


Présentation des fonctions de la bibliothèque LabView BIOLOÏB

La bibliothèque LabView BioIoïd créée à partir du fichier DLL « dynamixel.dll » comprend 21 fonctions. Ces fonctions ne sont pas toutes utiles pour accéder aux servomoteurs et capteurs connectés à l'interface USB2Dynamixel. Seules les fonctions d'initialisation de la communication et les fonctions de transmission de paquets de haut niveau sont nécessaires, soit 8 fonctions au total.

Ces 8 fonctions sont présentées ci-après en détail.


	dxi_initialize	dxi_initialize.vi
Initialise la connexion avec le boîtier USB2Dynamixel		

Entrée : **devindex** : Numéro de port COM
Baudnum : Vitesse de communication suivant la valeur du tableau ci-dessous

Valeur	Vitesse BPS
1	1000000.0
3	500000.0
4	400000.0
7	250000.0
9	200000.0
16	115200.0
34	57600,0
103	19200,0
207	9600,0


Sortie : Code retour : 1 si OK, 0 si erreur.

Le numéro de port dépend du matériel présent sur l'ordinateur et du type de communication avec le contrôleur (USB ou bluetooth).

	dxi_terminate	dxi_terminate.vi
Met fin à la communication avec le boîtier USB2Dynamixel		

Entrée : rien


Sortie : rien

	dxi_ping	dxi_ping.vi
Teste la présence d'un servomoteur ou d'un capteur		

Entrée : id : identificateur du servomoteur ou du capteur à tester

Sortie : rien


La réponse du test doit être lue ensuite avec la fonction « dxl_get_result » ; Le test est concluant si la valeur renvoyée par « dxl_get_result » est 0 (COMM_TXSUCCESS)

	dxl_get_result	dxl_get_result.vi
	Renvoie l'octet de statu suite à une communication de lecture ou écriture sur le bus.	

Entrée : rien

Sortie : octet de statu obtenu suite à la communication suivant le tableau ci-dessous


Octet	Nom	Message
0	COMM_TXSUCCESS	Paquet d'instructions transmis avec succès
1	COMM_RXSUCCESS	Paquet d'instructions reçu avec succès
2	COMM_TXFAIL	Erreur lors de la transmission du paquet d'instructions
3	COMM_RXFAIL	Erreur lors de la reception du paquet d'instructions
4	COMM_TXERROR	Problème dans le codage du paquet d'instructions
5	COMM_RXWAITING	Status non encore parvenu
6	COMM_RXTIMEOUT	L'actionneur ou le capteur ne répond pas
7	COMM_RXCORRUPT	Problème dans le statu du Paquet

	dxl_read_byte	dxl_read_byte.vi
	Lit un octet de données depuis la mémoire du servomoteur ou du capteur identifié	

Entrée : id : identificateur du servomoteur ou du capteur à tester
address : adresse de lecture dans la mémoire

Sortie : valeur de l'octet lu dans la mémoire.


Remarque : Utiliser la fonction « dxl_get_result » pour voir si aucun problème de communication n'est survenu pendant la lecture.

	dxl_read_word	dxl_read_word.vi
	Lit un mot de 2 octets de données depuis la mémoire du servomoteur ou du capteur identifié	

Entrée : id : identificateur du servomoteur ou du capteur à tester
address : adresse de lecture dans la mémoire

Sortie : valeur du mot lu dans la mémoire.


Remarque : Utiliser la fonction « dxl_get_result » pour voir si aucun problème de communication n'est survenu pendant la lecture.

	dxl_write_byte	dxl_write_byte.vi
	Ecrit un octet de données dans la mémoire du servomoteur ou du capteur identifié	

Entrée : id : identificateur du servomoteur ou du capteur à tester
address : adresse d'écriture dans la mémoire
value : valeur de l'octet à écrire

Sortie : rien.

Remarque : Utiliser la fonction « dxl_get_result » pour voir si aucun problème de communication n'est survenu pendant l'écriture.

	dxl_write_word	dxl_write_word.vi
	Ecrit un mot de deux octets de données dans la mémoire du servomoteur ou du capteur identifié	

Entrée : id : identificateur du servomoteur ou du capteur à tester
address : adresse d'écriture dans la mémoire
value : valeur du mot à écrire

Sortie : rien.

Remarque : Utiliser la fonction « dxl_get_result » pour voir si aucun problème de communication n'est survenu pendant l'écriture.