

CALCUL DE L'IMAGE DEMODULEE

Durée : 2 H.

A. OBJECTIF DU T.P.

Plan du T.P.

A. Objectif du T.P.

B. Travail demandé

B.1. But

B.2. Cahier des charges

B.3. Analyse du programme

On désire faire une application complète qui permet la génération de quatre mires sinusoïdales puis du calcul de l'image phasée sous LabView. Pour ce TP, on ne réalisera que la deuxième partie : calcul des images phasées puis création de l'application principale.

B. TRAVAIL DEMANDÉ

B.1. But :

On veut écrire un programme qui permet de générer quatre mires sinusoïdales puis qui permet le calcul de l'image phasée de ces quatre mires puis la soustraction de l'image phasée objet et de l'image phasée référence puis la démodulation de cette soustraction.

B.2. Cahier des charges :

- ✚ On souhaite réaliser un programme qui crée 4 mires sinusoïdales décalées de $\pi/2$ dont la pas en pixels est donné par l'utilisateur suite à une question posée à travers une boîte de dialogue et affichage de ces mires dans 4 fenêtres externe au VI.
- ✚ Avec ces 4 images, on calcule l'image phasée $\varphi = \text{Arc tan} \frac{I_4 - I_2}{I_1 - I_3}$ que l'on stocke dans la variable globale « Image référence » pour la référence et « Image Objet » pour l'objet.
- ✚ On calcule la soustraction modulo 256 de « Image référence » et de « Image Objet ».
- ✚ On démodule cette image pour faire une représentation 3D.

B.3. Analyse du programme :

B.3.1. Création des VIs « MireREF » et « MireOBJ »

Vu dans le TP19.

B.3.2. Création des VIs « PhaseREF » et « PhaseOBJ »

Vu dans le TP20.

B.3.3. Création du VI « ImModulée »

Sur la face avant :

Un indicateur d'image « ImModulée »

Un bouton « Sortir ».

Sur le diagramme :

A l'initialisation, calculer l'image modulée : « lphasobj » - « lphasref » modulo 256. Stocker cette image dans « ImModulée » et sur une fenêtre extérieure nommée « Image modulée ».

Dans la boucle While :

Si bouton « sortir » Vrai ALORS stocker « ImModulée » dans la variable globale « ImModulée » puis quitter le VI.

Sauvegarder le VI sous « ImModulee.VI » dans votre répertoire

Sauvegarder à nouveau les variables globales « VGPHASEE.VI » dans votre répertoire

B.3.4. Création du VI « ImDémodulée »

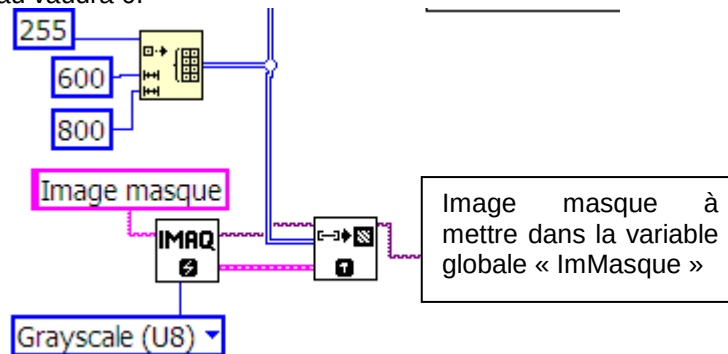
Sur la face avant :

Un indicateur d'image « ImDémodulée »

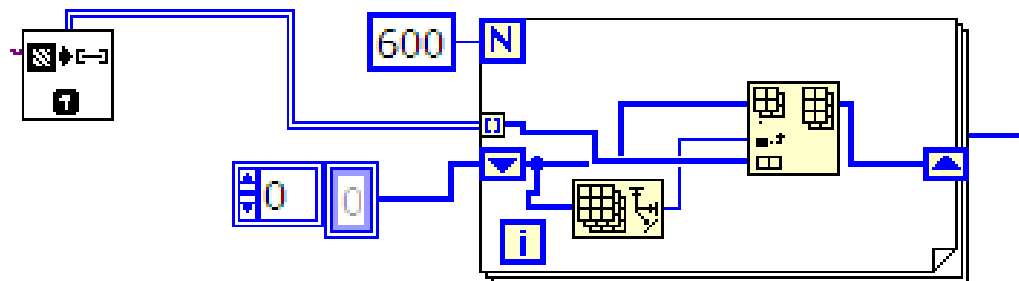
Un bouton « Sortir ».

Sur le diagramme :

Il faut créer une fausse image masque, on créera un tableau d'entier U8 initialiser en taille 800*600 et chaque valeur du tableau vaudra 0.



Il faut dans un premier tant, prendre l'image et mettre chaque ligne à la suite de l'autre.

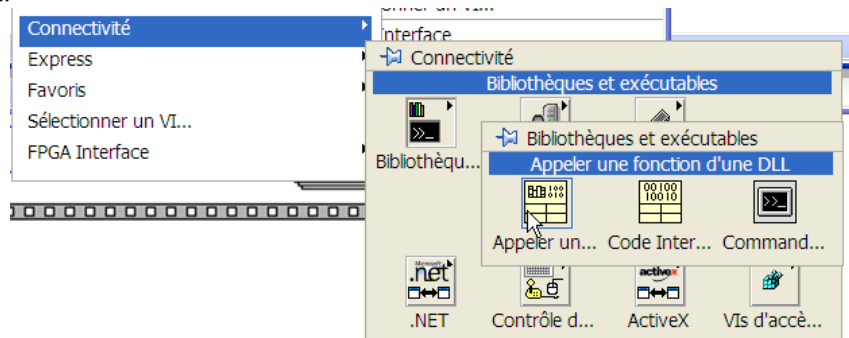


On utilisera :

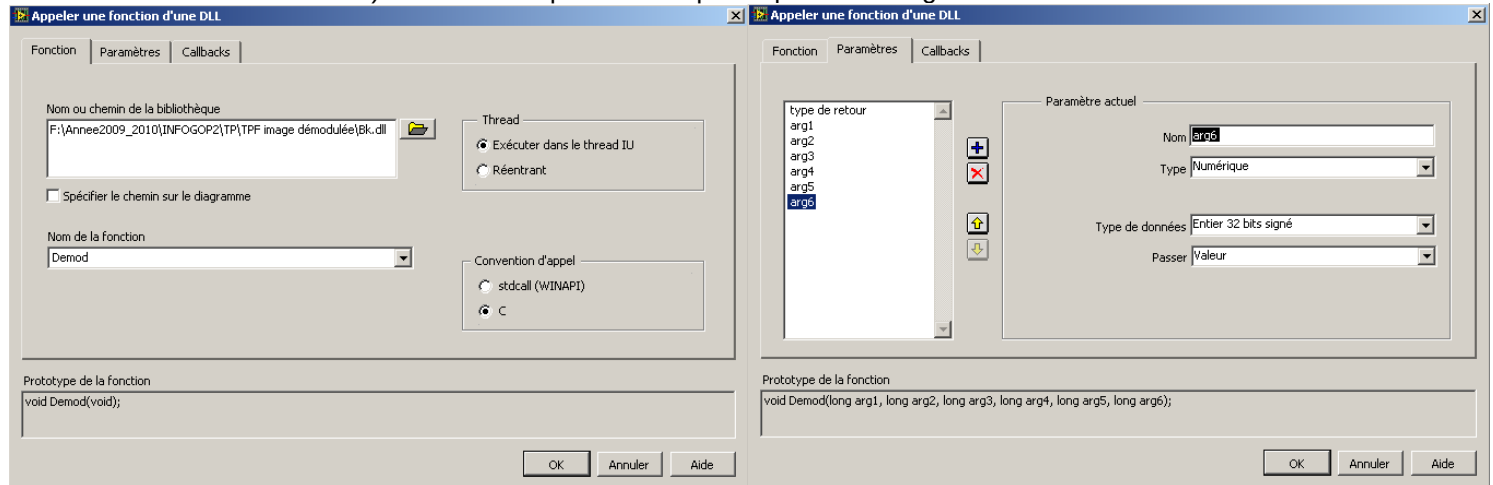
Cette opération sera à faire pour l'image modulée et l'image masque.

Utilisation de la dll :

Il faut appeler la dll : on utilisera la fonction « Appeler une dll » dans « Connectivité »-« Bibliothèques et exécutables »



Pour la fonction Demod, il faut configurer les entrées et sorties suivant son prototype : short int Demod(short int Nbre col, short int Nbre lignes, long *Im source, long *Im masque, long *Taille im, long Mode demod). Il suffit de cliquer sur « le plus » pour les 6 arguments.



Appeler une fonction d'une DLL

Fonction Paramètres Callbacks

Nom ou chemin de la bibliothèque
F:\Annee2009_2010\INFOGOP2\TP\TPF image démodulée\Bk.dll

Thread
☒ Exécuter dans le thread IU
☐ Réentrant

Spécifier le chemin sur le diagramme

Nom de la fonction
Demod

Convention d'appel
☐ stdcall (WINAPI)
☒ C

Prototype de la fonction
void Demod(void);

OK Annuler Aide

Appeler une fonction d'une DLL

Fonction Paramètres Callbacks

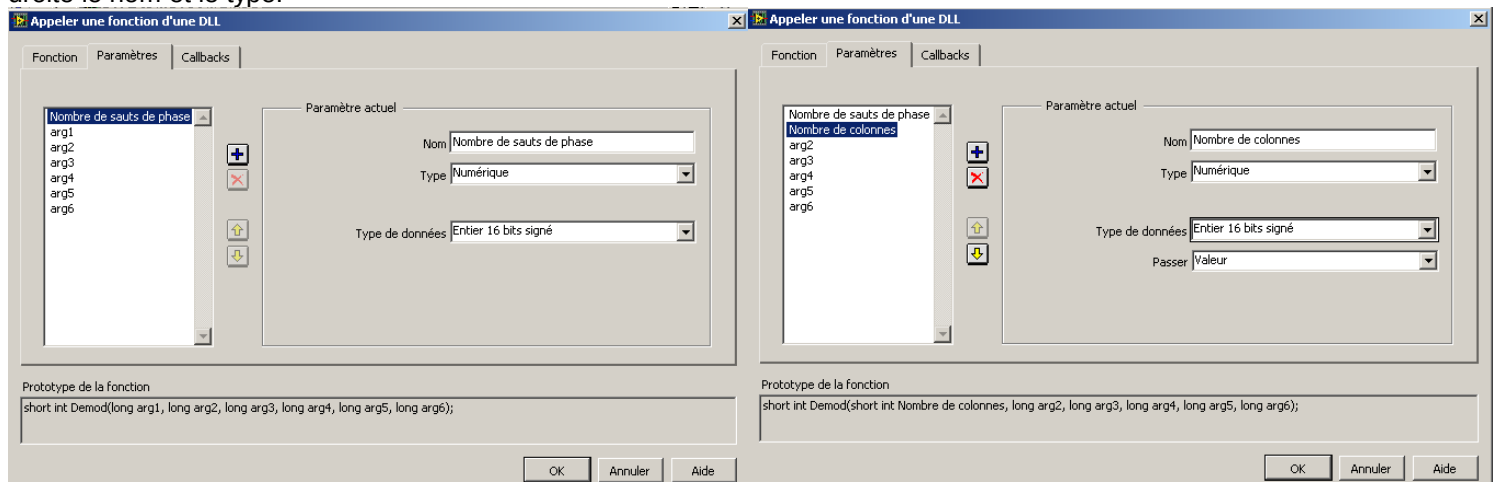
type de retour
arg1
arg2
arg3
arg4
arg5
arg6

Paramètre actuel
Nom: arg6
Type: Numérique
Type de données: Entier 32 bits signé
Passer: Valeur

Prototype de la fonction
void Demod(long arg1, long arg2, long arg3, long arg4, long arg5, long arg6);

OK Annuler Aide

Il faut caractériser le type de variable de retour et d'argument. Pour cela cliquez sur la variable à modifier et adapter à droite le nom et le type.



Appeler une fonction d'une DLL

Fonction Paramètres Callbacks

Nombre de sauts de phase
arg1
arg2
arg3
arg4
arg5
arg6

Paramètre actuel
Nom: Nombre de sauts de phase
Type: Numérique
Type de données: Entier 16 bits signé

Prototype de la fonction
short int Demod(long arg1, long arg2, long arg3, long arg4, long arg5, long arg6);

OK Annuler Aide

Appeler une fonction d'une DLL

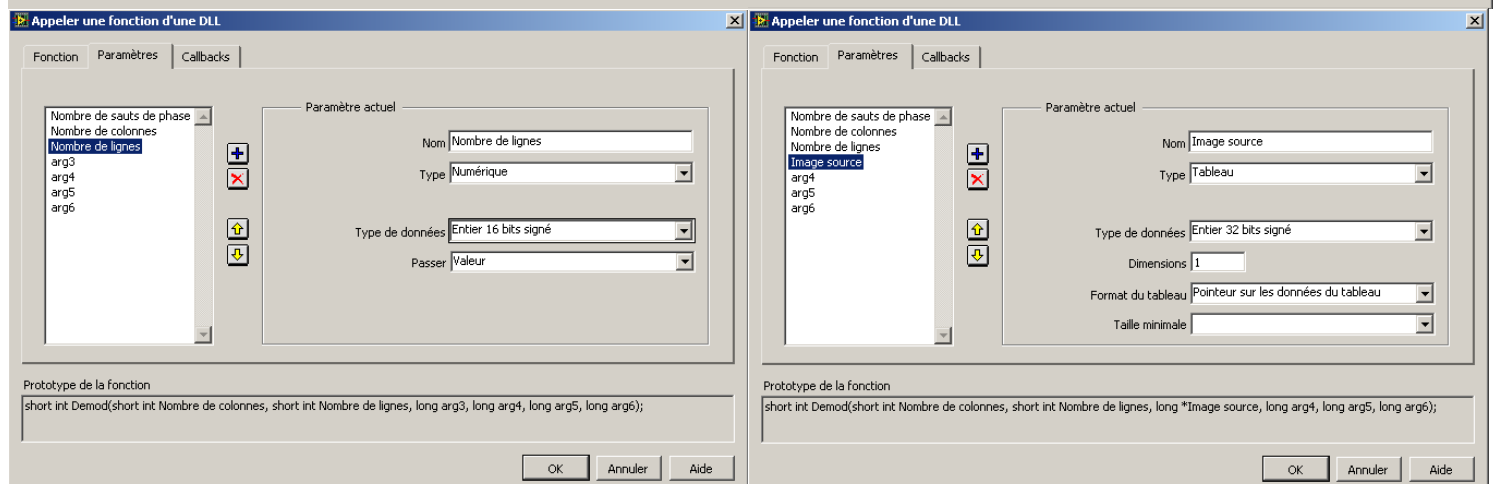
Fonction Paramètres Callbacks

Nombre de sauts de phase
Nombre de colonnes
arg2
arg3
arg4
arg5
arg6

Paramètre actuel
Nom: Nombre de colonnes
Type: Numérique
Type de données: Entier 16 bits signé
Passer: Valeur

Prototype de la fonction
short int Demod(short int Nombre de colonnes, long arg2, long arg3, long arg4, long arg5, long arg6);

OK Annuler Aide



Appeler une fonction d'une DLL

Fonction Paramètres Callbacks

Nombre de sauts de phase
Nombre de colonnes
Nombre de lignes
arg3
arg4
arg5
arg6

Paramètre actuel
Nom: Nombre de lignes
Type: Numérique
Type de données: Entier 16 bits signé
Passer: Valeur

Prototype de la fonction
short int Demod(short int Nombre de colonnes, short int Nombre de lignes, long arg3, long arg4, long arg5, long arg6);

OK Annuler Aide

Appeler une fonction d'une DLL

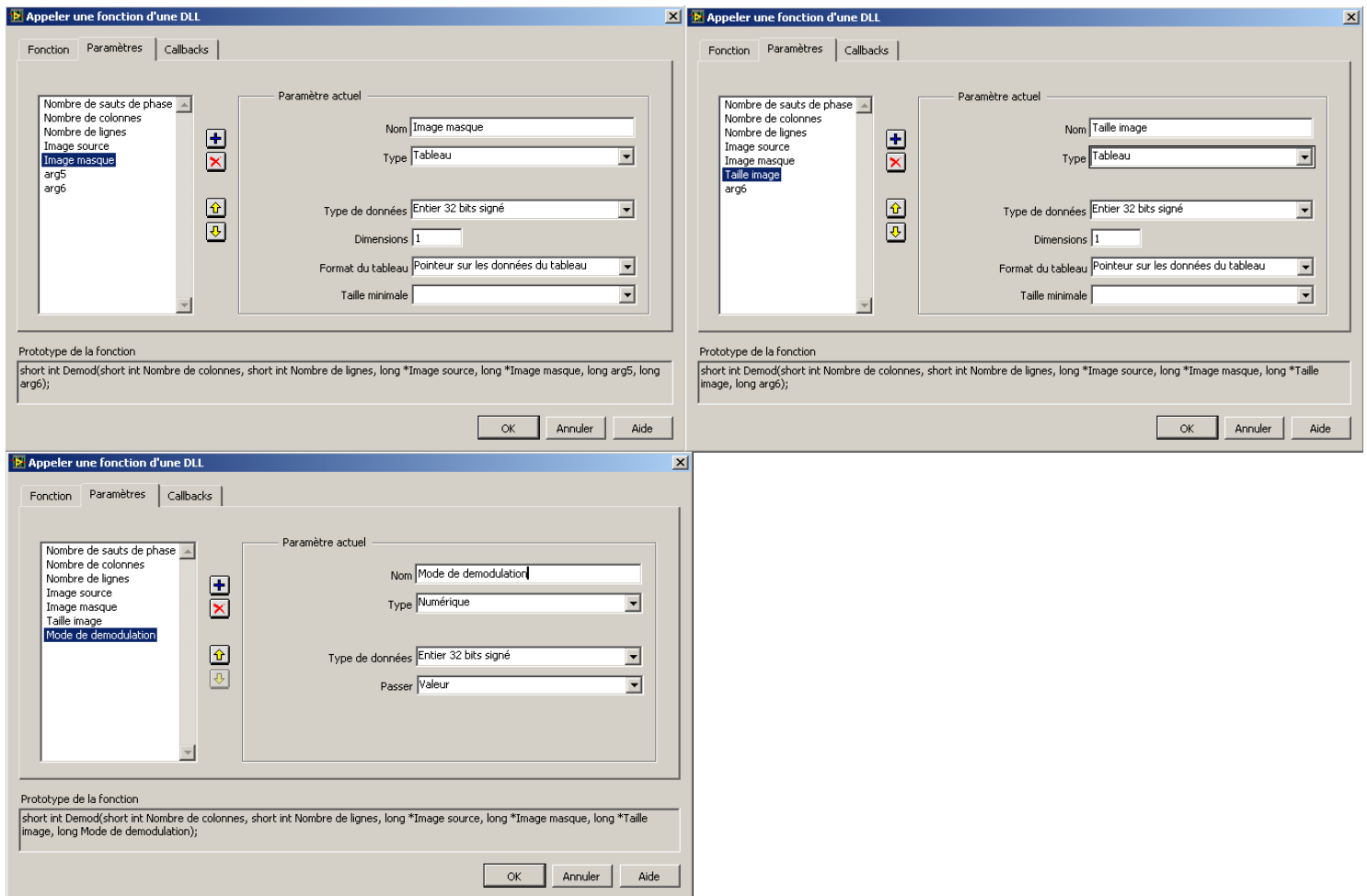
Fonction Paramètres Callbacks

Nombre de sauts de phase
Nombre de colonnes
Nombre de lignes
Image source
arg4
arg5
arg6

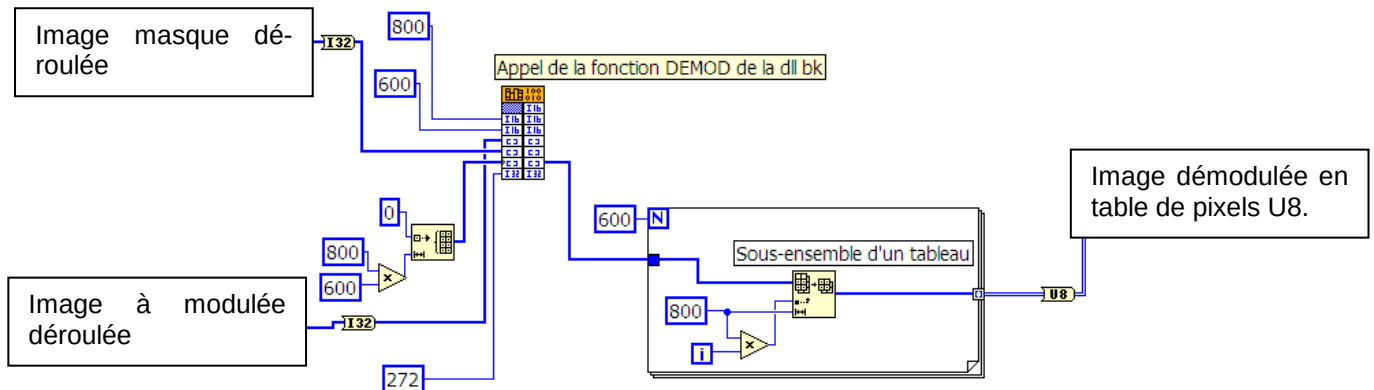
Paramètre actuel
Nom: Image source
Type: Tableau
Type de données: Entier 32 bits signé
Dimensions: 1
Format du tableau: Pointeur sur les données du tableau
Taille minimale:

Prototype de la fonction
short int Demod(short int Nombre de colonnes, short int Nombre de lignes, long *Image source, long arg4, long arg5, long arg6);

OK Annuler Aide



Reste à câbler la dll comme ci-dessous :



Convertir l'image démodulée en table de pixels en image et la stocker dans « ImDémodulée » et sur une fenêtre extérieure nommée « Image démodulée ».

Dans la boucle While :

Si bouton « sortir » Vrai ALORS stocker « ImDémodulée » dans la variable globale « ImDémodulée » puis quitter le VI.

Sauvegarder le VI sous « ImDemodulée.VI » dans Votre répertoire

Sauvegarder à nouveau les variables globales « VGPHASEE.VI » dans votre répertoire

B.3.5. Création du VI « Visu3D » :

Sur la face avant :

Un graphe de surface 3D « Visu3D »

Un bouton « Sortir ».

Sur le diagramme :

A l'initialisation, convertir « ImDémodulée » en table de pixels puis tracer dans « Visu3D ».

Dans la boucle While :

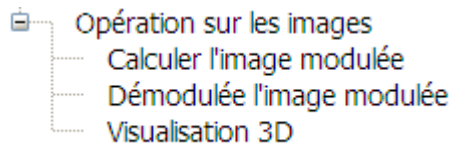
Si bouton « sortir » Vrai ALORS quitter le VI.

Sauvegarder le VI sous « Visu3D.VI » dans votre répertoire

B.3.6. Modification du programme principal :

Ouvrir le VI « Imagerie.vi »

Modifier le menu, rajouter :



Enregistrer sous « TP192021.rtm » dans votre répertoire

Sur le diagramme rajouter les 3 conditions correspondant aux modifications du menu.

Enregistrer le Vi sous « Imagerie.VI » dans votre répertoire