

2010

Droids Company

Manuel d'utilisation Mini Tourelle Caméra sans fil
référence MICA-SF-CCTV-HR-360-12-V-1012

[MANUEL D'UTILISATION MINI TOURELLE CAMERA SANS FIL]

V1.0

Pour télécharger la dernière version de ce manuel, rendez-vous sur
www.droids-company.com/manuel

Cher client,

Merci d'avoir choisi ce produit Droïds Company. Nous espérons que vous aurez plaisir à l'utiliser et qu'il vous donnera satisfaction pendant de nombreuses années. Le présent document en décrit les caractéristiques et les informations nécessaires pour le piloter.

L'équipe Droïds Company.

Toutes les informations contenues dans ce manuel étaient correctes au moment de sa publication. Toutefois, dans la mesure où nous améliorons et mettons fréquemment à jour nos produits, votre appareil peut présenter des différences mineures d'apparence ou de fonctionnalités par rapport aux indications figurant dans ce manuel.

Toute reproduction, même partielle, de ce document est interdite sans un accord préalable signé de la société Droïds-Company. Le code fourni ne peut être utilisé à des fins commerciales et reste l'entière propriété de Droïds-Company. Le produit ne peut être démonté sans en annuler sa garantie.

1. Table des matières

2.	Table des révisions	4
3.	Contenu du bundle.....	5
4.	Caractéristiques.....	7
5.	Synoptique général.....	7
6.	Unités	7
7.	Branchements	8
8.	Description des trames.....	10
9.	Exemple de code C#	11
10.	Assistance technique.....	12

2. Table des révisions

[20/01/2010]	Création du document
[05/02/2010]	Ajout Commande MOTO Ajout Schéma de connexion HE10

3. Contenu du bundle

- 1> Tourelle Motorisée avec caméra sans fil CCTV haute définition.



- 2> Récepteur Vidéo USB Autoalimenté avec CD d'installation, câble USB et analogique.



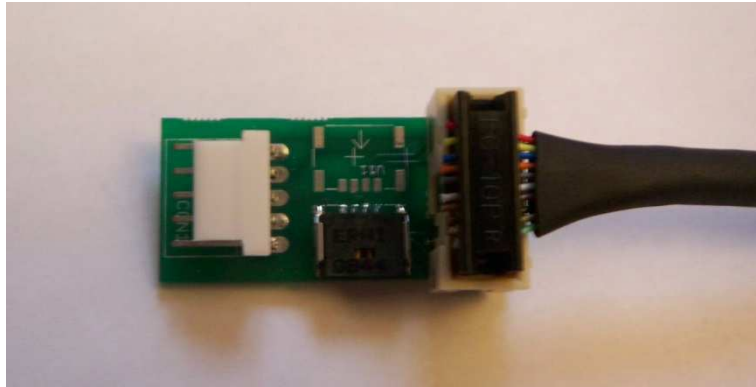
- 3> Récepteur Vidéo TV avec bloc d'alimentation et cordon de liaison analogique.



4> Platine NETBURNER et CD NBEclipse (optionnel)

Le code du processeur est disponible sur demande spécifique. Il requiert l'achat de la licence d'utilisation de l'environnement de développement pour MOD5213.

5> Adaptateur RS232/I2C (optionnel)



6> Adaptateur Série (optionnel)



7> CD d'installation

Manuel d'utilisation (manuel1012V1.pdf)

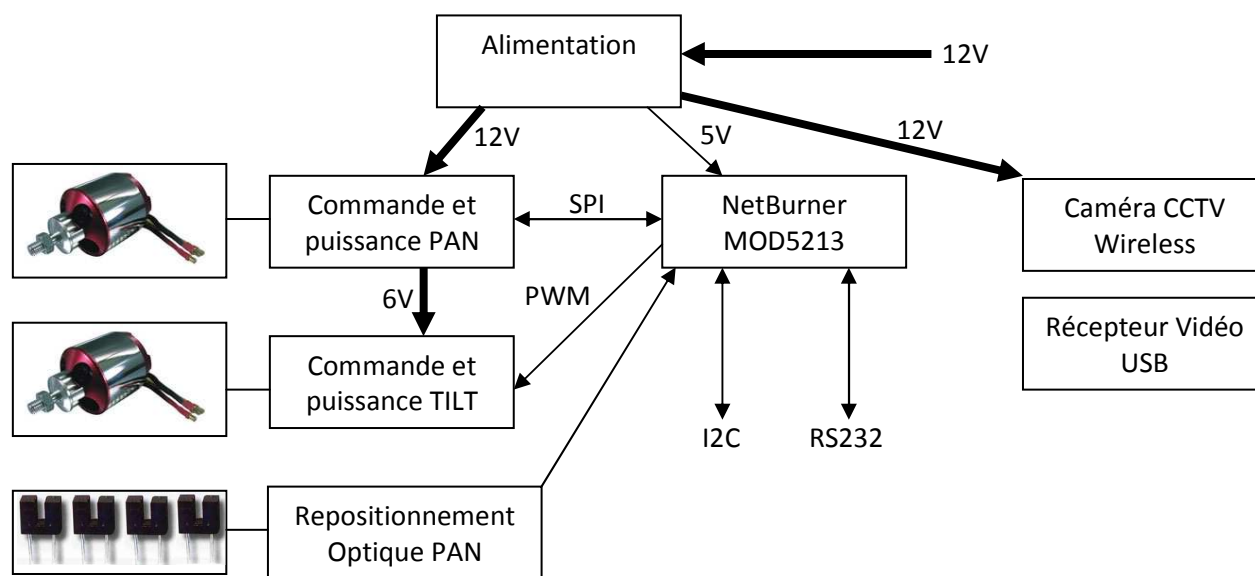
Projet C# (MiniTourelleV1)

Projet MOD5213 (optionnel)

4. Caractéristiques

Paramètres	valeur	observation
Alimentation	+12V	Maximum +14V
Angle Pan	360° infini	
Angle Tilt	+20° à -90°	
Vitesse Pan max	200°/s	
Vitesse Tilt Max	90°/s	
Processeur	Coldfire MCF5213	
Consommation	500mA hors charge	
Dimension	Diamètre : 136 mm Hauteur : 120 mm	Hors antenne Caméra horizontale
Niveaux I2C Vitesse	5V 100 Kbps	
Niveaux RS232 Vitesse Paramétrage	-12V/+12V 115200 bps 8 bits	Compatible PC

5. Synoptique général



6. Unités

Vitesse Pan : en 1/100^{ème} de degré (de -20000 à 20000)

Position Pan : en 1/100^{ème} de degré (de -18000 à 18000)

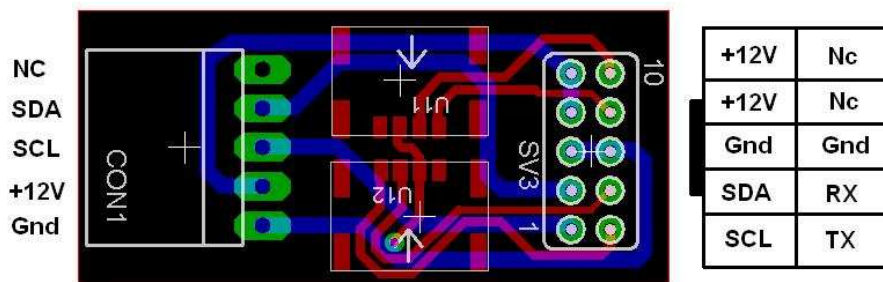
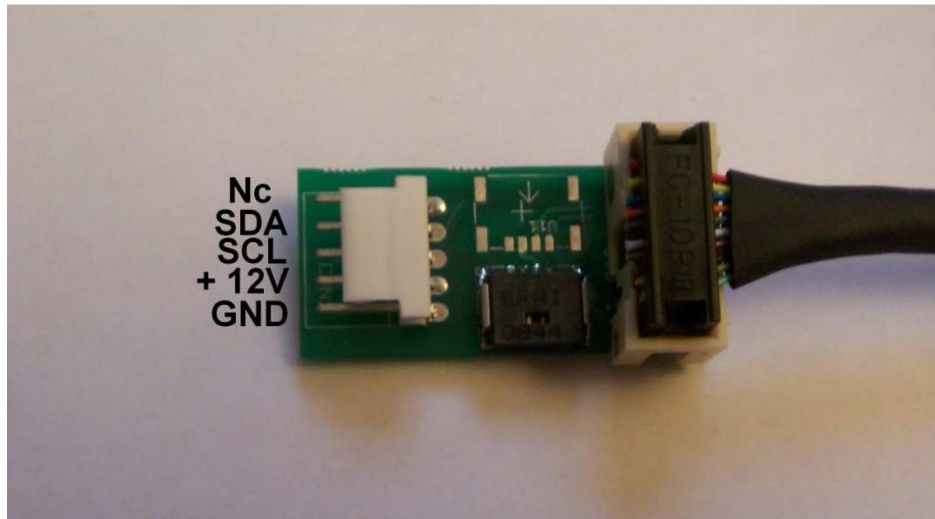
Vitesse Tilt : unité processeur (de 0 à 7000)

Position Tilt : en degré (de -90 à 20)

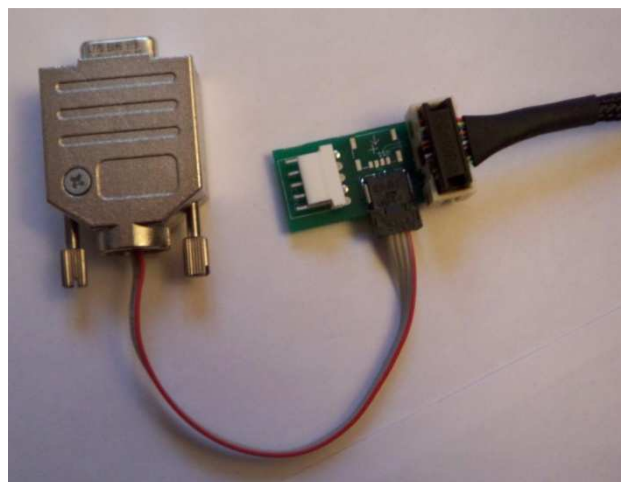
7. Branchements

Avant tout branchement, assurez-vous que la tourelle soit en position stable et qu'aucun objet ne puisse empêcher sa libre rotation. Assurez-vous de la bonne connexion de l'alimentation. Toute inversion de polarité risque d'endommager de manière irréversible le système.

En vous branchant sur le connecteur blanc, vous pouvez piloter cette mini-tourelle en mode I2C.



Pour le pilotage en RS232, il vous faut utiliser l'adaptateur fourni et le connecter de la manière suivante. Vous devez toujours connecter le +12V et le Gnd du connecteur blanc.



Pour visualiser l'image caméra, vous pouvez utiliser le récepteur analogique fourni et connecter sa sortie sur un téléviseur.

Vous pouvez aussi utiliser le récepteur autoalimenté USB que vous devez connecter sur un port USB libre de votre PC. Pour l'installation, il vous sera demandé d'insérer le CD d'installation fourni. Pour avoir accès à l'interface de visualisation et d'enregistrement, il vous faudra installer le logiciel EasyViewer fourni sur ce même CD.

8. Description des trames

Les commandes sont identiques en RS232 et en I2C hormis un entête supplémentaire de 3 octets en I2C.

- RS232 : <trame de commande....>
- I2C : <Adresse I2C> <'A'> <taille des données> <trame de commande>

Toute trame débute par le caractère '*' et se termine par le caractère ';' ;

Les paramètres sont donnés sous forme d'entier.

Exemples :

*PSPE 100 ; correct

*PSPE 10.45 ; incorrect

*PSPE 0x40 ; incorrect

<trame de commande>	paramètres	Description	I2C	RS232
*PSPE X;	X : vitesse Pan	Commande en vitesse du PAN	X	X
*TSPE X;	X : vitesse tilt	Commande en vitesse du TILT	X	X
*PPOE X Y ;	X : position Pan Y : vitesse Pan	Commande en position du PAN en spécifiant la vitesse de déplacement	X	X
*PPOE X Y ;	X : position Tilt Y : vitesse Tilt	Commande en position du TILT en spécifiant la vitesse de déplacement	X	X
*SUBS ;	-	Permet d'être notifié automatiquement des positions en PAN (La notification automatique n'est fonctionnelle qu'en RS232)		X
*UNSU ;	-	Annule la notification automatique	X	X
*ASPE X Y ;	X vitesse Pan Y vitesse Tilt	Commande en vitesse du PAN et du TILT	X	X
*SPAR X ;	X : adresse I2C	Spécifie l'adresse I2C du module, Nécessite un redémarrage pour être prise en compte	X	X
*GPAR ;		Renvoie le nom du module, sa version logicielle et son adresse I2C actuelle		X
*GEPT ;		Renvoie les positions en PAN et TILT	(1)	X
*MOTO X ;	X : 0 ou 1	0 : coupe l'alimentation des moteurs 1 : Alimentation des moteurs	X	X

(1)

La trame I2C équivalente est la suivante (3 octets) : <Adresse I2C> <'G'> <9>

La réponse est sur 9 octets :

- Octet 0 à 3 : position Pan (Entier 32 bits, Big Endian)
- Octet 4 à 7 : position Tilt (Entier 32 bits, Big Endian) format unité processeur
- Octet 8 : adresse I2C du module

9. Exemple de code C#

```
// port RS232 de communication
SerialPort _serialPort = null;
string _scom="COM6";

// ouverture du port de communication
private void button_Connect_Click(object sender, EventArgs e)
{
    try{
        _serialPort = new SerialPort(_scom);
        _serialPort.BaudRate = 115200;
        _serialPort.Open();
        _serialPort.ReadTimeout = 500;
        _serialPort.DataReceived += new
        SerialDataReceivedEventHandler(DataReceived);
    }
    catch (Exception ex)
    {
        Console.WriteLine("Unable to open "+_scom);
    }
}

private void SendVitesse(int vitesse)
{
    Send("*PSPE"+ vitesse +";");
}

public void Send(String chaine,bool i2c)
{
    try
    {
        _serialPort.Write(chaine);
    }
    catch (Exception ex)
    {
        Console.WriteLine("Please Open COM port !!!");
    }
}

void DataReceived(object sender, SerialDataReceivedEventArgs e)
{
    byte[] _buffanswer = new byte[100];
    int _bytes = 0;
    int index=0;
    char[] machaine=new char[100];

    _bytes = _serialPort.Read(_buffanswer, 0, 100);

    for (int i = 0; i < _bytes; i++)
    if (_buffanswer[i] != 13 && _buffanswer[i] != 10)
    {
        machaine[index]=Convert.ToChar(_buffanswer[i]);
        index++;
    }
    Console.WriteLine(machaine);
}
```

10. Assistance technique

Besoin d'aide ? Vous pouvez trouver les informations dont vous avez besoin :

En interrogeant notre équipe d'assistance technique : rendez-vous sur

www.droids-company.com/support pour une assistance par email.

Les images et illustrations peuvent ne pas toujours correspondre au contenu. Toutes les informations contenues dans ce manuel étaient exactes au moment de la publication.

COPYRIGHT Droids Company © 2010.