**Bibliothèque CDS5500 (En cours de rédaction)**

Cible : Servomoteur AX12+ Langage : C++ (Arduino) IDE : Visual studio 2013

Carte : Arduino Méga2560, Arduino Méga ADK

**Caractéristiques et documentation du servomoteur**

**A compléter**

**Exemples**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom** | **IDE** | **Description** | **Détails** |
| RHEX2016V2 | Visual Studio 2016 +  Visual Micro | Commande de deux servomoteurs (position et rotation permanente) dans les deux sens. |  |
| Test10Serial | Commande d’un moteur (rotation permanente).  Changement de la vitesse de rotation en fonction de la position angulaire.  Bouton Marche / Arrêt  Transmission de la position avec la liaison série 2 pour sauvegarde dans un fichier .txt via le terminal de CVAVR. | Annexe 1 |
| Test10SerialV2a  Et b | Commande de deux moteurs (rotation permanente)  Changement de la vitesse de rotation en fonction de la position angulaire.  Bouton Marche / Arrêt  Resynchronisation lors de chaque changement de vitesse.  (V2a : solution avec des booléens, V2b : solution avec des compteurs) | Annexe1a |
| Test10SerialV3 | Commande de trois moteurs (rotation permanente)  Changement de la vitesse de rotation en fonction de la position angulaire.  2 Boutons : BP2 Marche / BP3 Arrêt (Clavier Digilent)  Resynchronisation lors de chaque changement de vitesse. | Annexe1b |
| Test11Serial | Lecture de la table de contrôle d’un servomoteur et affichage dans le moniteur série Arduino. | Annexe 2 |

**Constantes**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Logiciel** | | **Table de contrôle (Matériel)**  **(EEPROM en gras, sinon RAM)** | | | |
| **Désignation** | **Nom de la constante** | **Add** | **Item** | **Access** | **Initial Value** |
| **Modèle de servomoteur** | CTRAdd\_MODEL\_NUMBER\_L | **0(0x00)** | Model Number(L) | RD | 12(0x0C) |
| CTRAdd\_MODEL\_NUMBER\_H | **1(0x01)** | Model Number(H) | RD | 0(0x00) |
| **Version du micro logiciel** | CTRAdd\_VERSION | **2(0x02)** | Version of firmware | RD | ? |
| **Identifiant** | CTRAdd\_ID | **3(0x03)** | ID | RD,WR | 1(0x01) |
| **Débit binaire** | CTRAdd\_BAUD\_RATE | **4(0x04)** | Baud Rate | RD,WR | 1(0x01) |
| **Temporisation avant transmission de la réponse** | CTRAdd\_RETURN\_DELAY\_TIME | **5(0x05)** | Return Delay Time | RD,WR | 250(0xFA) |
| **Valeur minimum de l’angle à atteindre** | CTRAdd\_CW\_ANGLE\_LIMIT\_L | **6(0x06)** | CW Angle Limit(L) | RD,WR | 0(0x00) |
| CTRAdd\_CW\_ANGLE\_LIMIT\_H | **7(0x07)** | CW Angle Limit(H) | RD,WR | 0(0x00) |
| **Valeur maximum de l’angle à atteindre** | CTRAdd\_CCW\_ANGLE\_LIMIT\_L | **8(0x08)** | CCW Angle Limit(L) | RD,WR | 255(0xFF) |
| CTRAdd\_CCW\_ANGLE\_LIMIT\_H | **9(0x09)** | CCW Angle Limit(H) | RD,WR | 3(0x03) |
|  |  | **10(0x0A)** | Reserved | - | 0(0x00) |
| **Température max** | CTRAdd\_LIMIT\_TEMPERATURE | **11(0x0B)** | The Highest Limit Temperature | RD,WR | 85(0x55) |
| **Tension min** | CTRAdd\_DOWN\_LIMIT\_VOLTAGE | **12(0x0C)** | The Lowest Limit Voltage | RD,WR | 60(0x3C) |
| **Tension max** | CTRAdd\_UP\_LIMIT\_VOLTAGE | **13(0x0D)** | The Highest Limit Voltage | RD,WR | 190(0xBE) |
| **Couple max** | CTRAdd\_MAX\_TORQUE\_L | **14(0x0E)** | Max Torque(L) | RD,WR | 255(0xFF) |
| CTRAdd\_MAX\_TORQUE\_H | **15(0x0F)** | Max Torque(H) | RD,WR | 3(0x03) |
| **Condition de renvoie d’un Status Packet** | CTRAdd\_STATUS\_RETURN\_LEVEL | **16(0x10)** | Status Return Level | RD,WR | 2(0x02) |
| **Led clignote si erreur** | CTRAdd\_ALARM\_LED | **17(0x11)** | Alarm Led | RD,WR | 4(0x04) |
| **Couple moteur désactivé si erreur** | CTRAdd\_ALARM\_SHUTDOWN | **18(0x12)** | Alarm Shutdown | RD,WR | 4(0x04) |
|  | CTRAdd\_OPERATING\_MODE | **19(0x13)** | Reserved | RD,WR | 0(0x00) |
|  | CTRAdd\_DOWN\_CALIBRATION\_L | **20(0x14)** | Down Calibration(L) | RD | ? |
| CTRAdd\_DOWN\_CALIBRATION\_H | **21(0x15)** | Down Calibration(H) | RD | ? |
|  | CTRAdd\_UP\_CALIBRATION\_L | **22(0x16)** | UP Calibration(L) | RD | ? |
| CTRAdd\_UP\_CALIBRATION\_H | **23(0x17)** | UP Calibration(H) | RD | ? |
|  | CTRAdd\_TORQUE\_ENABLE | 24(0x18) | Torque Enable | RD,WR | 0(0x00) |
| Etat de la led | CTRAdd\_LED | 25(0x19) | LED | RD,WR | 0(0x00) |
|  | CTRAdd\_CW\_COMPLIANCE\_MARGIN | 26(0x1A) | CW Compliance Margin | RD,WR | 0(0x00) |
|  | CTRAdd\_CCW\_COMPLIANCE\_MARGIN | 27(0x1B) | CCW Compliance Margin | RD,WR | 0(0x00) |
|  | CTRAdd\_CW\_COMPLIANCE\_SLOPE | 28(0x1C) | CW Compliance Slope | RD,WR | 32(0x20) |
|  | CTRAdd\_CCW\_COMPLIANCE\_SLOPE | 29(0x1D) | CCW Compliance Slope | RD,WR | 32(0x20) |
| Position à atteindre | CTRAdd\_GOAL\_POSITION\_L | 30(0x1E) | Goal Position(L) | RD,WR | [Adress36]val |
| CTRAdd\_GOAL\_POSITION\_H | 31(0x1F) | Goal Position(H) | RD,WR | [Adress37]val |
| Vitesse souhaitée | CTRAdd\_MOVING\_SPEED\_L | 32(0x20) | Moving Speed(L) | RD,WR | 0(0x00) |
| CTRAdd\_MOVING\_SPEED\_H | 33(0x21) | Moving Speed(H) | RD,WR | 0(0x00) |
| Couple max | CTRAdd\_TORQUE\_LIMIT\_L | 34(0x22) | Torque Limit(L) | RD,WR | [Adress14]val |
| CTRAdd\_TORQUE\_LIMIT\_H | 35(0x23) | Torque Limit(H) | RD,WR | [Adress15]val |
| Position angulaire | CTRAdd\_PRESENT\_POSITION\_L | 36(0x24) | Present Position (L) | RD | ? |
| CTRAdd\_PRESENT\_POSITION\_H | 37(0x25) | Present Position (H) | RD | ? |
| Vitesse | CTRAdd\_PRESENT\_SPEED\_L | 38(0x26) | Present Speed(L) | RD | ? |
| CTRAdd\_PRESENT\_SPEED\_H | 39(0x27) | Present Speed(H) | RD | ? |
| Charge | CTRAdd\_PRESENT\_LOAD\_L | 40(0x28) | Present Load(L) | RD | ? |
| CTRAdd\_PRESENT\_LOAD\_H | 41(0x29) | Present Load(H) | RD | ? |
| Tension | CTRAdd\_PRESENT\_VOLTAGE | 42(0x2A) | Present Voltage | RD | ? |
| Température | CTRAdd\_PRESENT\_TEMPERATURE | 43(0x2B) | Present Temperature | RD | ? |
| Ecriture terminée | CTRAdd\_REGISTERED\_INSTRUCTION | 44(0x2C) | Registered Instruction | RD,WR | 0(0x00) |
|  |  | 45(0x2D) | Reserved | - | 0(0x00) |
| Servomoteur entraîné par la charge | CTRAdd\_MOVING | 46(0x2E) | Moving | RD | 0(0x00) |
| Verrouillage de la table | CTRAdd\_LOCK | 47(0x2F) | Lock | RD,WR | 0(0x00) |
| Courant minimum pendant l’opération | CTRAdd\_PUNCH\_L | 48(0x30) | Punch(L) | RD,WR | 32(0x20) |
| CTRAdd\_PUNCH\_H | 49(0x31) | Punch(H) | RD,WR | 0(0x00) |

**Constructeur**

|  |  |  |
| --- | --- | --- |
|  | **Syntaxe** | **Description** |
|  | CDS5500(byte ID); | Construit un objet CDS5500 avec le numéro ID.  **ID** : Identifiant du servomoteur (0 à 253). |

**Méthodes publiques**

|  |  |  |
| --- | --- | --- |
|  | **Syntaxe** | **Description** |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | void DisableEndlessTurn(); | Activation du mode "commande en position" avec les valeurs par défaut. |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | void EnableEndlessTurn(); | Activation du mode "rotation continue". |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | unsigned int GetDataInStatusPacket (byte numberOfBytes); | Renvoie la donnée contenue dans un *Status Packet* sous la forme d’un entier positif codé sur 16 bits..  **numberOfBytes** : nombre d’octets à extraire (1 ou 2)  Remarques: Cette méthode doit être précédée de *WriteInstructionPacketforReading.*  si numberOfBytes = 2 la méthode retourne LowBytes + 255\*HigthBytes |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | unsigned int GetPosition(); | Renvoie la position angulaire du servomoteur sous la forme d’un entier positif codé sur 16 bits.  Celui-ci est :  - soit représentatif de la **position angulaire** du servomoteur (lue dans la table de contrôle).  - Entre 0° et 300°  0(0x000) ≤ position ≤ 1023(0x03FF)  - Entre 301° et 359° : position = 0  - soit représentatif d’une **erreur de transmission :**  **- En-tête erronée** = 4094,  **- Pas de réception** = 4095 (Rx\_time > 400µs)  Conditions de fonctionnement  Baud Rate = 1Mb/s (par défaut) Return Delay Time : par défaut |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | boolean Ping(); | Renvoie true si le servomoteur est présent dans le réseau, false sinon. |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | void Reset(); | Initialise la table de contrôle du servomoteur avec les valeurs par défaut. Le nouvel ID est égal à 1. |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | void SetDataInControlTable(byte CTRegisterAdd, byte valeur); | Ecriture de l'octet valeur, à l'adresse CTRegisterAdd, dans la table de contrôle du servomoteur.  **CTRegisterAdd**: Adresse du registre dans la table de contrôle.  Peut être remplacée par son nom. (Commence par CTR)  **valeur** : octet à placer dans la table. |
| void SetDataInControlTable(byte CTRegisterAdd, byte valeur1, byte valeur2); |
| void SetDataInControlTable(byte CTRegisterAdd, byte valeur1, byte valeur2, byte valeur3, byte valeur4); |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | Void SetInstructionPacketforReading (byte CTRegisterAdd, byte numberOfBytes); | Envoie une trame pour lire une valeur, codée sur 1 ou deux octets, dans la table de contrôle du servomoteur.  **CTRegisterAdd**: Adresse du registre dans la table de contrôle.  Peut être remplacée par son nom. (Commence par CTR)  numberOfBytes : un ou deux octets |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | void SetLed(boolean action); | Comande de la led du servomoteur.  Eclairée si action = true. Eteinte si action = false ; |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | void SetPosition(unsigned int position, unsigned int velocity); | Déplace l'axe du servomoteur de sa position actuelle à la position spécifiée par *position* avec la vitesse *velocity.*  **position**: position à atteindre. 0°(0x00) à 300°(0x3ff)  **velocity:** vitesse angulaire. 14RPM(0x3ff)  Remarque : La vitesse angulaire maximum dépend de la tension d’alimentation. velocity = 0 permet de régler la vitesse angulaire à sa valeur maximum pour la tension appliquée. (pas de contrôle de la vitesse) |
| C:\Users\phili\AppData\Local\Microsoft\Windows\INetCache\Content.Word\general_method_VS.PNG | void Speed(unsigned int velocityWithDirection); | Règle le sens et la fréquence de rotation du servomoteur.  **velocityWithDirection:** sens et vitesse. Vitesse maxi = 114RPM  - sens AH si 0(0x0000) ≤ velocityWithDirection≤ 1023(0x3FF).  - sens H si 1024(0x400) ≤ velocityWithDirection≤ 2045(0x7FF) |

**Annexe 1 : Test10Serial**

**Matériel**

- Carte Arduino 2560

- Carte Tinkerkit Mega

- Interface Rb-Dfr-142

- Alimentation continue 9V 1A

- Servomoteur Dynamixel AX12+

- Clavier 4 touches Digilent Pmod Button Header

- ModuleUSB/RS232 Digilent PMODRS232

- Adaptateur USB/RS232

**Logiciel**

|  |  |
| --- | --- |
| **Algorithme Prog\_princ**  // variables  position : entier positif codé sur 16 bits initialisé à 0  début  lire(position)  si (position ≤ 1023) alors début  Exécuter Graphe d’état  Transmettre la position  sinon rien  fin si  Attendre(n) // Règle la période d’échantillonnage n en millisecondes  fin | **Algorithme SP\_Interruption**  // variables  EtatID18 : booléen initialisé à faux  début  si (EtatID18 = faux)  alors début  activer servomoteur  EtatID18 ← vrai  fin  sinon début  désactiver servomoteur  EtatID18 ← faux  fin  finsi  fin |

Note : le sous-programme d’interruption est exécuté sur le ↑ de BTN3 (bouton-poussoir du clavier 4 touches)

**Graphe d’états**

146°< angle <149° with vitesse servo18 ← 1FFh

14,6°< angle <17,5° with vitesse servo18 ← 1FFh

234°< angle <237° with vitesse servo18 ← 9Fh

87,9°< angle <90,8° with vitesse servo18 ← 9Fh

↑BTN3 with

vitesse servo18 ← 9Fh

**Annexe 1a : Test10SerialV2a et b (Commande de deux moteurs avec resynchronisation sur chaque changement de vitesse)**

**α1**

temps

**Moteur1**

Seuil\_GV1

Seuil\_PV1

Dz

α1max

**α2**

temps

**Moteur2**

Seuil\_GV2

Seuil\_PV2

Dz

α2max

**Contraintes à respecter**

1 – Régler les moteurs en mode « rotation continue » (CW=CCW=0).

2 - Positionner mécaniquement les moteurs pour avoir les seuils hors de la DZ.

3 - Identifier les valeurs des seuils.

3 – Adapter le sens de rotation.

**αx**: angle du moteur x PVx: petite vitesse du moteur x GVx : grande vitesse du moteur x Dz : pas de lecture de la position

si (Seuil\_PV1)

avec M1 <- Arrêt

synchro1 <- 1

si (Seuil\_GV1)

avec M1 <- Arrêt

synchro1 <- 1

**GF\_M1**

si (Synchro2) > 0

avec M1 <- GV1

Synchro2 <- Synchro2 - 1

si (Synchro2)>0

avec M1 <- PV1

Synchro2 <- Synchro2 - 1

M1 <- PV1

Traitement d’une transition et passage à l’autre graphe.

si (Seuil\_PV2)

avec M2 <- Arrêt

synchro2 <- 1

si (Seuil\_GV2)

avec M2 <- Arrêt

synchro2 <- 1

**GF\_M2**

si (Synchro1) > 0

avec M2 <- GV2

Synchro1 <- Synchro1 - 1

si (Synchro1)

avec M1 <- PV1

Synchro1<- Synchro1 - 1

M2 <- PV2

**Annexe 1b : Test10SerialV3 (Commande de trois moteurs avec resynchronisation sur chaque changement de vitesse)**

si (Seuil\_PV1)

avec M1 <- Arrêt

synchro1 <- 2

si (Seuil\_GV1)

avec M1 <- Arrêt

synchro1 <- 2

**GF\_M1**

si (Synchro2)> 0 et synchro3 >0)

avec M1 <- GV1

Synchro2 <- Synchro2 – 1

Synchro3 <- Synchro3 - 1

si (Synchro2>0 et synchro3>0)

avec M1 <- PV1

Synchro2 <- Synchro2 - 1

Synchro3 <- Synchro3 - 1

M1 <- PV1

si (Seuil\_PV2)

avec M2 <- Arrêt

synchro2 <- 2

si (Seuil\_GV2)

avec M2 <- Arrêt

synchro2 <- 2

**GF\_M2**

si (Synchro1 > 0 et synchro3>0)

avec M2 <- GV2

Synchro1 <- Synchro1 - 1

Synchro3 <- Synchro3 - 1

si (Synchro1) et synchro3>0

avec M2 <- PV2

Synchro1<- Synchro1 – 1

Synchro3 <- Synchro3 - 1

M2 <- PV2

si (Seuil\_PV3)

avec M3 <- Arrêt

synchro3 <- 2

si (Seuil\_GV3)

avec M3<- Arrêt

synchro3 <- 2

**GF\_M3**

si (Synchro2>0 et synchro1>0)

avec M3 <- GV3

Synchro2 <- Synchro2 - 1

Synchro1 <- Synchro1 - 1

si (Synchro2>0 et synchro1>0)

avec M3<- PV3

Synchro2 <- Synchro2 - 1

Synchro1 <- Synchro1 - 1

M3 <- PV3

**α3**

temps

**Moteur3**

Seuil\_GV3

Seuil\_PV3

Dz

α3max

**Tableau des paramètres de réglage**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | M1 | M2 | M3 | M4 | M5 | M6 |
| ID |  |  |  |  |  |  |
| SeuilPV |  |  |  |  |  |  |
| SeuilGV |  |  |  |  |  |  |
| PV |  |  |  |  |  |  |
| GV |  |  |  |  |  |  |
| 100\*|PV – GV|/1023 |  |  |  |  |  |  |

**Annexe 2 : Exemple d’affichage du contenu d’une table de contrôle**

Lecture de l’EEPROM dans la table de contrôle



Lecture de la RAM dans la table de contrôle

