



**CAMERA LINEAIRE**

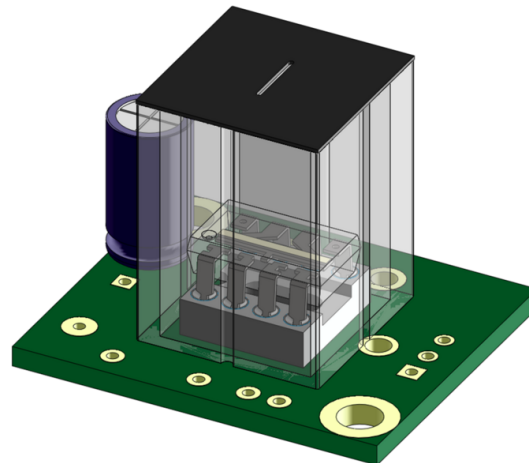
**BLUETOOTH**

**ENCODEURS**

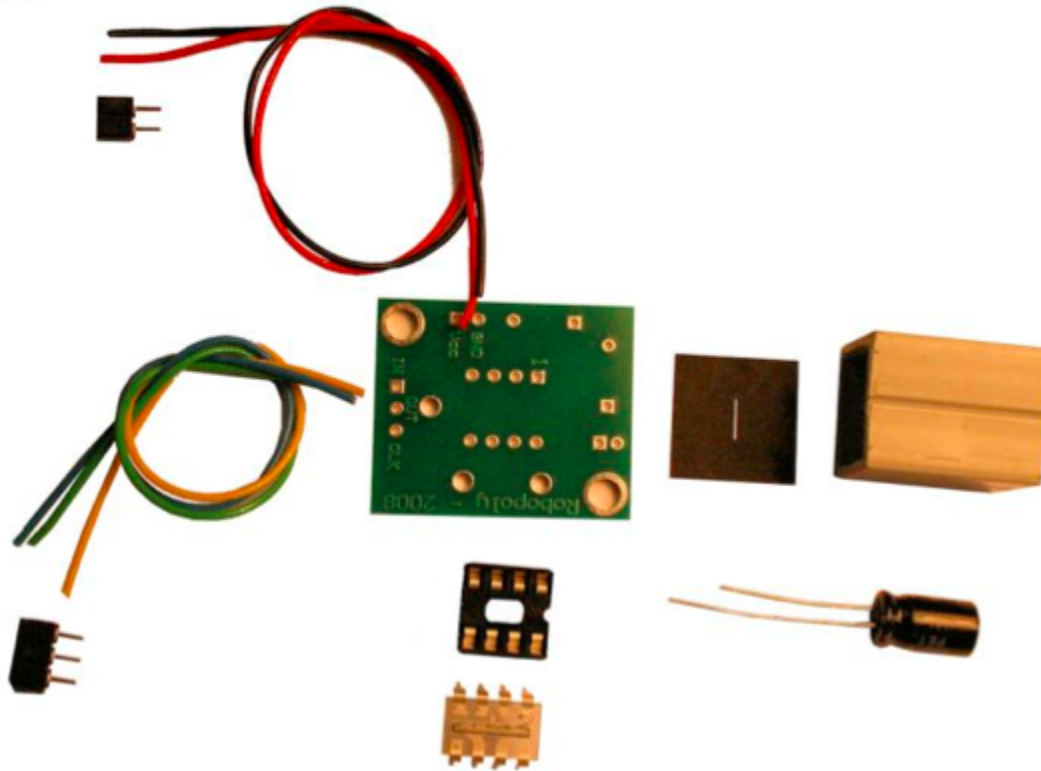
# La caméra linéaire



- Prend une image
- En noir et blanc
- En une dimension (une ligne de 102 pixels)
- Mesure l'intensité lumineuse

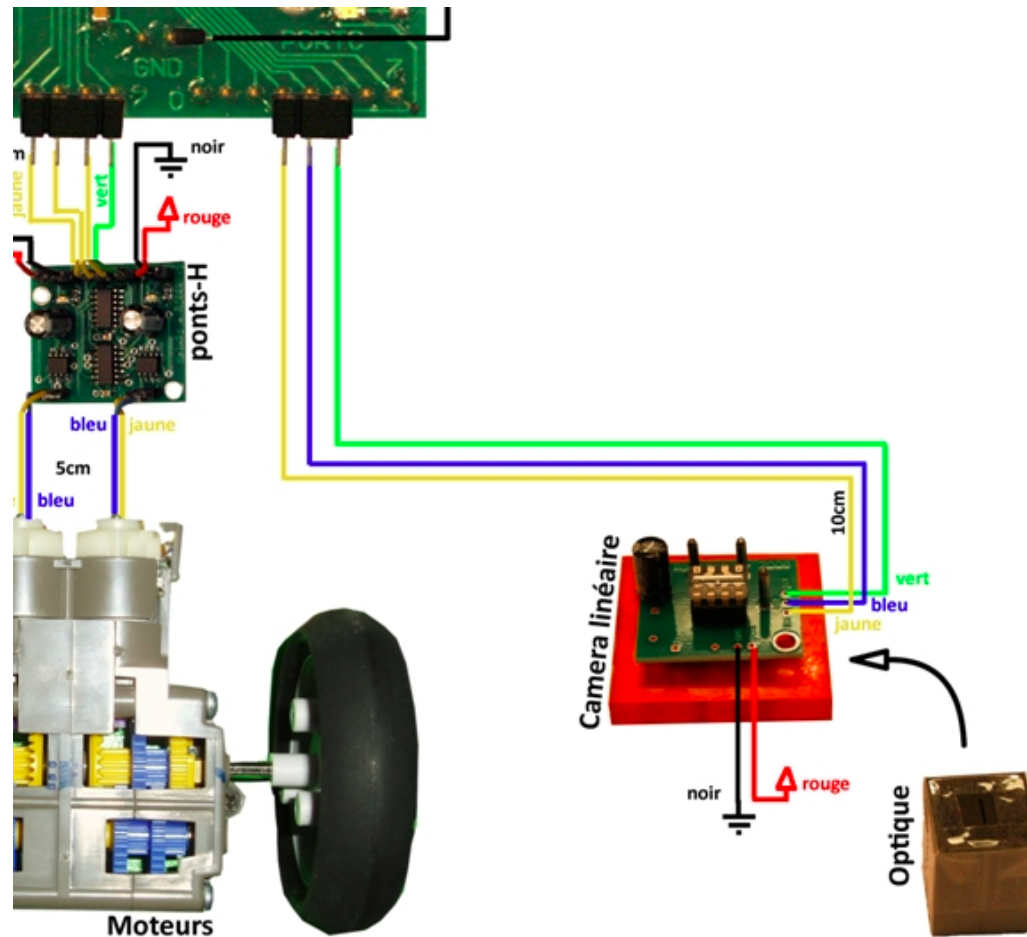


# Montage



[Guide de montage](#)

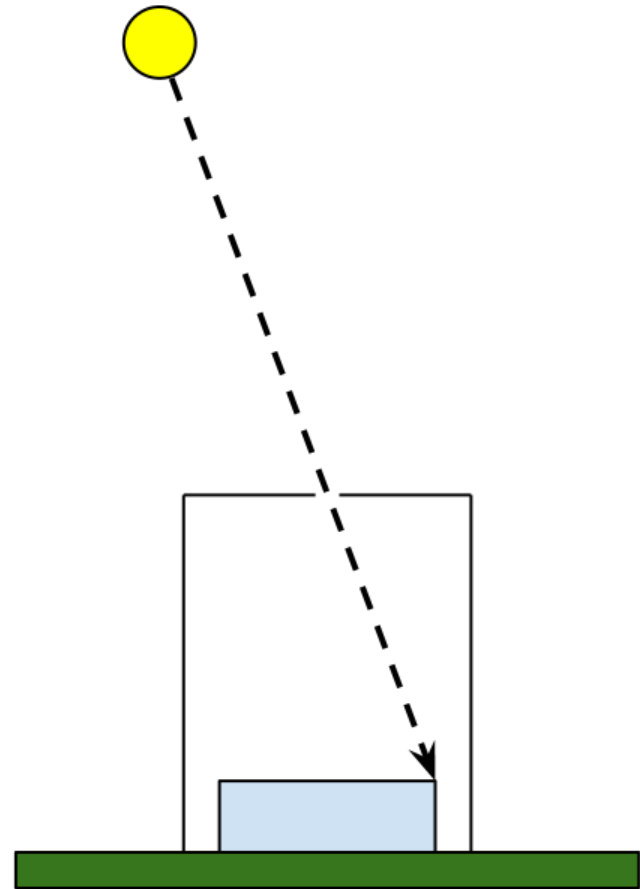
# Connectique



# Principe de fonctionnement



- La lumière arrive par la fente sur les 102 récepteurs
- Plus le temps d'exposition est long plus l'intensité moyenne sera haute
- Attention à la saturation



# Programmation



```
void lcam_setup(void);
```

```
void lcam_integrate(unsigned int microseconds);
```

```
// retourne le début de la liste des 102 pixels
```

```
unsigned char* lcam_getdata(void);
```

```
// fait une moyenne divisé en 25 zones et retourne celle  
qui est la plus intense
```

```
void lcam_read(void);
```

```
unsigned char lcam_getpic(void);
```

Documentation détaillé sur le site de Robopoly

# Exemple: robot héliophile

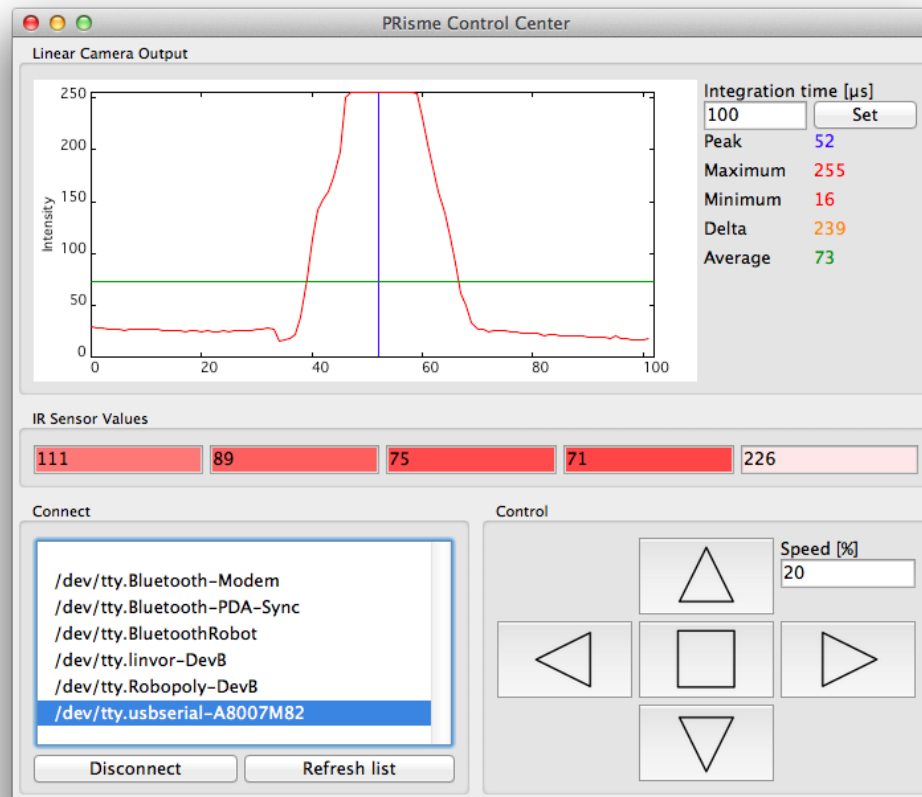


1. Trouver la lumière
2. S'aligner avec le pic de lumière
3. Aller vers la lumière

# Démonstration



## Programme qui lit les capteurs du PRisme





# Achat



- Prix caméra: CHF 5.-
- Guide de montage au local ou sur le site de Robopoly
- Kit PRisme: CHF 70.-
- 8 Piles rechargeables: CHF 15.-
- Autres éléments: demander



CAMERA LINEAIRE

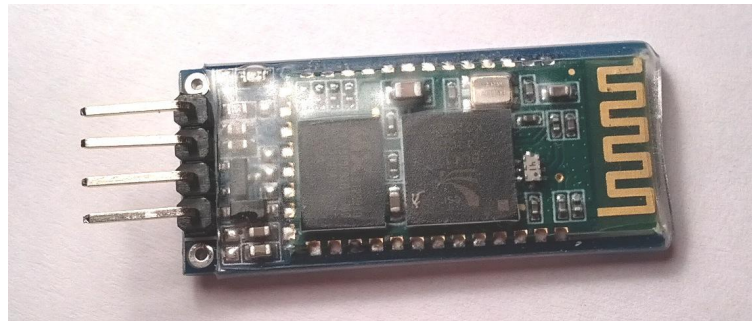
**BLUETOOTH**

ENCODEURS



# Communication sans fil

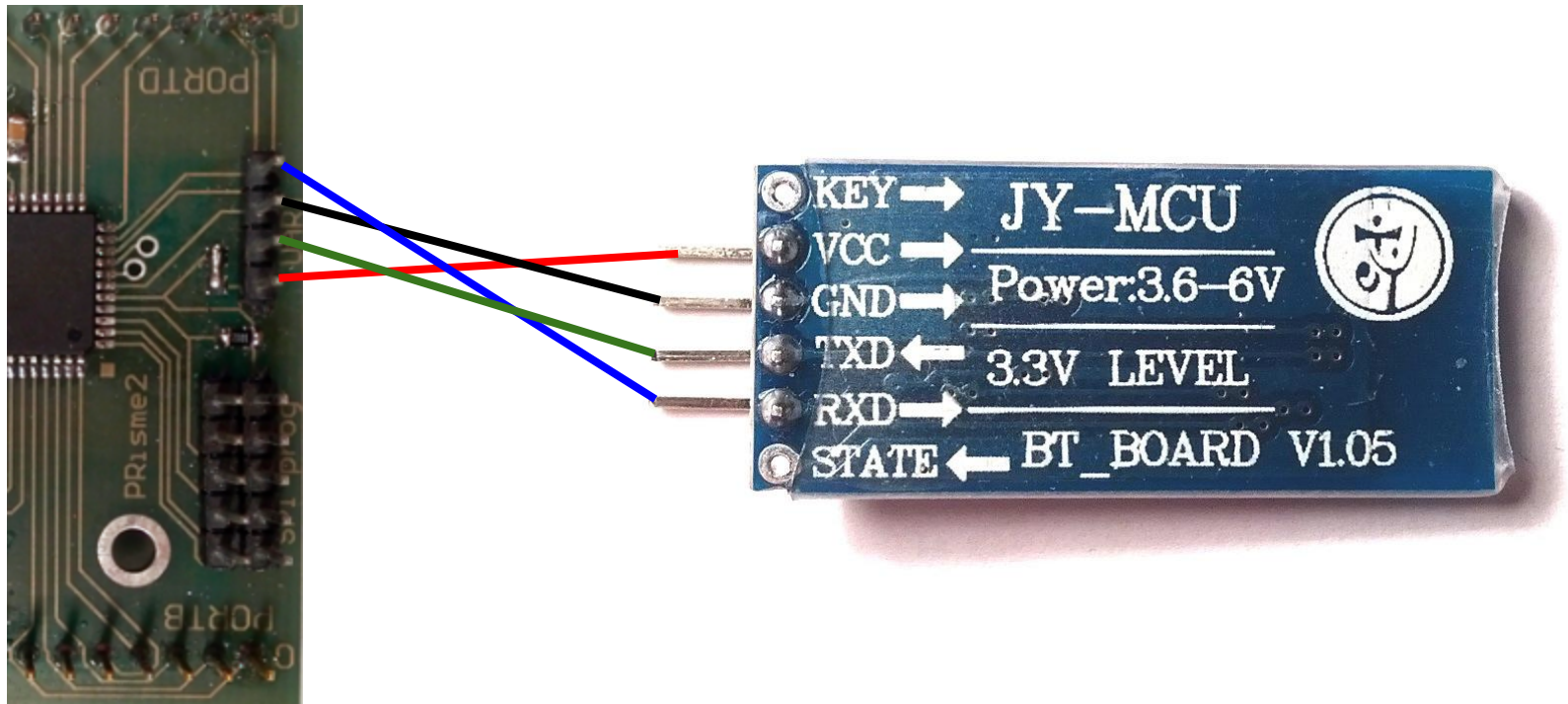
- Différents méthodes:
  - Bluetooth
  - XBee
  - Infra Rouge
  - ...
- Prix: \$8.20 sur [DealExtreme](#)



# Connectique



- Pas pin-compatible!
- Connecter le module correctement!





CAMERA LINEAIRE

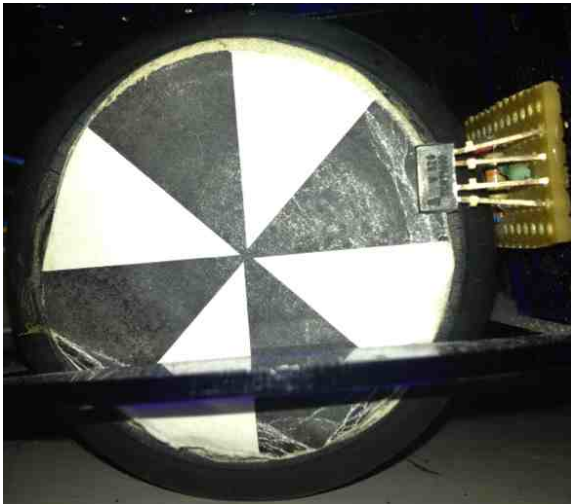
BLUETOOTH

**ENCODEURS**

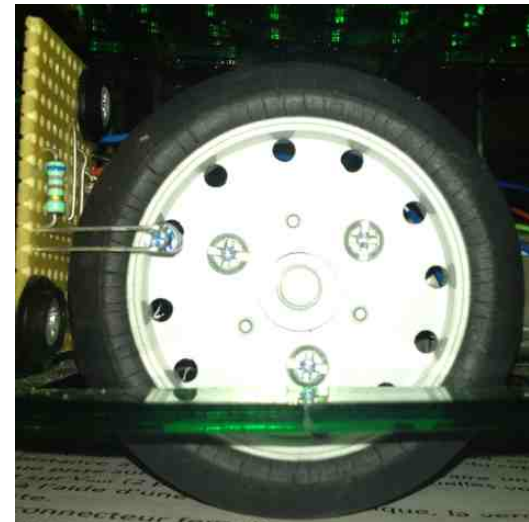


# Principe encodeurs

- Permet de connaitre le mouvement des roues: position, vitesse
- Avec un motif



ou des trous



- Lecture des increments? Interruptions!



# Connaître la position du robot



- Une intégration permet d'estimer la position

$$p = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}, \Delta s = \frac{\Delta s_d + \Delta s_g}{2}, \Delta \theta = \frac{\Delta s_d - \Delta s_g}{b} \quad \Delta s_d = R \cdot \phi_d$$

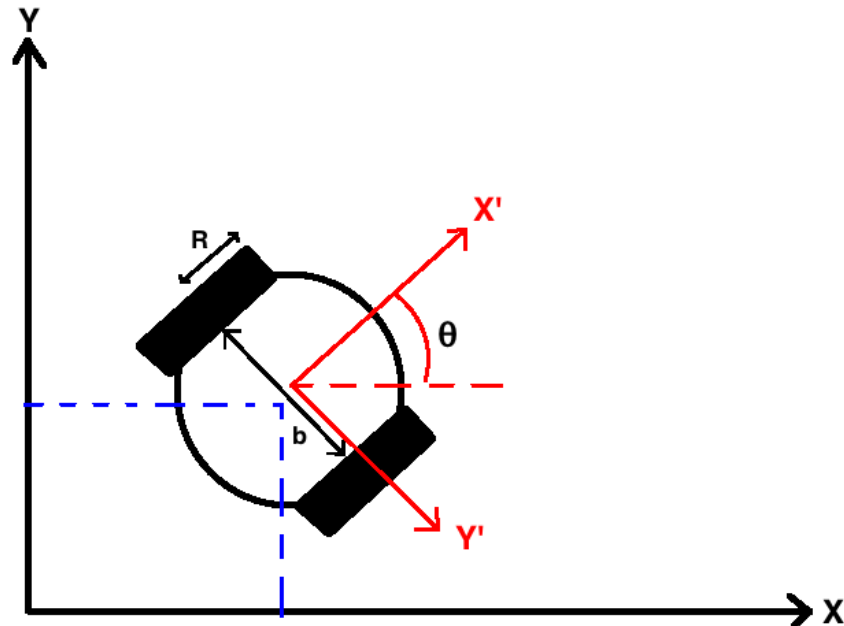
$p$  : position actuelle du robot

$\Delta s$  : avance du robot

$\Delta \theta$  : rotation du robot

$\phi$  : angle que la roue a tourné

- Sources d'erreurs?



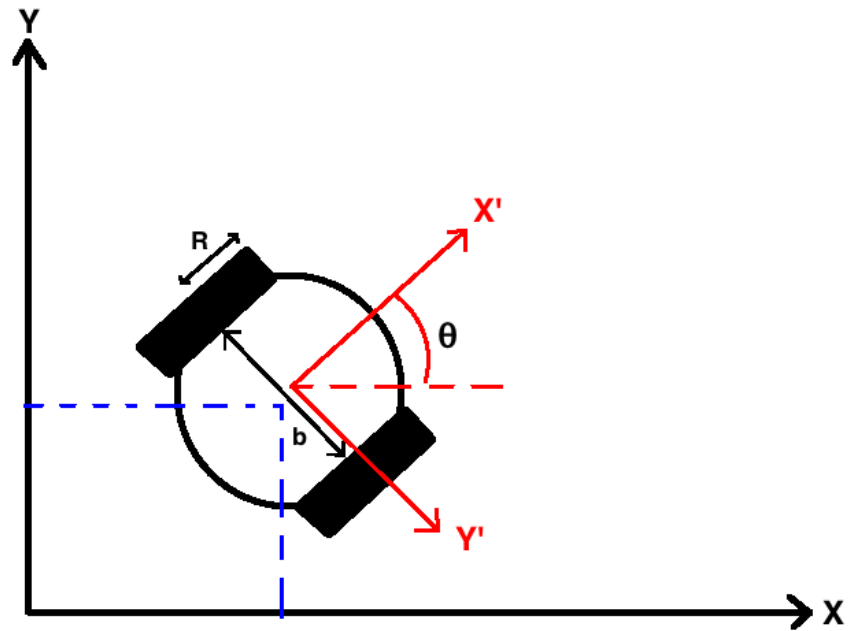


# Connaître la position du robot



- Une intégration permet d'estimer la position

$$p = p_{prec} + \begin{pmatrix} \Delta s \cdot \cos(\theta + \frac{\Delta\theta}{2}) \\ \Delta s \cdot \sin(\theta + \frac{\Delta\theta}{2}) \\ \Delta\theta \end{pmatrix}$$



- Sources d'erreurs?



## La suite

- Lundi prochain: invité
  - Alessandro Crespi, robot salamandre
- Samedi 9 Mars: Fireday Due!
- Samedi 20 Avril: Grand Concours



**FIN**

Questions?