

CALCUL DE L'IMAGE DEMODULEE

<u>Durée</u> : 2 H.

A. OBJECTIF DU T.P.

Plan du T.P.

A. Objectif du T.P.

- B. <u>Travail demandé</u>
 - 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 phasée 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 créée 4 mires sinusoïdales décalées de π/2 dont la pas en pixels est donné par l'utilisateur suite à une question posée à travers une boite de dialogue et affichage de ces mires dans 4 fenêtres externe au VI.
- + Avec ces 4 images, on calcul l'image phasée $\varphi = \operatorname{Arc} \tan \frac{I_4 I_2}{I_1 I_3}$ que l'on stocke dans la variable glo-

bale « Image référence » pour la référence et « Image Objet» pour l'objet.

- 4 On calcule la soustraction modulo 256 de « Image référence » et de « Image Objet».
- 4 On démodule cette image pour faire une représentation 3D.

B.3. <u>Analyse du programme :</u>

- B.3.1. <u>Création des VIs « MireREF » et « MireOBJ »</u> Vu dans le TP19.
- B.3.2. <u>Création des VIs « PhaseREF» et « PhaseOBJ »</u> 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 : « Iphasobj » - « Iphasref » modulo 256. Stocker cette image dans « ImModulée» et sur une fenêtre extérieure nommée « Image modulée ».



TP21 : Acquisition et traitement de l'image

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 « Connectivite »-« Bibliothèques et exécutables »





TP21 : Acquisition et traitement de l'image

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.

Reference to the second s	Appeler une ronction d'une DLL	스
Proteion Paramètres Callbacks Nom ou chemin de la bibliothèque Thread F:\Annee2009_2010\INFOGOP2\TP\TPF image démodulée\Bk.dl Image: Callbacks Spécfier le chemin sur le diagramme Image: Convention d'appel Nom de la fonction Image: Convention d'appel Demod Image: Convention d'appel Prototype de la fonction Image: Convention void Demod(void); Image: Convention	Image: Service on concord of due DL Fonction Paramètres Callbacks Image: Service of the concord of due DL Image: Service of the concord of the conconcord of the concord of the concord of the	
OK Annuler Aide	OK Annule	r 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.

The Appeler due fonction of the DLL	
Fonction Paramètres Callbacks	Fonction Paramètres Callbacks
Nombre de sauts de phase Paramètre actuel arg1 Nom Nombre de sauts de phase arg2 Type Numérique arg5 Type de données Entier 16 bits signé Type	Nombre de sauts de phase Paramètre actuel Nom/Nor de colonnes arg2 arg3 arg4 arg6 Immérique Immérique Immérique Im
Bratatuna da la fanction	Prototype de la fonction
riocoppe de la fonction chart int Demod/long ava1, long ava2, long ava3, long ava5, long ava5, long ava5)	short int Demod(short int Nombre de colonnes, long arg2, long arg3, long arg4, long arg5, long arg6);
short inc Demou(long arg1, long arg2, long arg3, long arg4, long arg5, long arg5);	
)	
OK Annuler Aide	OK Annuler Aide
Appeler une fonction d'une DLL	X Appeler une fonction d'une DLL
Appeler une fonction d'une DLL Fonction Paramètres Callbacks	Endpoint of the provided of the provide
Paramètres Callbacks Fonction Paramètres Callbacks Paramètre actuel Nombre de sauts de phase Nombre de lignes arg3 arg4 arg5 Type Aumérique Image: Callbacks	Image source Nom Image source Image source Type Tableau
Paramètres Callbacks Fonction Paramètres Callbacks Paramètre actuel Nombre de souts de phase Nom Nombre de lignes arg5 Type arg5 Type de données Entier 16 bits signé Passer Valeur	Image: Source arg6 arg6 Image: Source arg6 arg6 Image: Source arg6 arg6 Image: Source arg6 arg6
Paramètres Callbacks Fonction Paramètres Callbacks Paramètre actuel Nombre de souts de phase Nom Nombre de lignes arg5 Type arg6 Image: State of the soute o	Image source Image source Image source
Paramètres Callbacks Fonction Paramètres Callbacks Paramètre actuel Nombre de souts de phase Nom Nombre de lignes arg4 Type arg5 Type de données Entier 16 bits signé Passer Valeur Valeur	Image source Nombre de sauts de phase Nombre de sauts de phase Nombre de colonnes Nombre de colonnes Nom Image source Image source Type Tableau arg6 Image source arg6 Image source Image source Type Tableau Image source Image source
Paramètres Callbacks Fonction Paramètres Callbacks Paramètre actuel Nombre de souts de phase Nom Nombre de lignes Type arg3 Type arg5 Type de données Entier 16 bits signé Passer Valeur Valeur	Image source Nombre de sauts de phase Paramètre actuel Nombre de colonnes Nom Image source Nombre de lignes Image source Image source Type [Tableau arg5 Image source arg6 Image source Image source Type [Tableau Image source Image source Image source Type [Tableau Image source Image source
Appeler une fonction d'une DLL Fonction Paramètres Calibacks Nombre de sauts de phase Nombre de lignes arg3 arg4 arg5 arg6 Image: State of the source of t	Image source Nombre de sauts de phase Image source Nom Image source arg6 Image source arg6 Image source Image source Type Tableau Image source Image source arg6 Image source Image source Type Tableau Image source Image source Image source Image source <t< td=""></t<>
Paramètres Calibacks Fonction Paramètres Calibacks Paramètre actuel Nombre de souts de phase Nom Nombre de lignes arg6 Type arg6 Image: Second Se	Y Paramètres Calibacks Fonction Paramètres Calibacks Nombre de sauts de phase Nom finage source Nombre de lignes Nom finage source arg6 Ype arg6 Ype arg6 Ype Type de données Enter 32 bits signé Dimensions 1 Format du tableau Yee Taile minimale Yee
Papeler une fonction d'une DLL Fonction Paramètres Calibacks Nombre de sauts de phase Nombre de colonnes Wombre de lignes arg6 arg6 arg6 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);	Y Appeler une fonction d'une DLL Image source Fonction Paramètres Calbacks Nombre de sauts de phase Image source Nombre de lignes Nom Image source arg6 Image source arg6 Image source Image source Image Image source Im



TP21 : Acquisition et traitement de l'image

😰 Appeler une fonction d'une DLL	🖌 🔯 Appeler une fonction d'une DLL 📃 🔀
	S Paramètre fonction d'une DLL Fonction Paramètres Callbacks Nombre de sauts de phase Nombre de colonnes Nombre de colonnes Nombre de sauts de phase Image masque Type Type Tableau Image masque argé Type de données Entier 32 bits signé Dimensions 1 Format du tableau Pointeur sur les données du tableau Taille minimale
OK Annuler Aide	OK Annuler Aide
Appeler une fonction d'une DLL	×
Fonction Paramètres Callbacks Nombre de sauts de phase Image acues Nom Mode de demodulation Nombre de lignes Image acues Type Mumérique Talle image Image acues Type de données Mode de demodulation Image acues Image acues Type de données Entier 32 bits signé Image acues Image acues Image acues Image a	
Prototype de la fonction short ht Demodishort int Nombre de colonnes, short int Nombre de linnes, long *Tmage source, long *Tmage margue, long *Tmile	-
image, long Mode de demodulation); [mage, long Mode de demodulation);	
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 « ImDemodulée » et sur une fenêtre extérieure nommée « Image démodulée ».

Dans la boucle While :

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

Sauvegarder le VI sous « ImDemodulee.VI » dans Votre répertoire Sauvegarder à nouveau les variables globales « VGPHASEE.VI » dans votre répertoire



B.3.5. <u>Création du VI « Visu3D » :</u>

Sur la face avant :

Un graphe de surface 3D « Visu3D » Un bouton « Sortir ».

Sur le digramme :

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 :

Opération sur les images
Calculer l'image modulée
Démodulée l'image modulée
Visualisation 3D

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