**RACK EPIC**

Le rack epic est constitué de deux arduinos, l’Arduino Méga qui s’occupe de l’acquisition des entrées analogiques et numériques et l’Arduino Léonardo s’occupe de la communication avec l’extérieur (PC) et des sorties numériques.

Les deux arduinos sont interconnectés en I2C.

Pour un bon fonctionnement, il est nécessaire d’installer les drivers arduinos.

**Arduino Mega**

Il lit en permanence les entrées analogiques et les entrées numériques.

Il se comporte comme un slave I2C vis-à-vis de l’Arduino Léonardo

Debouncing entrées numériques :

Il y a une fonction de debouncing sur les entrées numériques.

Les entrées numériques sont lues successivement 3 fois ; les états sont enregistrés dans un double tableau de 3 par 24 (de 8bit soit 24x8=192 entrées).

Un vote est fait entre les trois lectures pour déterminer l’etat dominant.

Pour chaque booléen ont incrémente une variable pour un état vrai et décrémente cette meme variable pour un état faux.

Si le résultat est positif alors les états vrais l’emportent et la valeur est enregistrée à vrai.

Si le résultat est négatif alors les états faux l’emportent et la valeur est enregistrée à faux.

Les valeurs sont stockées dans un tableau en attendant la demande de lecture de l’Arduino Léonardo.

Communication :

L’arduino Mega possède deux registres :

Le registre 1 correspond aux entrées numériques, il est composé de 24 bytes de 8 bit soit 192 bits pour 192 entrées.

Le registre 2 correspond aux entrées analogiques, il est composé de 32 bytes de 8bit car chaque entrée analogique fait 2 byte (chaque entrée analogique est sur 10 bit).

**Arduino Léonardo**

L’Arduino Léonardo communique en USB avec l’ordinateur.

Il créé un port COM et un périphérique de jeux sur l’ordinateur.

Le périphérique de jeux est constitué de 6 joysticks.

Chaque joystick a 32 entrées numériques.

Les 4 premiers joysticks ont chacun 4 entrées analogiques (X,Y,Z, RX).

Entrées analogiques et numériques

Cette Arduino envoie une requête au Méga pour lire ses registres ( 1 et 2) .

Les valeurs récupérées sont ensuite affectées aux joysticks.

Sorties numériques :

Elles sont pilotées en I2C, il y a deux composants MCP23017 équipées de 16 sorties chacun.

Pour contrôler les sorties numériques, il faut envoyer un message JSON sur le port COM.

Tous les messages doivent se terminer par « ! ».

Voici le type de message à envoyer :

{"cmd":"digOutput","data":[0,0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]}!

{"cmd":"digOutput","data":[0,0,0,1,0,0,0,0,0,0,0,0,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0]}!

Trois valeurs sont possibles :

0 🡪 la sortie s’éteint

1 🡪 la sortie s’allume

7 🡪 la sortie ne change pas d’état

Si la commande est prise en compte le message suivant est retourné :

{"cmd":"command\_return","data" : "OK"}

Sinon:

{"cmd":"command\_return","data" : "KO","description":"short description of problem"}

Echo mode:

On peut envoyer une commande pour avoir l’état des entrées numériques et analogiques sur le port COM.

Message à envoyer pour activer le mode :

{"cmd":"setEcho","data":true}!

Message pour désactiver le mode :

{"cmd":"setEcho","data":false}!

Le message reçu avec les entrées numériques et analogiques :

{"cmd": "stAnaIn","data":[235,467,432,678,789,1024,1000,800,765,485,456,123,234,345,456,789]}

{"cmd": "stDigIn","data":[1,0,1,0,1,0, … ,0,1,0]}