Sensor data is greater or less than value

Ultrasonic Sensor Settings:

Interrupt Port #: (Value Range: 1..12)
Description:

Output Port #: (Value Range: 1..12)
Description:

The program will not proceed until the condition below has been met.

Condition Settings:

☐ Value is greater than: (Value Range: 1..12)
Description:

☒ Value is less than: (Value Range: 1..12)
Description:

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Print To Screen

Message:

Variable: Directive: Type-cast:

☒ Newline Character

Code:

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

PRATIQUE #14: Télécommande à l'accéléromètre



Batterie



CORTEX + Câble



Télécommande



USB



Moteur

Avant de commencer, vous devez :

- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter le moteur 2 au **port 2** (MOTOR) et le moteur 3 au **port 3** (MOTOR) du PIC/Cortex indiqué dans le code ci-dessous.
- C. Connecter le PIC ou le Cortex à l'ordinateur avec le câble USB pour télécharger le programme. Puis connecter la clé USB compatible à celle de la télécommande.

Après les connexions des différents composants et les vérifications, vous devez :

- D. Démarrer de votre ordinateur le programme « EasyC V5 »
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** ».
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »
- E. Programmer le capteur ultrasonique en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Get Joystick Accelerometer

Joystick #: 1 (Value Range: 1..2)

Axis #: 1 (Axis X = 1, Axis Y = 2)

Retrieve to: axisX

Code:
axisX = GetJoystickAccelerometer(1 , 1) ;

Comment:
read Joystick accelerometer values from axis X

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

Get Joystick Accelerometer

Joystick #:

1

(Value Range: 1..2)

Axis #:

2

(Axis X = 1, Axis Y = 2)

Retrieve to:

axisY

Code:

axisY = GetJoystickAccelerometer(1 , 2) ;

Comment:

read Joystick accelerometer values from axis Y

F6 – Globals and Constants

Ctrl + F6 – Local Variables

OK

Cancel

Help

Local Variables

#	Type	Name	Value	Comment	
1	int	axisX			
2	int	axisY			
»					

OK

Delete

Cancel

Help

Motor Module [X]

Motor Number: (Value Range: 1..10)

Motor Direction (Value Range: -127..127):

Counter-clockwise ☐ 127

Stop ☐ 0

Clockwise ☐ -127

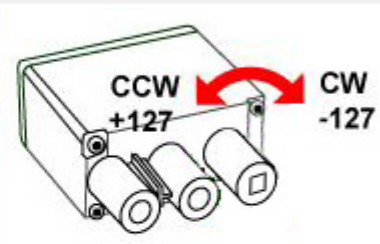
User Value ☒ axisX

Code:
SetMotor (2 , axisX) ;

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help



Motor Module [X]

Motor Number: (Value Range: 1..10)

Motor Direction (Value Range: -127..127):

Counter-clockwise ☐ 127

Stop ☐ 0

Clockwise ☐ -127

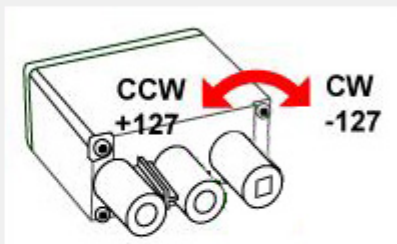
User Value ☒ axisY

Code:
SetMotor (3 , axisY) ;

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help



PRATIQUE #15 : Télécommande DIGITAL et moteur



Batterie



CORTEX + Câble



Télécommande



USB



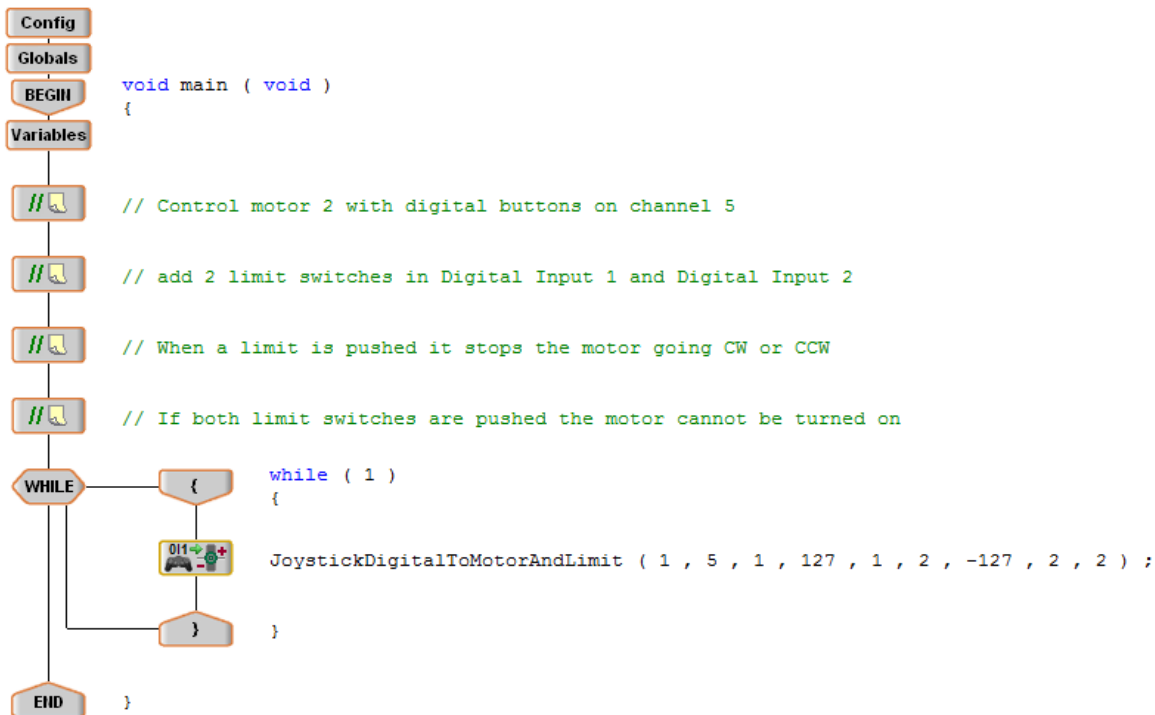
Moteur

Avant de commencer, vous devez :

- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter le moteur au **port 2 (MOTOR)** du PIC/Cortex indiqué dans le code ci-dessous.
- C. Connecter le PIC ou le Cortex à l'ordinateur avec le câble USB pour télécharger le programme. Puis connecter la clé USB compatible à celle de la télécommande.

Après les connexions des différents composants et les vérifications, vous devez :

- D. Démarrer de votre ordinateur le programme « EasyC V5 ».
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** ».
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »
- E. Programmer le capteur ultrasonique en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Joystick Digital to Motor & Limit

Joystick #: 1 (Value Range: 1..2)

Channel #: 5 (Value Range: 5..8)

Forward Direction:

Button #: 1 (Down = 1, Up = 2)

Motor Value: 127 (Value Range: -127..127)

Limit Switch: 1 (Value Range: 0 - Disabled, 1..12 - Input Number)

Reverse Direction:

Button #: 2 (Down = 1, Up = 2)

Motor Value: -127 (Value Range: -127..127)

Limit Switch: 2 (Value Range: 0 - Disabled, 1..12 - Input Number)

Motor #: 2 (Value Range: 1..12)

Code:

```
JoystickDigitalToMotorAndLimit ( 1 , 5 , 1 , 127 , 1 , 2 , -127 , 2 , 2 ) ;
```

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

PRATIQUE # X : Télécommande et moteur



Batterie



CORTEX + Câble



Télécommande



USB



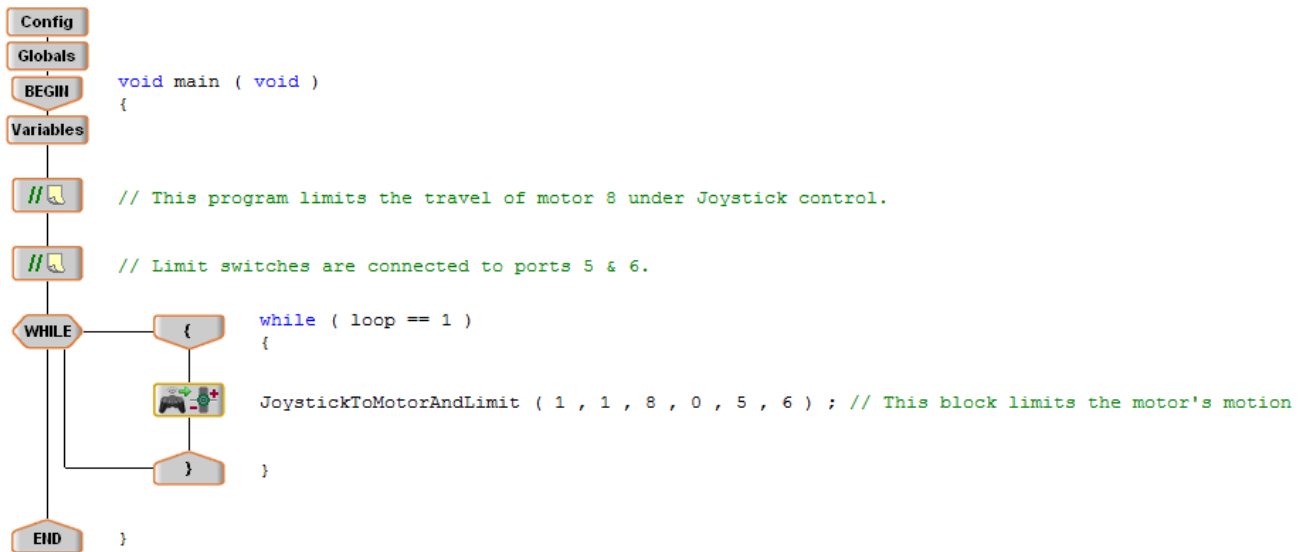
Moteur

Avant de commencer, vous devez :

- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter le moteur au **port 8 (MOTOR)** du PIC/Cortex indiqué dans le code ci-dessous.
- C. Connecter le PIC ou le Cortex à l'ordinateur avec le câble USB pour télécharger le programme. Puis connecter la clé USB compatible à celle de la télécommande.

Après les connexions des différents composants et les vérifications, vous devez :

- D. Démarrer de votre ordinateur le programme « EasyC V5 ».
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** ».
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »
- E. Programmer le capteur ultrasonique en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Joystick to Motor & Limit

Transmitter:
Joystick #: 1 (Value Range: 1..2)

Transmitter Channel:
Channel: 1 (Value Range: 1..4)

Motor:
8 (Value Range: 1..10)

Invert Direction:
0 (Value Range: 0 - Default, 1 - Invert)

Positive Limit Switch:
5 (Value Range: 0 - Disabled, 1..12 - Input Number)

Negative Limit Switch:
6 (Value Range: 0 - Disabled, 1..12 - Input Number)

Code:
JoystickToMotorAndLimit (1 , 1 , 8 , 0 , 5 , 6) ;

Comment:
This block limits the motor's motion

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

PRATIQUE #Y : Télécommande, moteur et capteur de limite



Batterie



CORTEX + Câble



Télécommande



USB



Limite



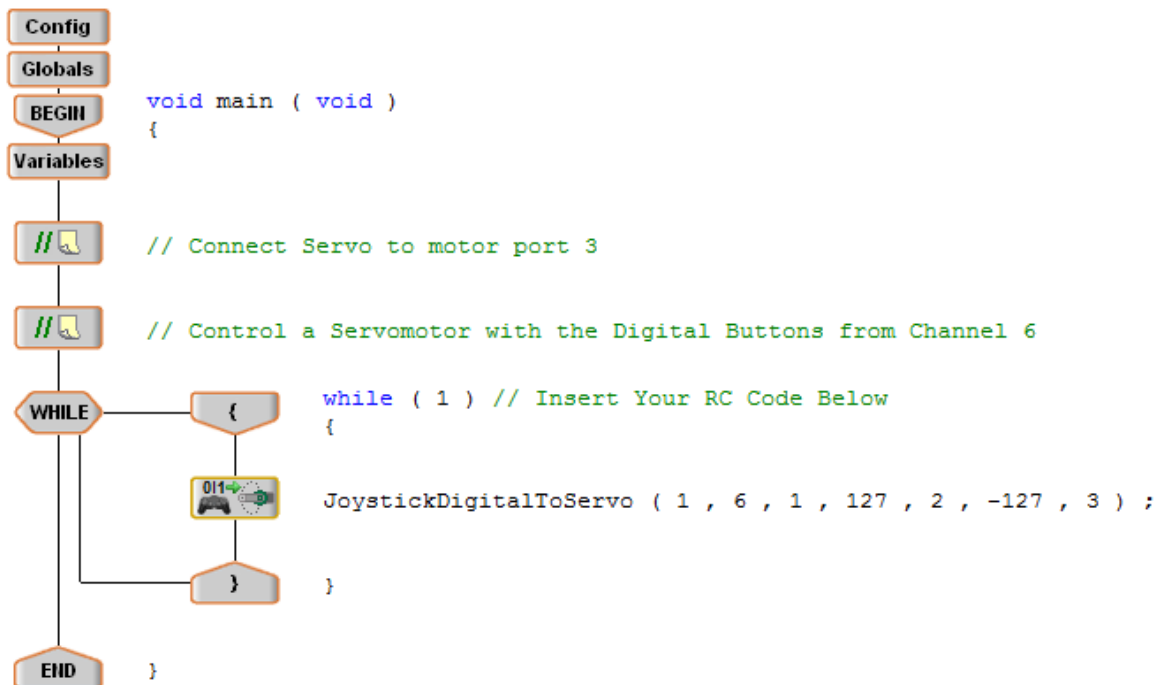
Moteur

Avant de commencer, vous devez :

- A. Vous assurer que la batterie est bien chargée et connectée au PIC ou au Cortex de la plateforme de test.
- B. Connecter le Servomoteur au **port 3 (MOTOR)** du PIC/Cortex indiqué dans le code ci-dessous.
- C. Connecter le PIC ou le Cortex à l'ordinateur avec le câble USB pour télécharger le programme. Puis connecter la clé USB compatible à celle de la télécommande.

Après les connexions des différents composants et les vérifications, vous devez :

- D. Démarrer de votre ordinateur le programme « EasyC V5 ».
 - a. Cliquer sur « **Fichier ou File** » puis sur « **Nouveau projet autonome ou New Standalone Project** ».
 - b. Choisissez l'option « **Autonome ou Autonomous Only Project** » et valider en cliquant sur le bouton « **OK** »
- E. Programmer le capteur ultrasonique en déplaçant et plaçant les blocs de codes utilisés dans le code ci-dessous dans votre nouveau programme.



Joystick Digital to Servo

Joystick #: 1 (Value Range: 1..2)

Channel #: 6 (Value Range: 5..8)

Forward Direction:

Button #: 1 (Down = 1, Up = 2)

Servo Value: 127 (Value Range: -127..127)

Reverse Direction:

Button #: 2 (Down = 1, Up = 2)

Servo Value: -127 (Value Range: -127..127)

Servo #: 3 (Value Range: 1..12)

Code:

```
JoystickDigitalToServo ( 1 , 6 , 1 , 127 , 2 , -127 , 3 );
```

Comment:

F6 – Globals and Constants Ctrl + F6 – Local Variables

OK Cancel Help

FONCTIONNEMENT AVEC TÉLÉCOMMANDE

Aller dans le dossier pratique guidée « 11_Programmation Télécommande » ouvrir la vidéo et programmer la télécommande en utilisant les éléments de la plateforme (Ex. Faire tourner le moteur à l'aide de la télécommande).

