

Systeme de controle d'axes

GalilV2 : ControlBox

Auteur : J. Coquet

Version courante du document : 1.1

Date de creation document : Mars 2007

Derniere modification : le 21 mai 2007

Historique des modifications

Date	Revision	Description	Auteur	Relecteur
13/03/07		Version initiale	J. Coquet	
21/05/07		Modification mineures	A.Buteau	

1	But de l'application.....	1
2	Structure materielle et logicielle de l'application	2
2.1	Architecture materielle	2
2.2	Architecture logicielle	3
3	Que sait faire ControlBox ?	3
4	Interface logicielle Tango.....	3
4.1	Properties.....	3
4.2	Les attributs	4
4.3	Les commandes.....	4
5	La mise en oeuvre d'une ControlBox	5
5.1	Configuration device ControlBox	5

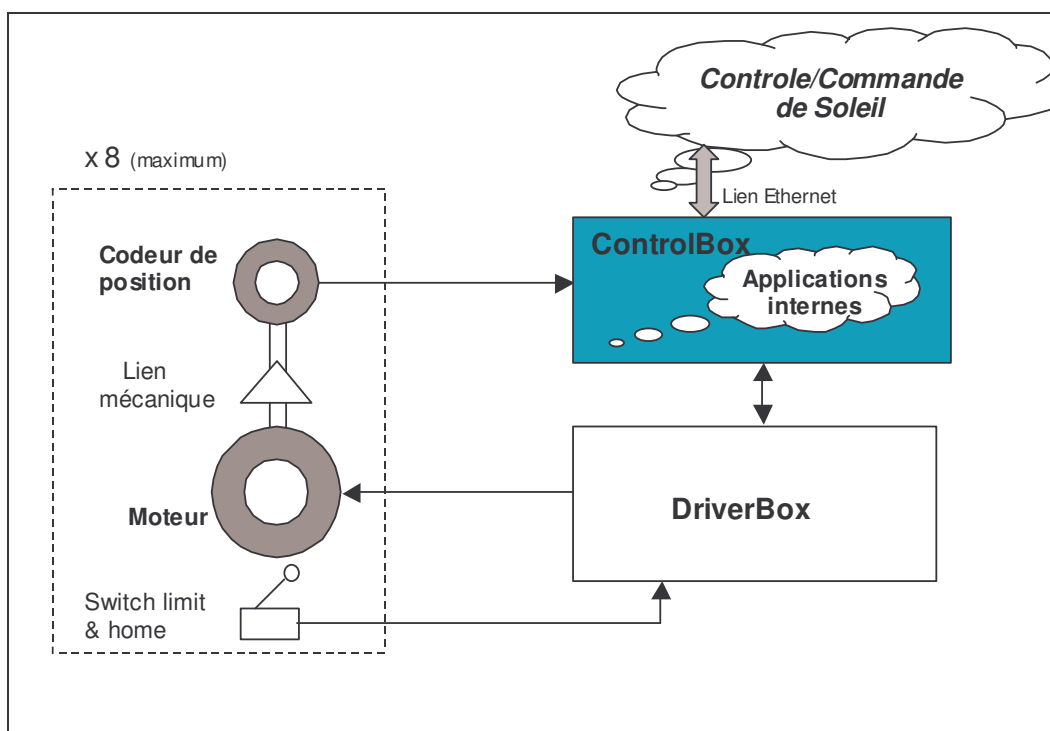
1 But de l'application

Ce device est l'interface de la ControlBox dans son ensemble.

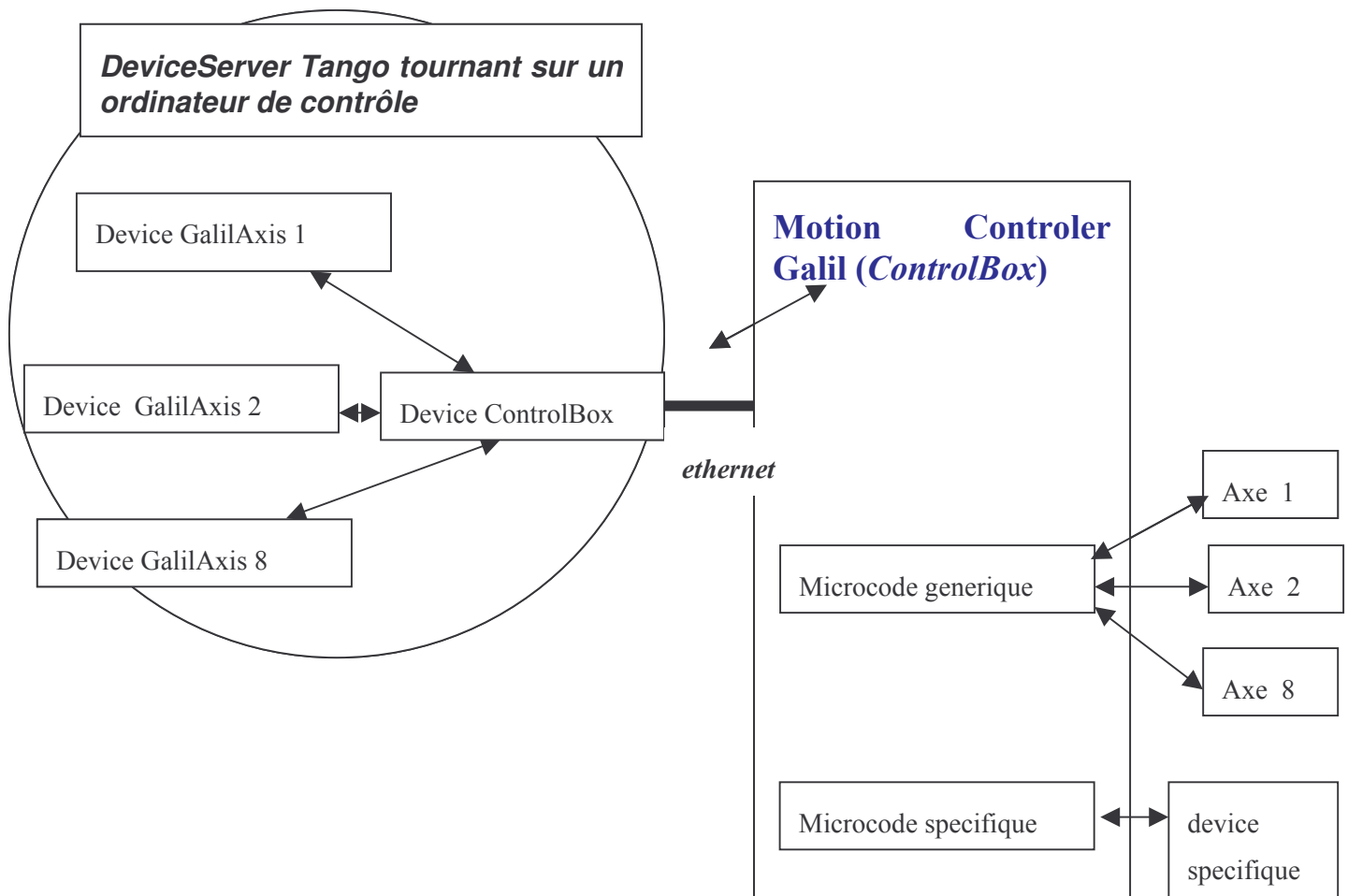
Il porte l'ensemble des commandes/attributs permettant de contrôler la ControlBox, et qui ne sont pas de la responsabilité de GalilAxis.

2 Structure matérielle et logicielle de l'application

2.1 Architecture matérielle



2.2 Architecture logicielle



- Le device server ControlBox embarque les devices suivants :
 - ControlBox pour les commandes du domaine de la CB toute entière : version/révision, RESET, informations sur les erreurs de communication...
 - GalilAxis permet un contrôle individuel d'un moteur
 - GalilSlit permet la gestion d'une fente à 2 volets pilotés par 2 moteurs indépendants
 - AxisRawDataReader permet de voir les valeurs codeur, moteur ; position commandées en unités hardware, pour des finalités de maintenance

3 Que sait faire ControlBox ?

- ControlBox est un device de bas niveau permettant de faire du diagnostic sur la communication, de connaître les révisions firmware/micro-code, passer des commandes expertes en jargon Galil.

4 Interface logicielle Tango

4.1 Properties

Les propriétés contiennent les paramètres fixes de l'axe. Ces paramètres sont fixés par le « motoriste » à l'installation de l'axe, ils ne sont pas supposés être modifiés en exploitation normale.

Les propriétés sont créées automatiquement au démarrage du device, mais leurs valeurs doivent être saisies par l'utilisateur

IPAddress	IP Address of the controller pas de valeur par défaut. Seule property créée par défaut.
Port	Port du controleur. Par défaut : 5500. N'est pas une property critique. La property n'est pas créée par défaut.
TCPTimeOut	valeur de timeout en secondes de la connection TCP. La property n'est pas créée par défaut.
DataRecordPollingPeriod	période de polling du Data Record. En millisecondes. 500 ms par défaut. mini admissible : 50 ; valeur recommandée : 200, acceptée : 100 La property n'est pas créée par défaut.

4.2 Les attributs

oKCommandCounter	ReadOnly. Compteur des commandes Galil passées avec succès. toutes les commandes/interrogations du contrôleur sont comptées.
badCommandCounter	compteurs des erreurs de communication avec le contrôleur.

4.3 Les commandes

Reset	commande de RESET de la ControlBox. Attention : <ul style="list-style-type: none"> • Les valeurs de positionnement, les positions initiales sont perdues. • Les micro-codes spécifiques sont arrêtés . • A n'utiliser uniquement qu'en cas de plantage de la CB.
StartMicrocode	Redémarre le micro-code générique.
State	Retourne l'état de la CB <ul style="list-style-type: none"> • CLOSE : pas fermée • OPEN : communication ouverte • ALARM : Deconnection sur erreur de communication • FAULT : erreur de communication sur la socket partagée • UNKNOWN : erreur non listée.
Status	Etat du device/description de l'erreur
GetFirmwareRevision	affiche la révision du firmware du contrôleur
GetMicrocodeRevision	affiche les indices de révision majeur/mineur du micro-code
ExecLowLevelCmd	permet l'exécution/l'affichage de commandes Galil en syntaxe Galil. Réservé aux véritables amateurs.

5.1 Configuration device ControlBox

- Inscrire le device dans la Database
- configurer l'adresse IP.
- si besoin changer le DataRecordPollingPeriod.
- Pinger la CB. Si la CB répond, démarrer le device. On devrait voir immédiatement le oKCommandCounter incrémenter => tout est OK
- exécuter la commande ExecLowLevelCmd, saisir : TH (Tell Handle), on devrait voir en réponse le 2 handles que le device a ouvert sur la CB