



Banc de validation pour écran multifonction (EMF)

Sommaire

Objectif du projet

Présentation du banc d'essai

Etude de l'écran multifonction

Etude sur l'envoi de trames et reconnaissance sous LabVIEW

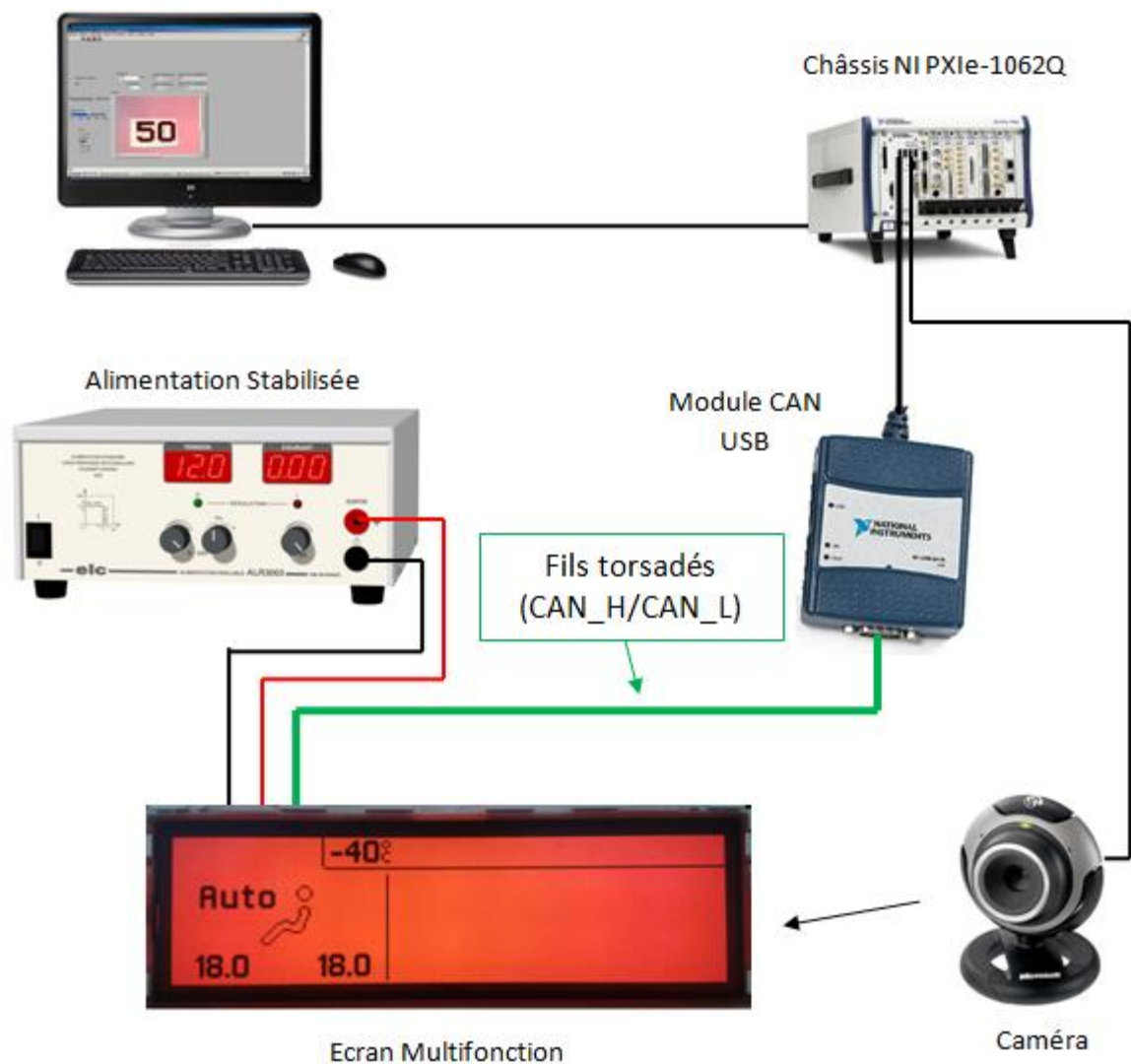
Mise en place de la validation avec TESTSTAND

Conclusion

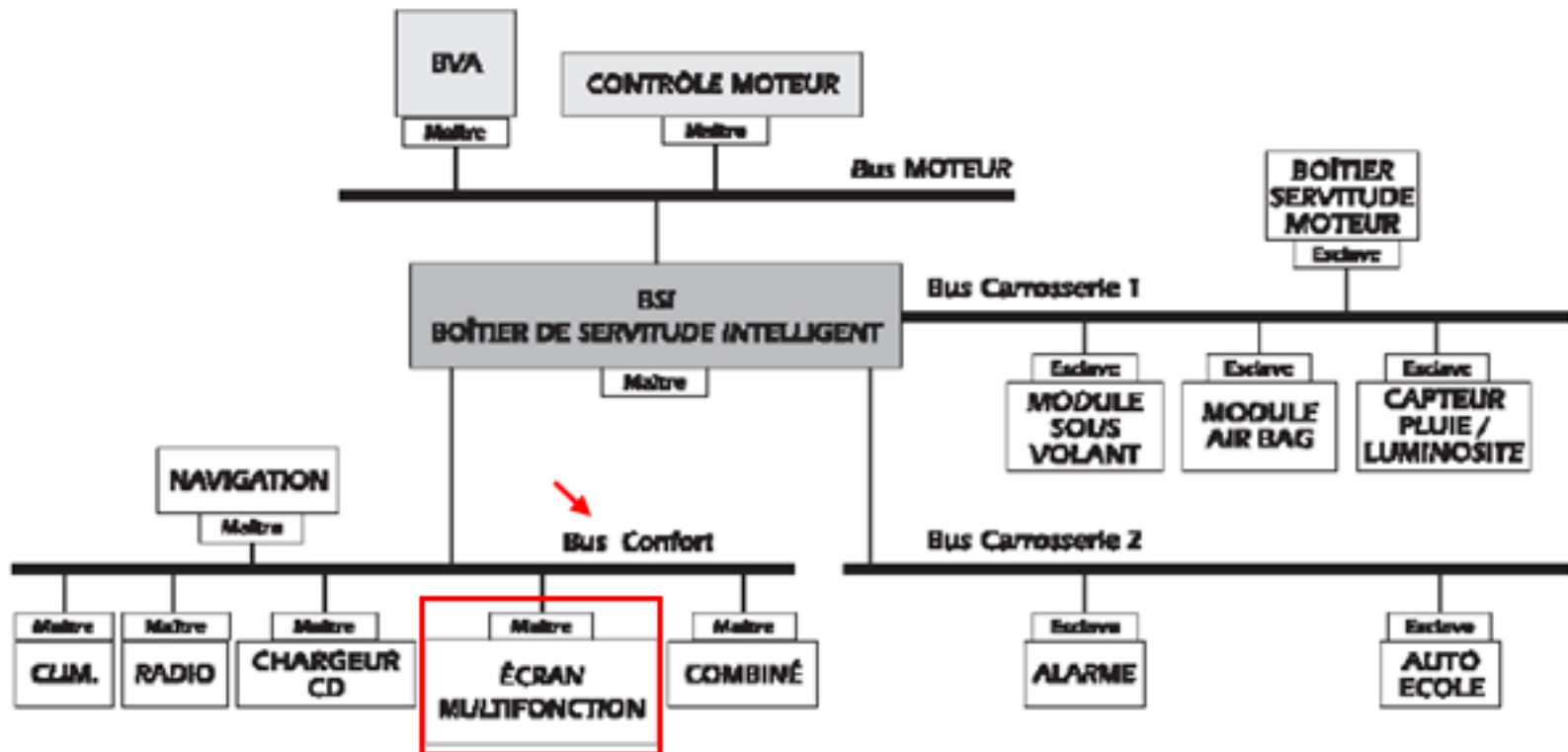
Objectif du projet

Créer un banc de test piloté sous LabVIEW et TestStand afin de valider le fonctionnement des afficheurs multiplexés

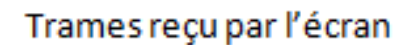
Présentation du banc d'essai



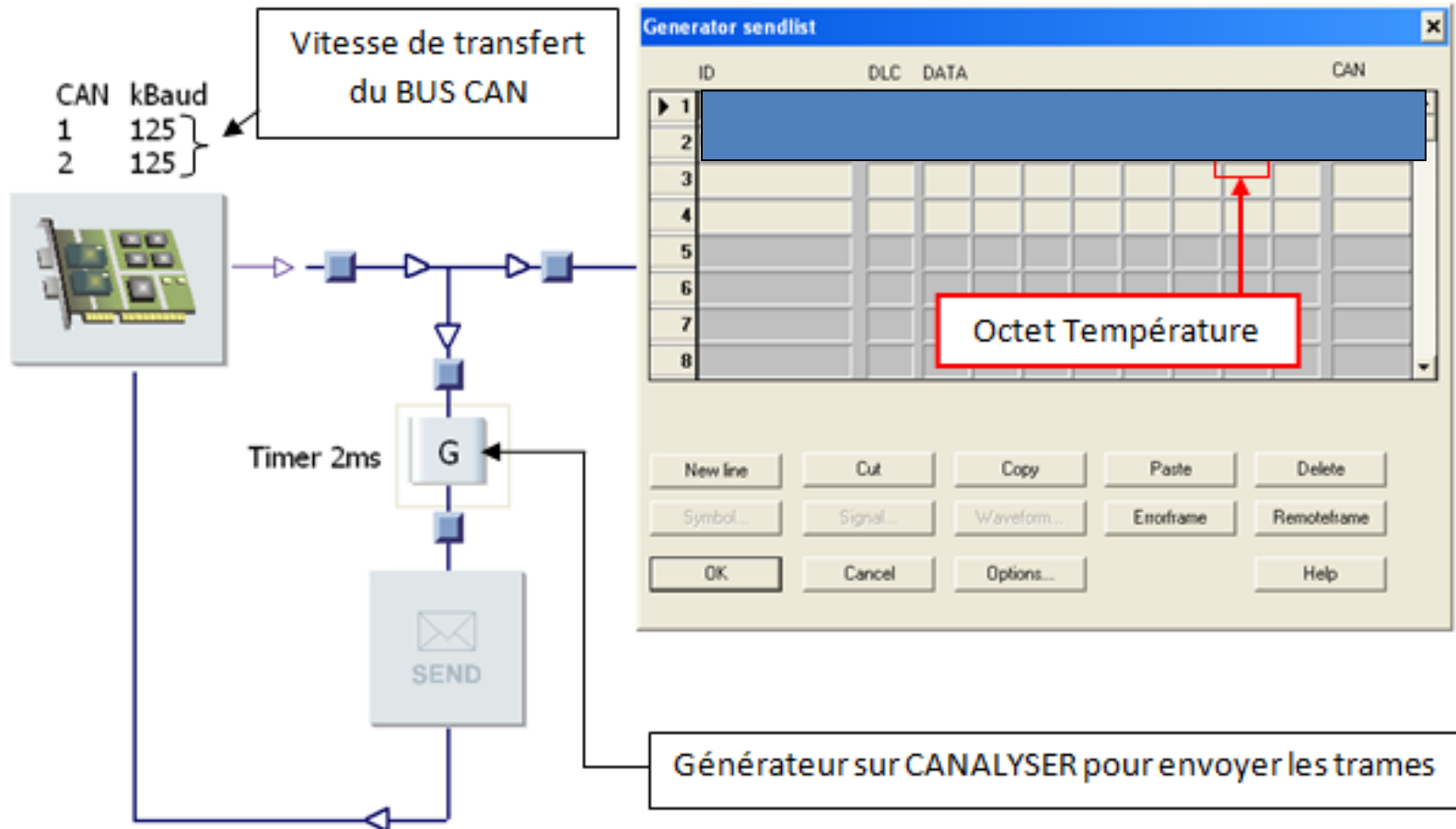
Etude de l'écran multifonction



Comme le schéma l'indique, pour lire les trames que reçoit et émet l'écran, le branchement s'effectue donc sur le BUS CONFORT (Low Speed 125kbts/s).

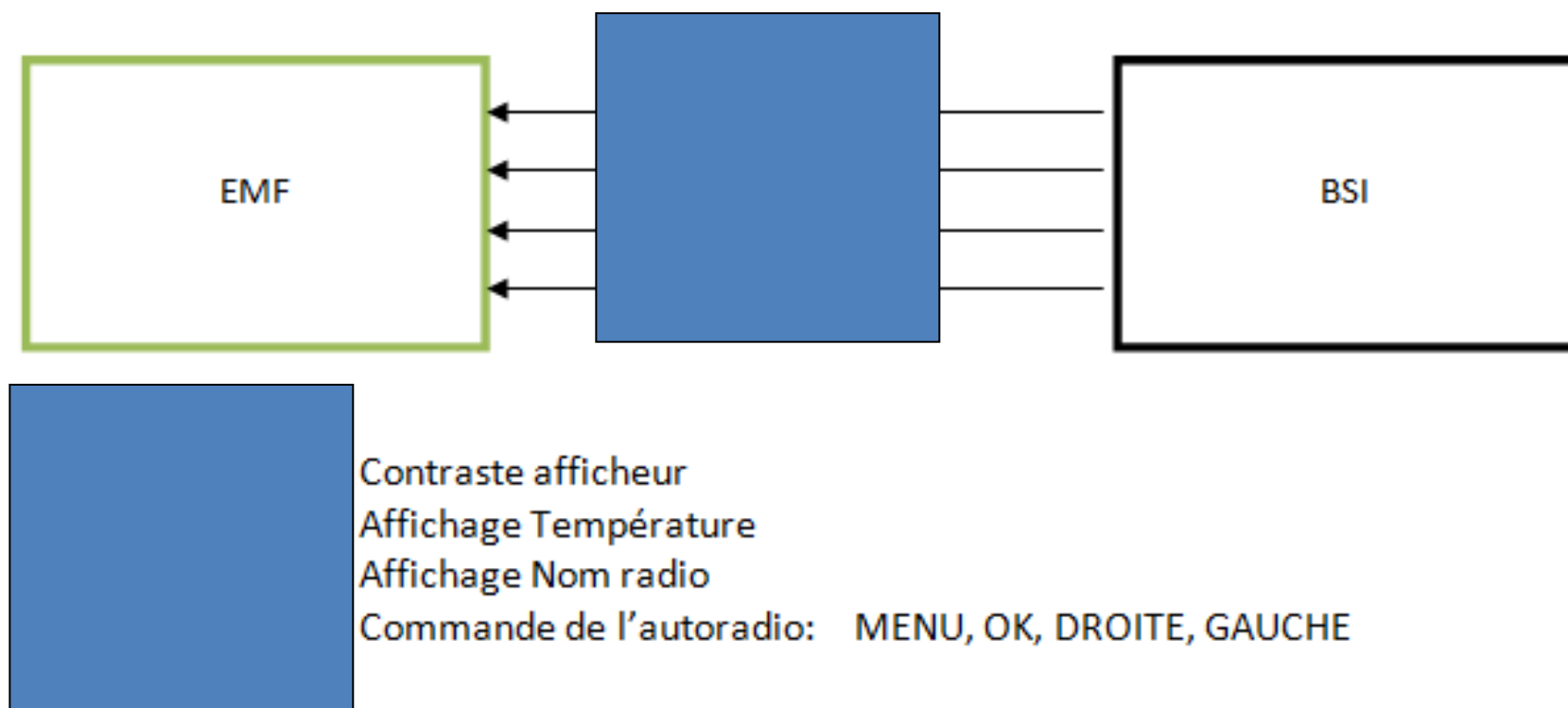


Etude de l'écran multifonction

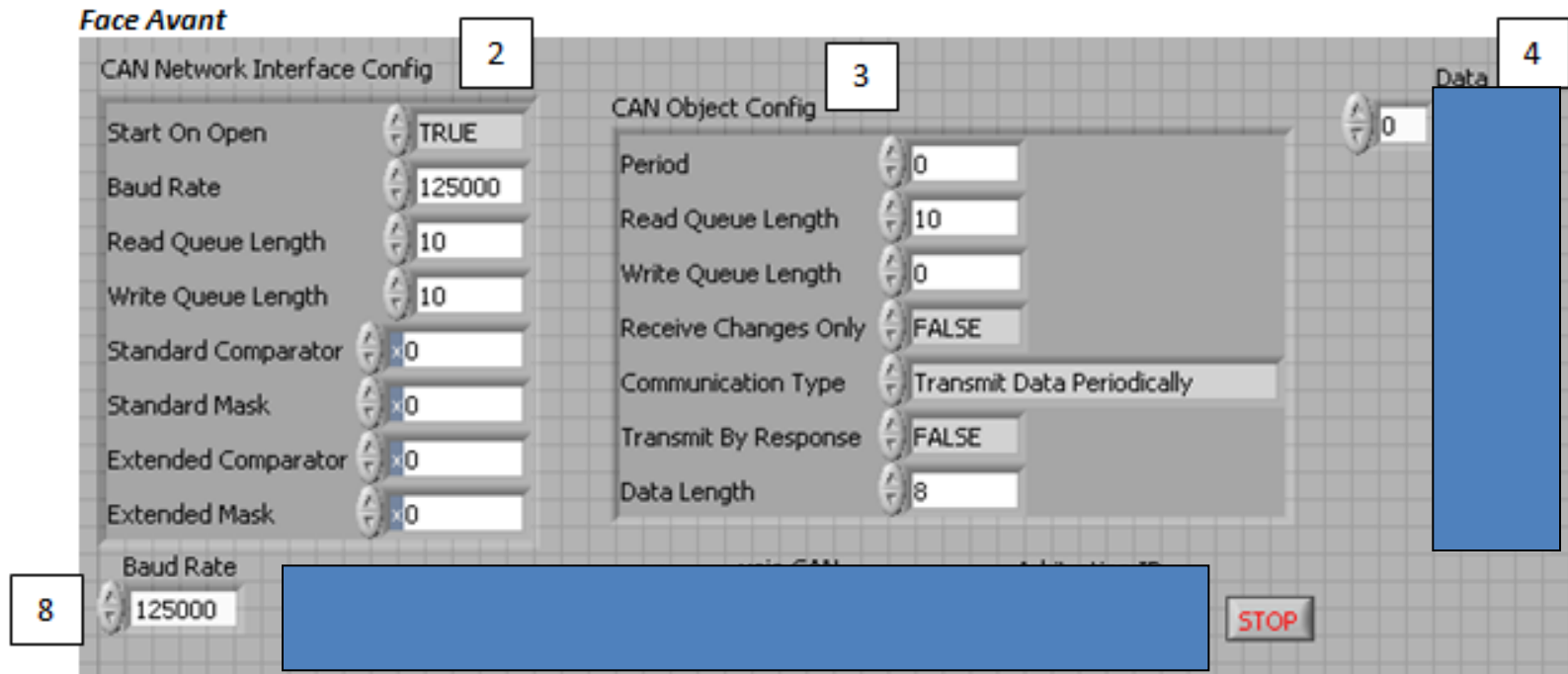


Etude de l'écran multifonction

Et en effectuant plusieurs essais nous avons pu identifier certaines fonctions des trames :



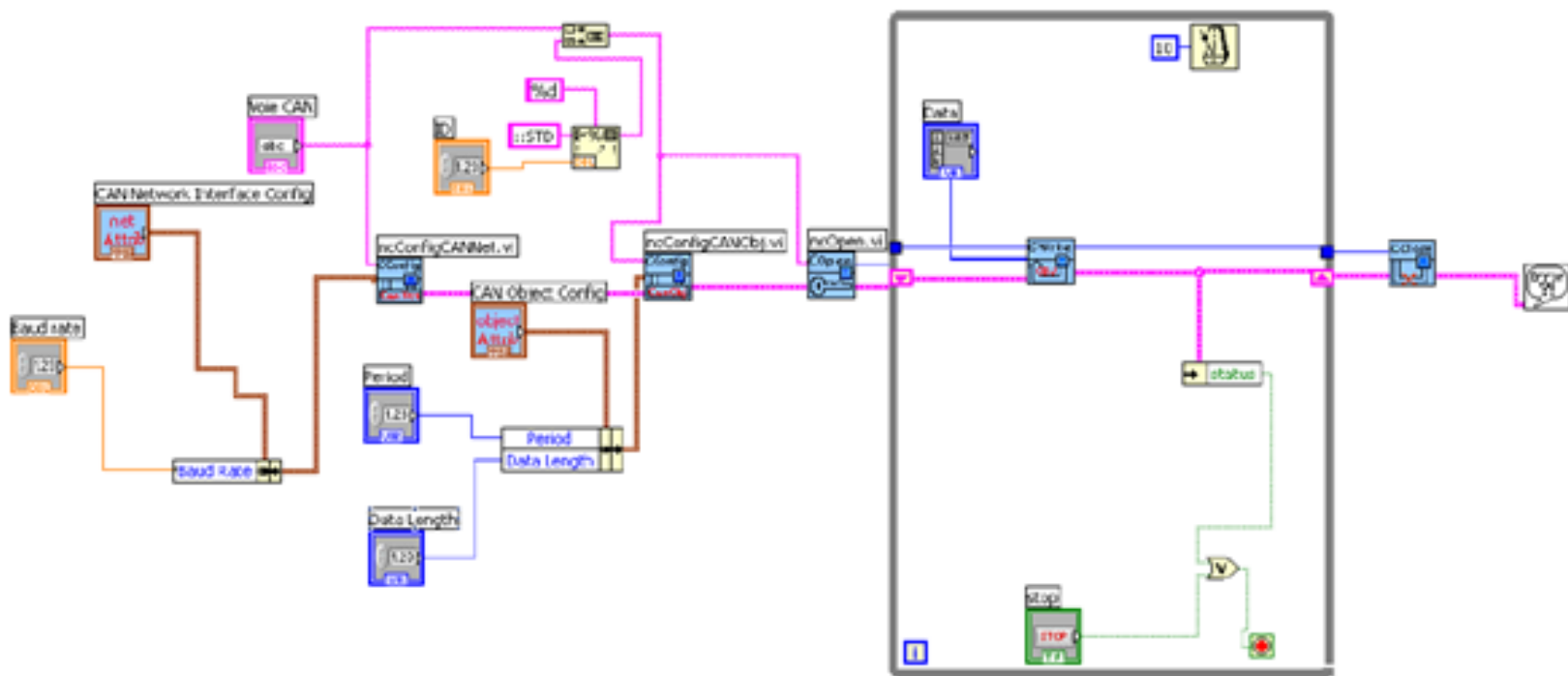
Etude sur l'envoi de trames sous LabVIEW



1. Sélection de la voie de la carte CAN
2. Configuration de l'interface
3. Configuration du mode d'envoi de la trame
4. Données de la trame
5. Période de la trame (en ms)
6. Identifiant de la trame
7. Longueur de la trame
8. Vitesse du Bus

Etude sur l'envoi de trames sous LabVIEW

Diagramme

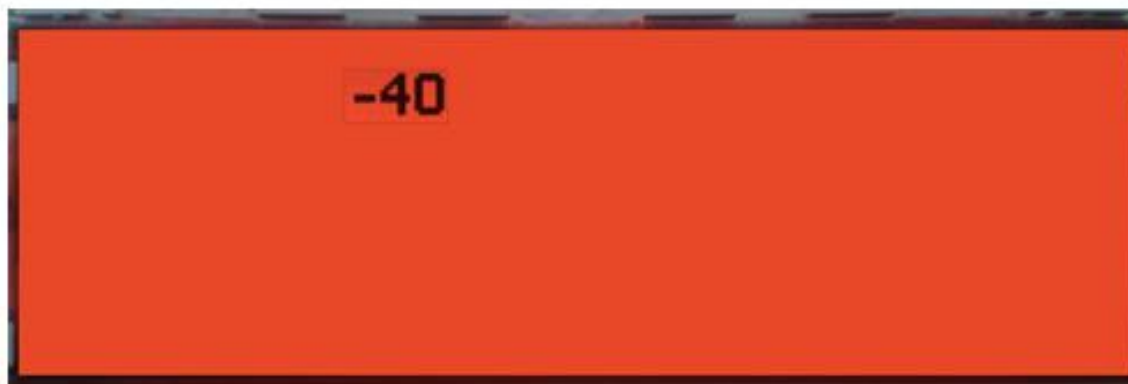


Reconnaissance sous LabVIEW

Vision



Les traits
perturbent la
lecture de la
température



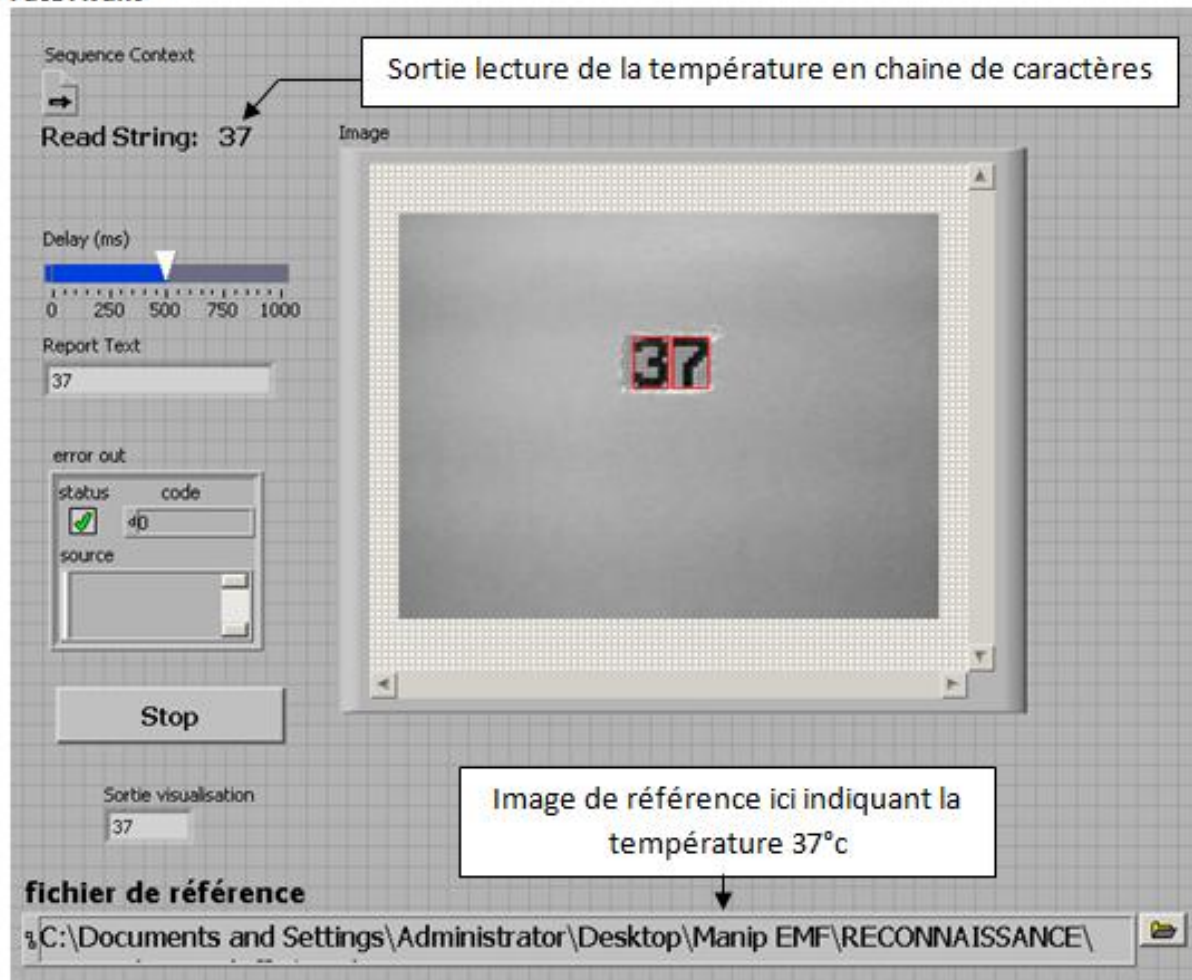
Ajout du Masque
Papier sur l'écran

La couleur du masque doit être identique au fond de l'écran car sinon la lecture de la température est aussi difficile !

Reconnaissance sous LabVIEW

Vision

Face Avant

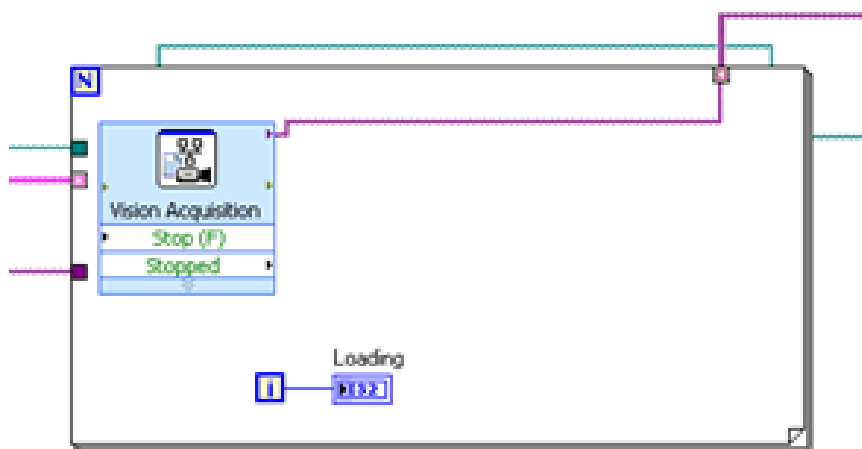


Reconnaissance sous LabVIEW

Vision

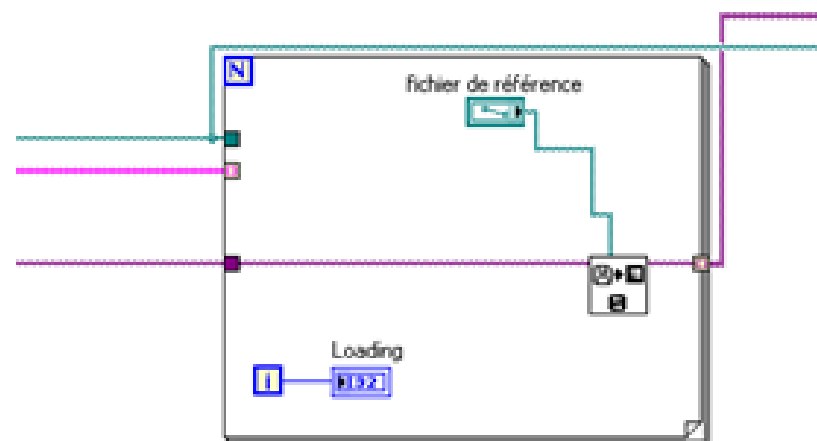
Avec Caméra

Load all the demo images into memory.



Avec une image

Load all the demo images into memory.



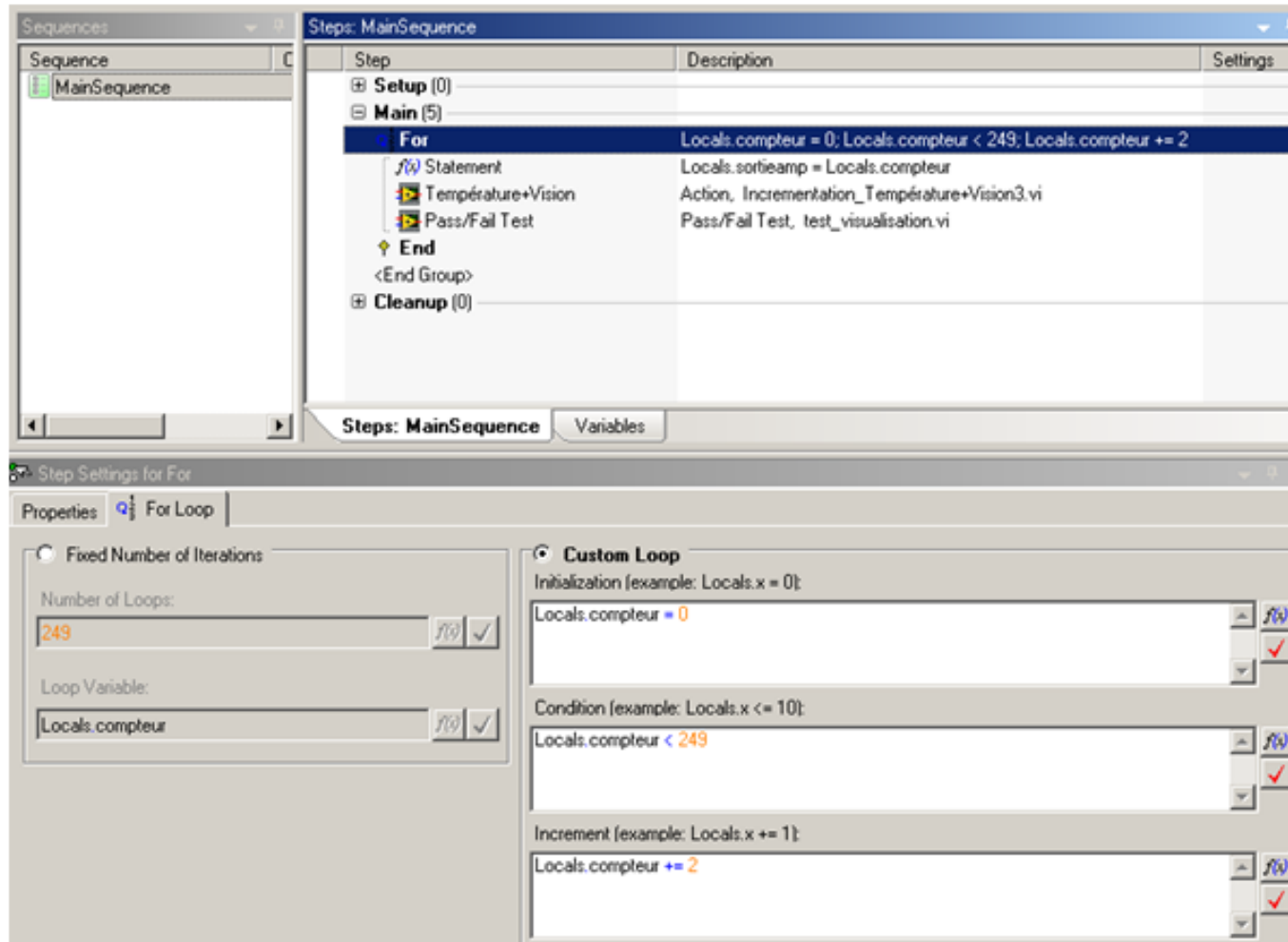
Mise en place de la validation avec TESTSTAND

Notre séquenceur comporte deux actions :

Steps: MainSequence		
Step	Description	Settings
⊕ Setup (0)		
⊖ Main (5)		
<i>For</i>	Locals.compteur = 0; Locals.compteu...	
<i>f(x) Statement</i>	Locals.sortieamp = Locals.compteur	
▶ Temperature+Vision	Action, Incrementation_Température..	
▶ Pass/Fail Test	Pass/Fail Test, test_visualisation.vi	
♦ End		
<End Group>		
⊕ Cleanup (0)		

L'une comporte l'envoi de la trame de la température et de l'acquisition de l'affichage de celle-ci. Et l'autre compare les valeurs envoyées sur l'EMF et celle de la température lue par la caméra.

Boucle FOR



The screenshot displays the VEGA software interface. The top window, 'Steps: MainSequence', shows a sequence of steps: Setup (0), Main (5), and Cleanup (0). The 'Main' step is expanded, revealing a 'For' loop configuration. The 'For' loop is defined by the expression 'Locals.compteur = 0; Locals.compteur < 249; Locals.compteur += 2'. The loop body contains three steps: a 'Statement' step (Locals.sortieamp = Locals.compteur), a 'Température+Vision' action step (Action, Incrementation_Température+Vision3.vi), and a 'Pass/Fail Test' step (Pass/Fail Test, test_visualisation.vi). The bottom window, 'Step Settings for For', provides detailed configuration for the 'For' loop. It shows the 'Fixed Number of Iterations' set to 249, the 'Loop Variable' as 'Locals.compteur', and the 'Custom Loop' settings: Initialization (Locals.compteur = 0), Condition (Locals.compteur < 249), and Increment (Locals.compteur += 2).

Step	Description	Settings
Setup (0)		
Main (5)		
For	Locals.compteur = 0; Locals.compteur < 249; Locals.compteur += 2	
Statement	Locals.sortieamp = Locals.compteur	
Température+Vision	Action, Incrementation_Température+Vision3.vi	
Pass/Fail Test	Pass/Fail Test, test_visualisation.vi	
End		
<End Group>		
Cleanup (0)		

Step Settings for For

Properties | **For Loop**

☐ Fixed Number of Iterations

Number of Loops: 249

Loop Variable: Locals.compteur

☒ Custom Loop

Initialization (example: Locals.x = 0):
Locals.compteur = 0

Condition (example: Locals.x <= 10):
Locals.compteur < 249

Increment (example: Locals.x += 1):
Locals.compteur += 2

La boucle FOR incrémente un compteur pour changer la valeur de la température affichée sur l'EMF. Quand la valeur du compteur est égale à 249, le test s'arrête.

ACTION Température et Reconnaissance

Cette Action regroupe deux programmes : un programme qui affiche la température sur l'EMF et un autre qui lit via la webcam la température affichée.

The screenshot displays the VEGA software interface. The top window, 'Steps: MainSequence', shows a sequence of steps: Setup (0), Main (5), and Cleanup (0). The 'Main' group contains a 'For' loop with a 'Statement' step, which is further expanded to show 'Température+Vision' and 'Pass/Fail Test'. The 'Température+Vision' step is selected, and its settings are shown in the bottom window.

The bottom window, 'Step Settings for Température+Vision', shows the 'Properties' tab. It displays the 'Incrementation_Température+Vision3.vi' module. The 'Module' tab is active, showing a table of parameters and their values.

Parameter Name	Type	In/Out	Default	Value
Sequence Context	Object Reference	in	☐	ThisContext
Numérique	Number (DBL)	in	☐	Locals.sortieamp
Report Text Visualisation	ASCII String	out		Step.Result.Report...
Sortie visualisation	Number (I32)	out		Locals.Sortie_Visu
SORTie Température	Number (DBL)	out		Locals.sortie_temp...
Report Text Température	ASCII String	out		Step.Result.Report...

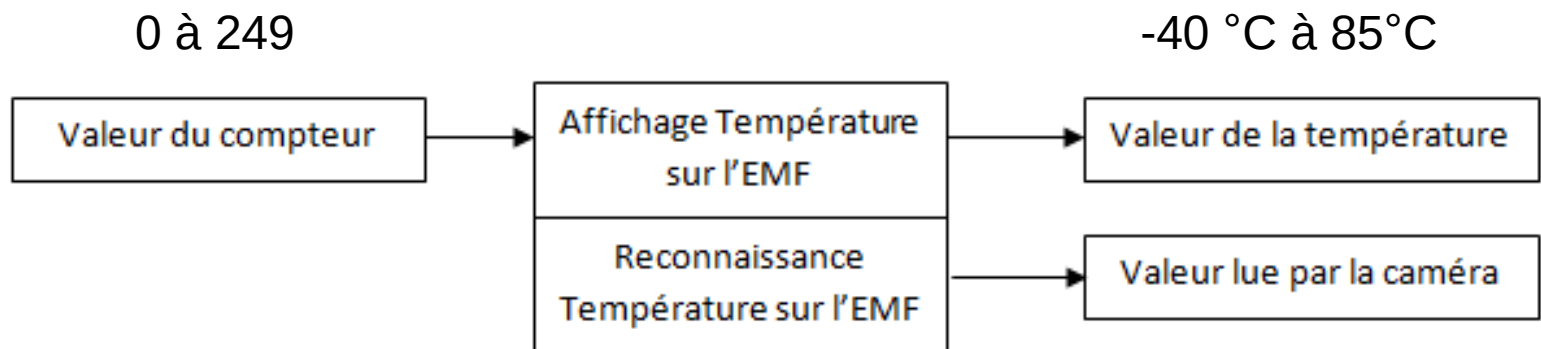
Below the table, a diagram shows the 'Incrementation_Température+Vision3.vi' module with its inputs and outputs. The inputs are 'Sequence Context' and 'Numérique'. The outputs are 'Report Text Visualisation', 'Sortie visualisation', 'SORTie Température', and 'Report Text Température'. A note below the diagram states: '<No VI description present>'. The 'Show VI Front Panel When Called' checkbox is also visible.

ACTION Température

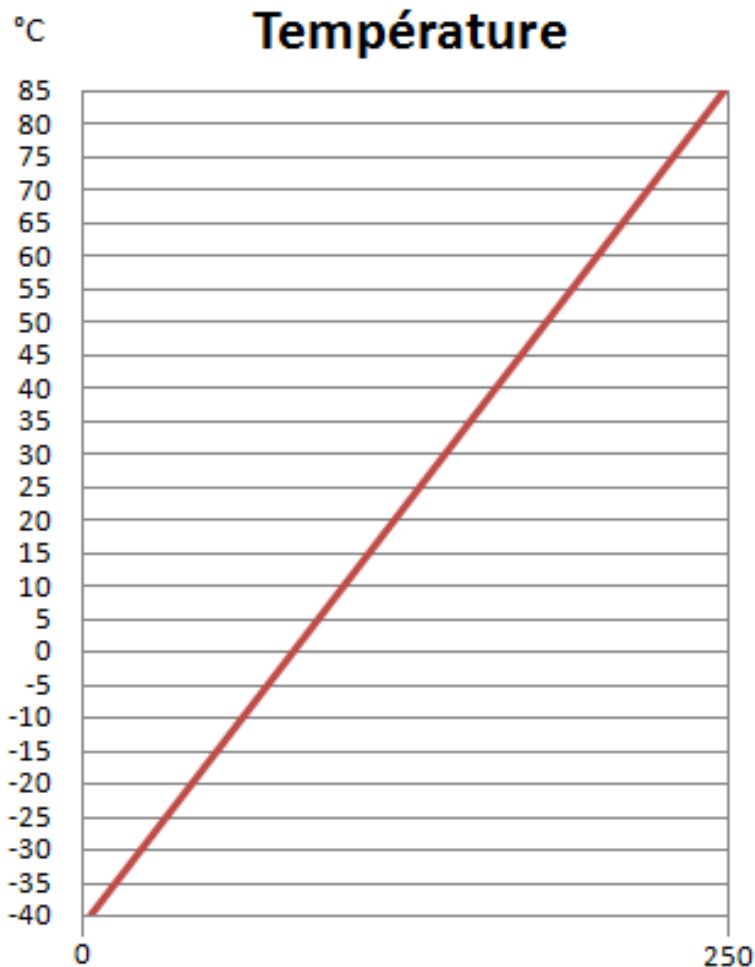
La valeur du compteur de la Boucle FOR est recopié à l'entrée de l'affichage de la température, a la fin du programme de cette action, deux variables sortent :

- Une variable indique la valeur de la température affiché (Ex : 40°C).
- Une variable indique la valeur lue sur l'écran multifonction.

Illustration du VI :



ACTION Température



Points :

A (0 ; -40)

B (250 ; 85)

Equation d'une droite :

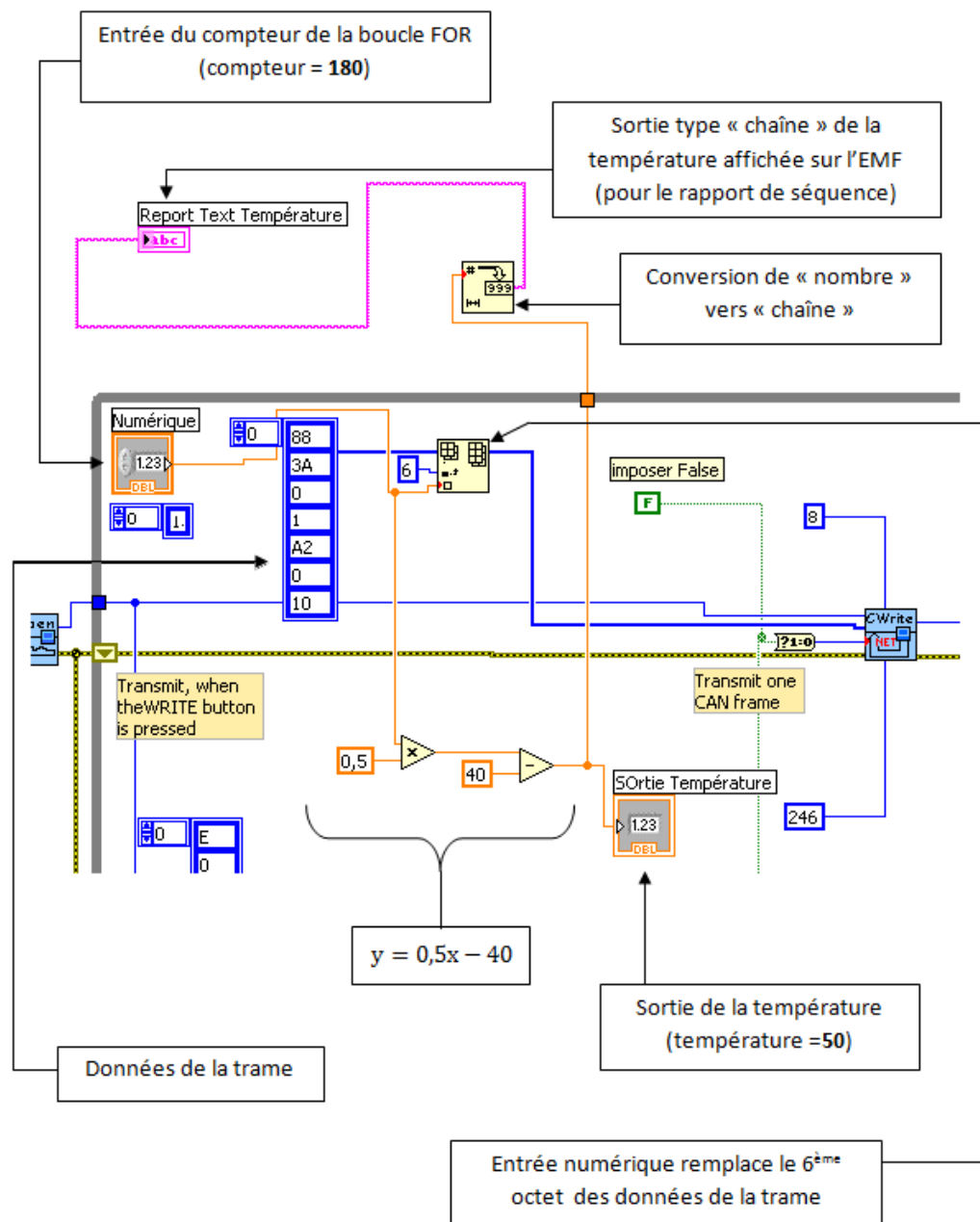
$$y = ax + b$$

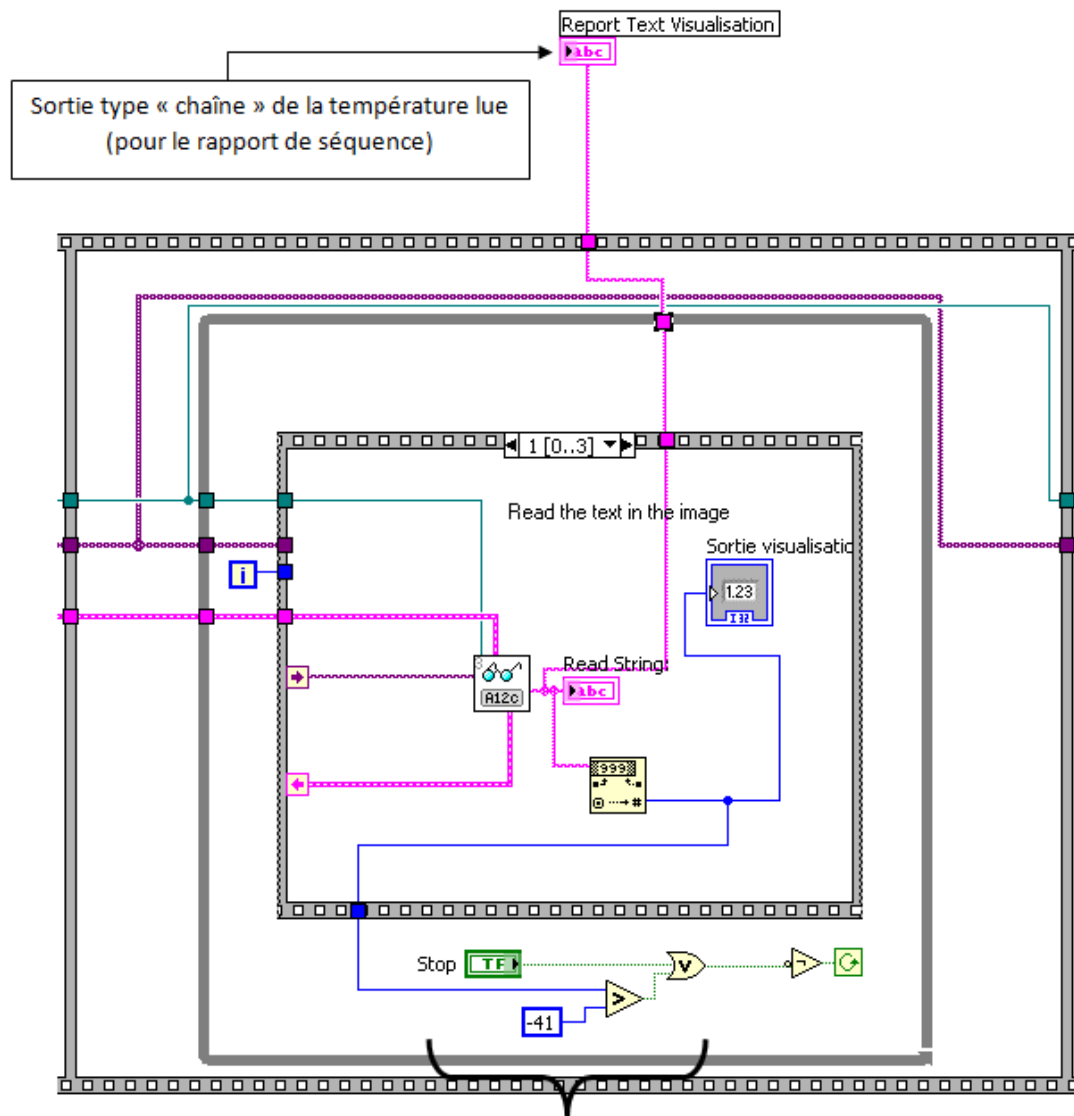
Coefficient directeur :

$$a = \frac{yb - ya}{xb - xa} = \frac{85 - (-40)}{250 - 0} = \frac{125}{250} = 0,5$$

Equation de la droite température :

$$y = 0,5x - 40$$

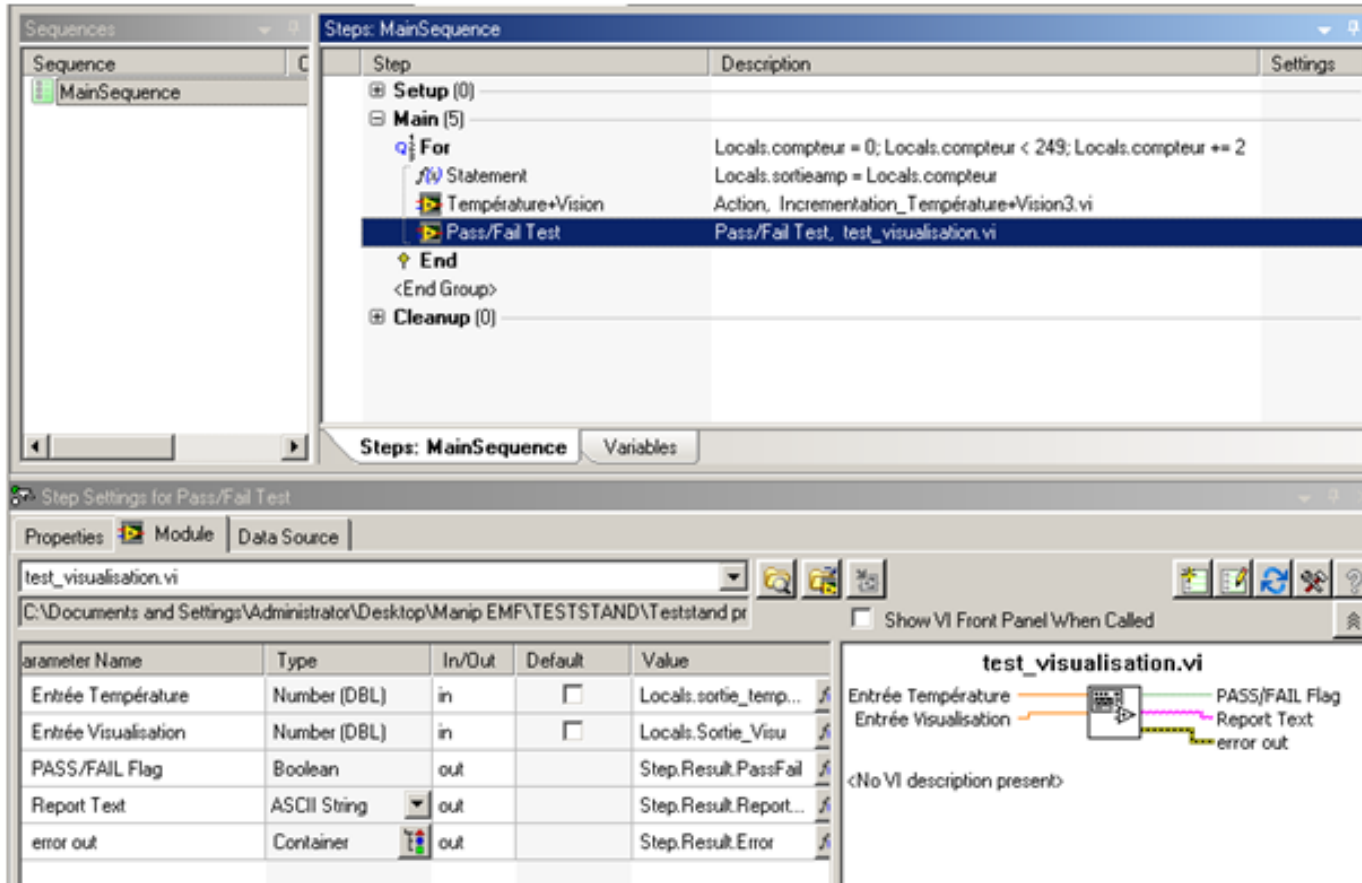




Pour pouvoir arrêter la boucle automatiquement après chaque lecture de valeur, nous avons rajouté un test ici permettant de sortir du VI.
(Sans cela, l'action restait bloquée dans TESTSTAND)

Mise en place de la validation avec TESTSTAND

Action : Pass/Fail Test



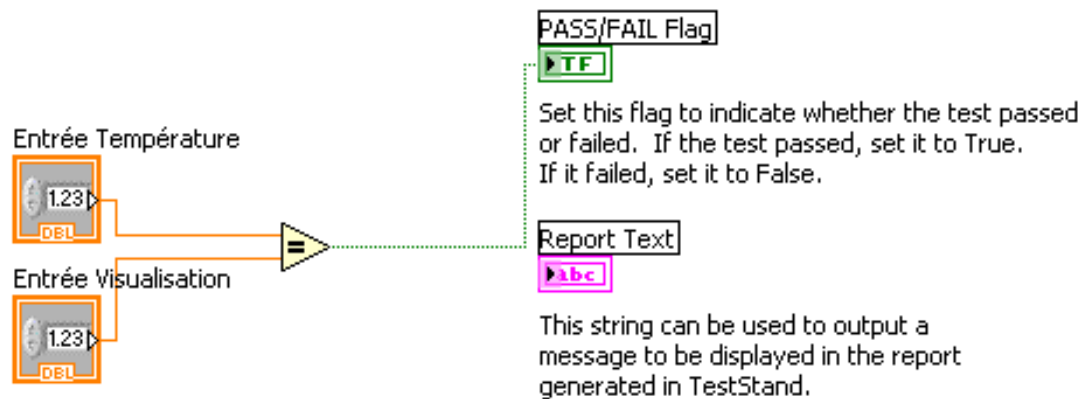
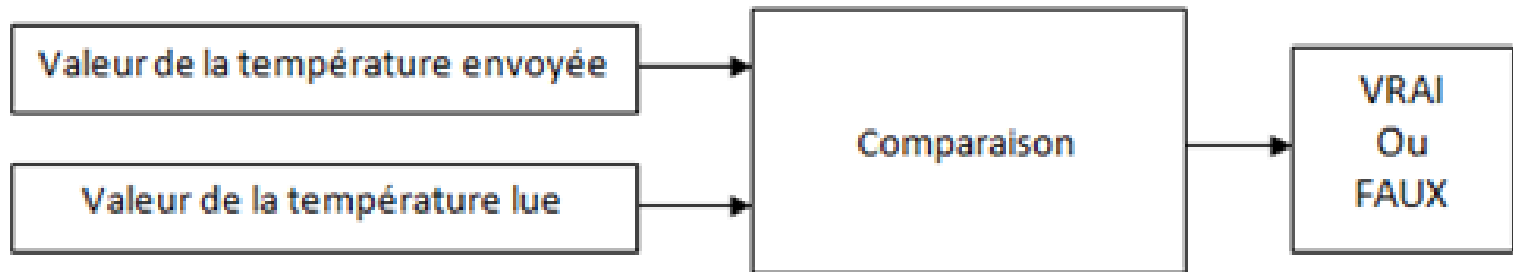
The screenshot displays the TestStand software interface. The top window, 'Steps: MainSequence', shows a sequence of steps: Setup (0), Main (5), and Cleanup (0). The 'Main' step is expanded, showing a 'For' loop with a 'Pass/Fail Test' step highlighted. The 'Pass/Fail Test' step is configured to call 'test_visualisation.vi'. Below this, the 'Step Settings for Pass/Fail Test' dialog is open, showing the 'Properties' tab. The 'Data Source' section shows the path 'C:\Documents and Settings\Administrator\Desktop\Manip EMF\TESTSTAND\Teststand pr'. The 'Parameters' table lists the inputs and outputs for the 'test_visualisation.vi' sub-VI.

Parameter Name	Type	In/Out	Default	Value
Entrée Température	Number (DBL)	in	☐	Locals.sortie_temp...
Entrée Visualisation	Number (DBL)	in	☐	Locals.Sortie_Visu...
PASS/FAIL Flag	Boolean	out		Step.Result.PassFail
Report Text	ASCII String	out		Step.Result.Report...
error out	Container	out		Step.Result.Error

The 'test_visualisation.vi' sub-VI is shown with its inputs and outputs: 'Entrée Température' and 'Entrée Visualisation' are inputs; 'PASS/FAIL Flag', 'Report Text', and 'error out' are outputs. The 'PASS/FAIL Flag' output is connected to a 'PASS/FAIL Flag' control.

Les deux valeurs du programme précédent sont recopiées à l'entrée de ce programme qui compare les deux valeurs et qui en sortie indique si c'est valide ou pas.

VI PASS/FAIL



Extrait Séquence de TEST

Step	Status	Measurement	Units	Limits		
				Low Limit	High Limit	Comparison Type
Statement	Done					
Temperature	Done					
Report Text:	36					
Vision	Done					
Report Text:	37					
Pass/Fail Test	Failed					

Température affichée

Température lue par la webcam

Step	Status	Measurement	Units	Limits		
				Low Limit	High Limit	Comparison Type
End For	Done					
For {0x9a} Locals.compteur = 0; Locals.compteur < 249; Locals.compteur += 2	Done					

Reprise de la boucle FOR :
Incrémentation compteur

Step	Status	Measurement	Units	Limits		
				Low Limit	High Limit	Comparison Type
Statement	Done					
Temperature	Done					
Report Text:	37					
Vision	Done					
Report Text:	37					
Pass/Fail Test	Passed					

Température affichée

Température lue par la webcam

Conclusion

Le banc d'essai fonctionne mais la reconnaissance n'est pas très fiable à cause :

- de la luminosité
- de la résolution de la caméra