MOBILE ROBOTICS

Eric dubreuil : eric.dubreuil@umontpellier.fr

POLYTECH
MONTPELLIER

MUSE
MONTPELLIER UNIVERSITY OF EXCELLENCE

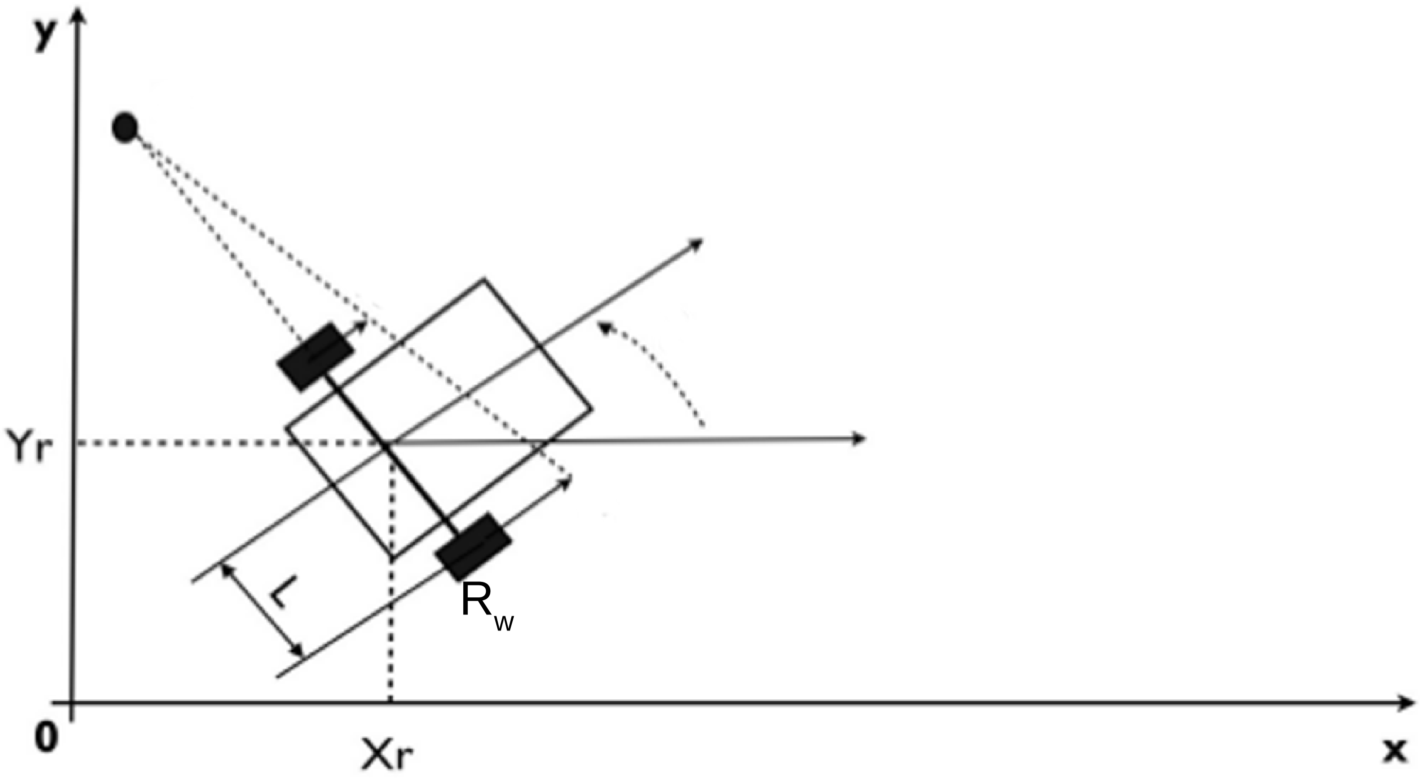
Goals of the day :

- Analyze three kinds of mobile robots :
    - Dual-wheel unicycle
    - Three-wheel omnidirectionnal
    - Four-wheel mecanum

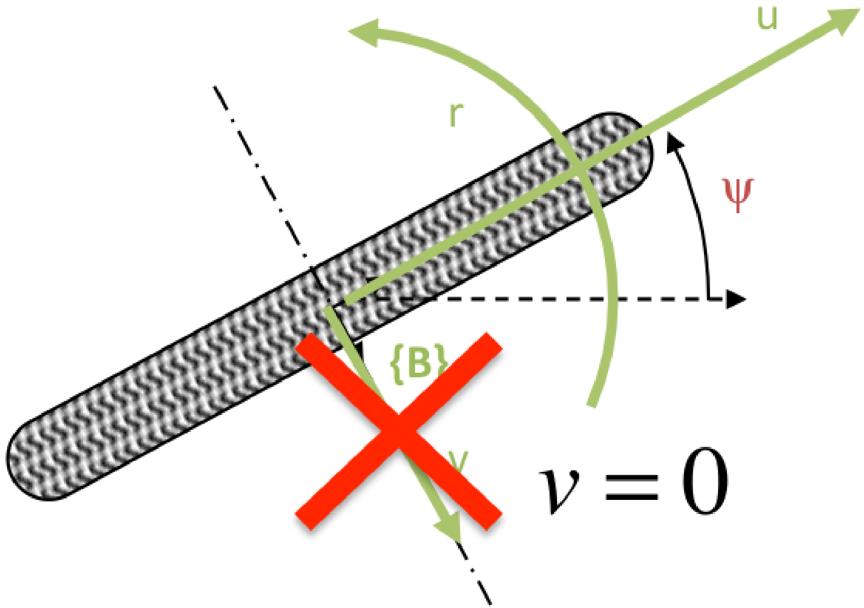
- Wheels
    - Classic
    - Swedish wheel
    - Mecanum wheel

- Control processes
    - Mathematical model
    - Vector control
    - Command programming

Dual-wheeled unicycle Robot

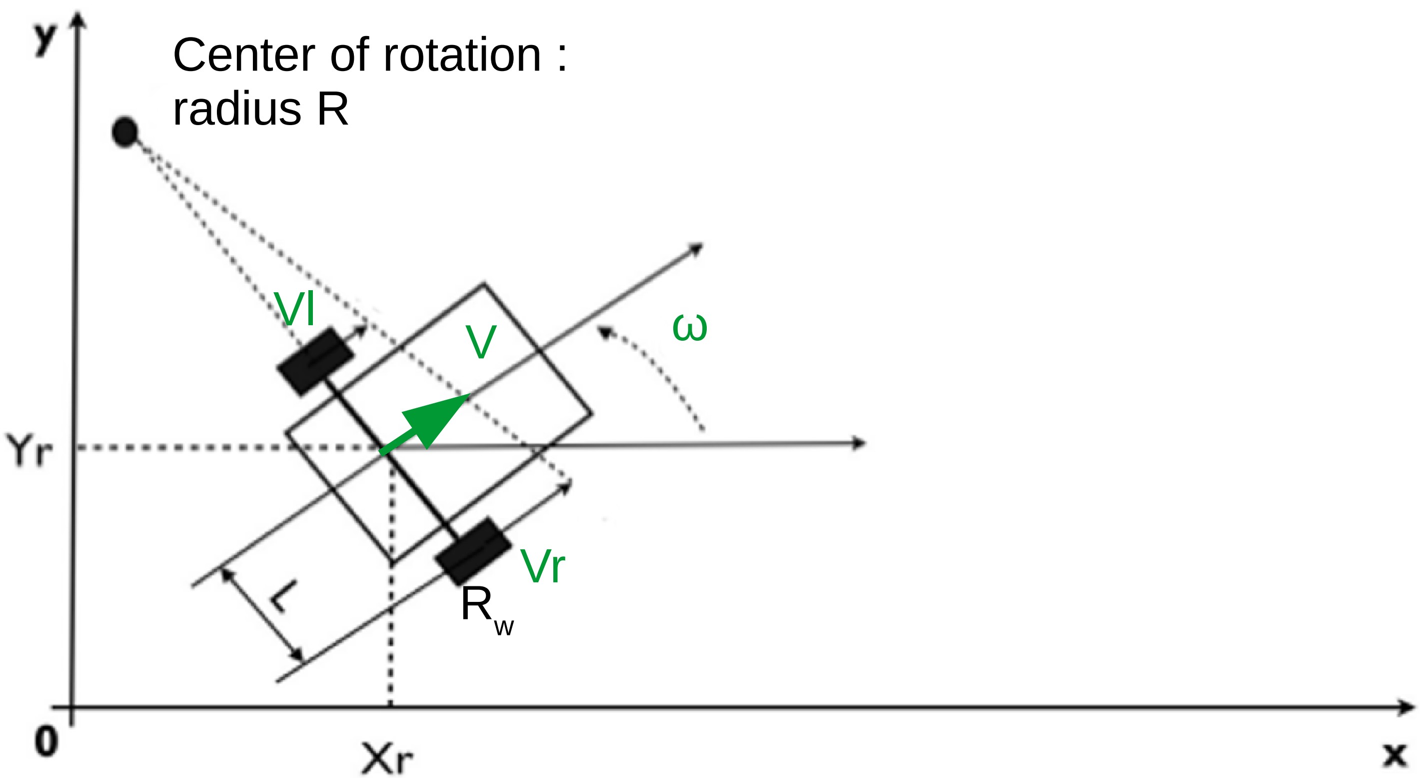


Classic wheel



- rotates on itself
- moves on u
- DOES NOT move on v

Dual-wheeled unicycle Robot



Using Thales's theorem

$$\frac{V_r}{R+L} = \frac{V_l}{R-L} = \frac{V}{R}$$

and

$$V = R.\omega$$

$$V_r = R_w.\omega_r \text{ and } V_l = R_w.\omega_l$$

Dual-wheeled unicycle Robot

Control parameters : $\vec{V}, R$

Control vectors : $\vec{V}, \vec{\omega}$

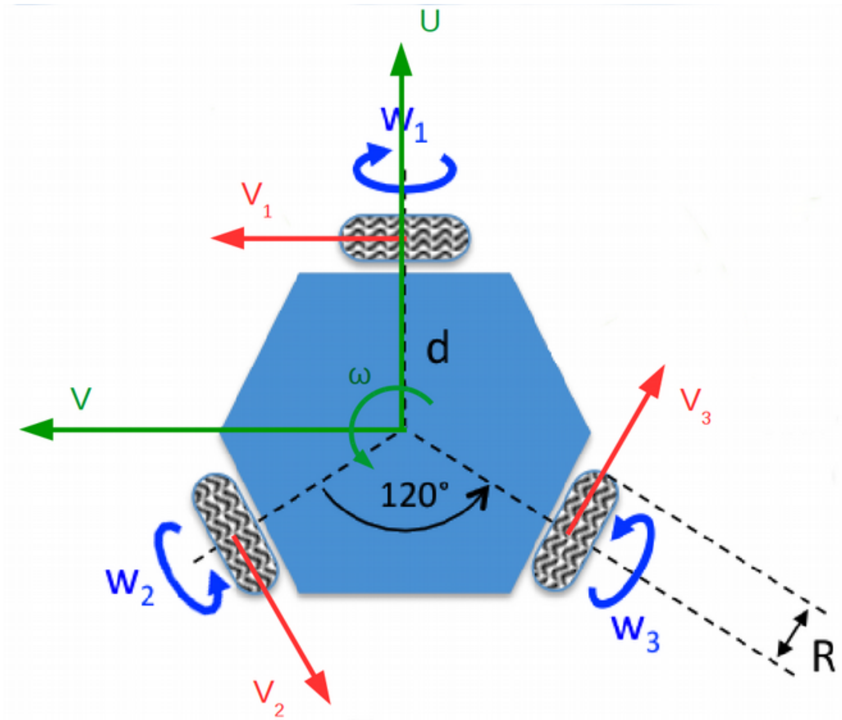
$$\omega_r = \frac{R+L}{R_w.R} V$$

$$\omega_l = \frac{R-L}{R_w.R} V$$

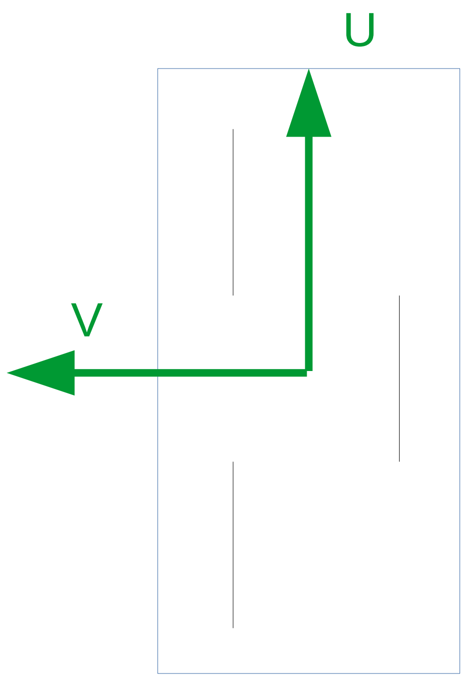
$$\omega_r = \frac{V + L.\omega}{R_w}$$

$$\omega_l = \frac{V - L.\omega}{R_w}$$

OMNIDIRECTIONAL Robot



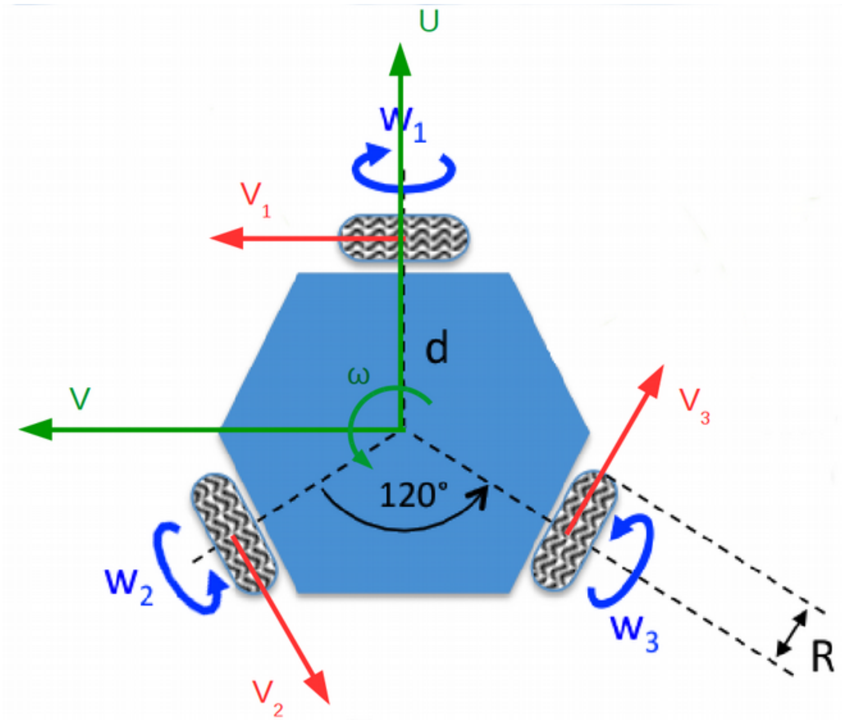
Omnidirectional wheel (Swedish wheel)



Bottom view

- rotates on itself
- moves on U (controlled)
- moves on V (free)

OMNIDIRECTIONAL Robot



Independant control vectors :

$$\vec{u}\,,\vec{v}\,,\vec{\omega}$$

Any translation, rotation
When and where you want...

HOW IT WORKS...

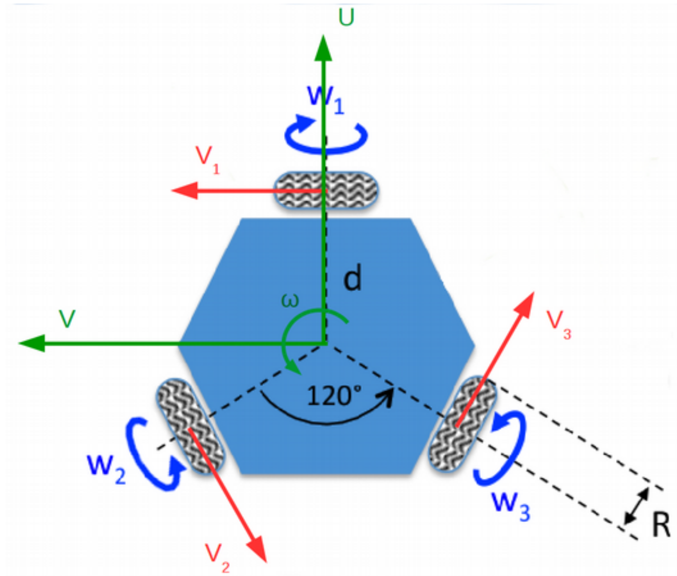
Speed equation :

Control Vectors : $\vec{u}, \vec{v}, \vec{\omega}$

$$v_1 = -u.\sin(0) + v.\cos(0) + d.\omega$$

$$v_2 = -u.\sin\left(\frac{2\pi}{3}\right) + v.\cos\left(\frac{2\pi}{3}\right) + d.\omega$$

$$v_3 = -u.\sin\left(\frac{-2\pi}{3}\right) + v.\cos\left(\frac{-2\pi}{3}\right) + d.\omega$$

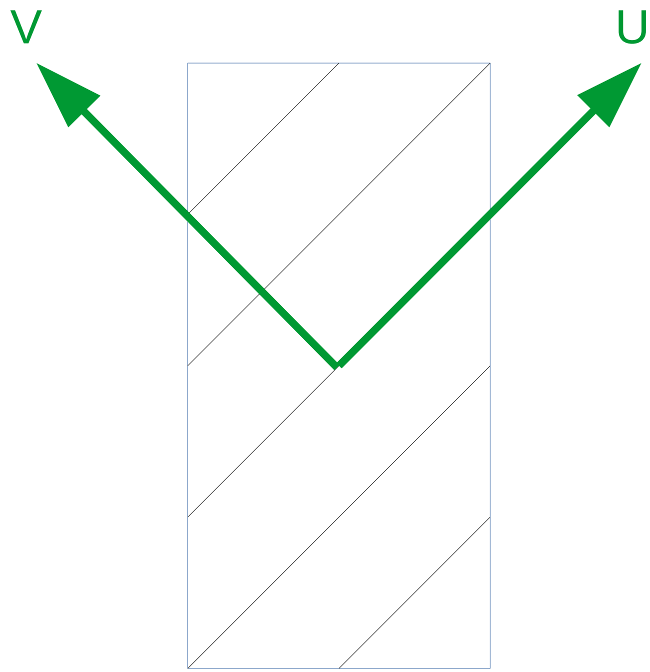


Motor command :

$$\omega_1 = \frac{v_1}{R} \qquad \omega_2 = \frac{v_2}{R} \qquad \omega_3 = \frac{v_3}{R}$$

Robot MECANUM

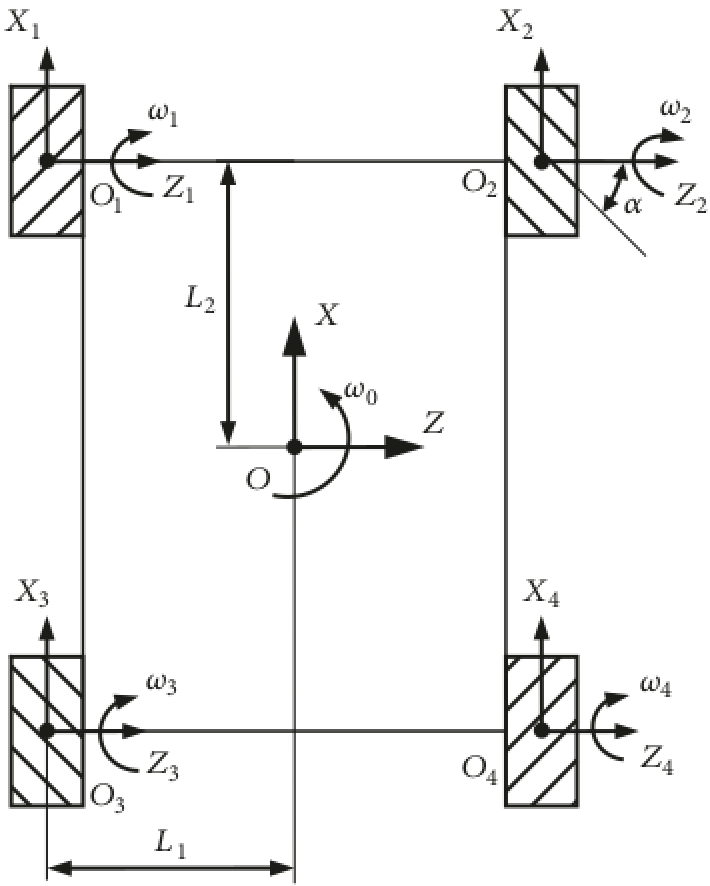
Mecanum wheel



Bottom view

- rotates on itself
- moves on U (controlled)
- moves on V (free)

Robot MECANUM

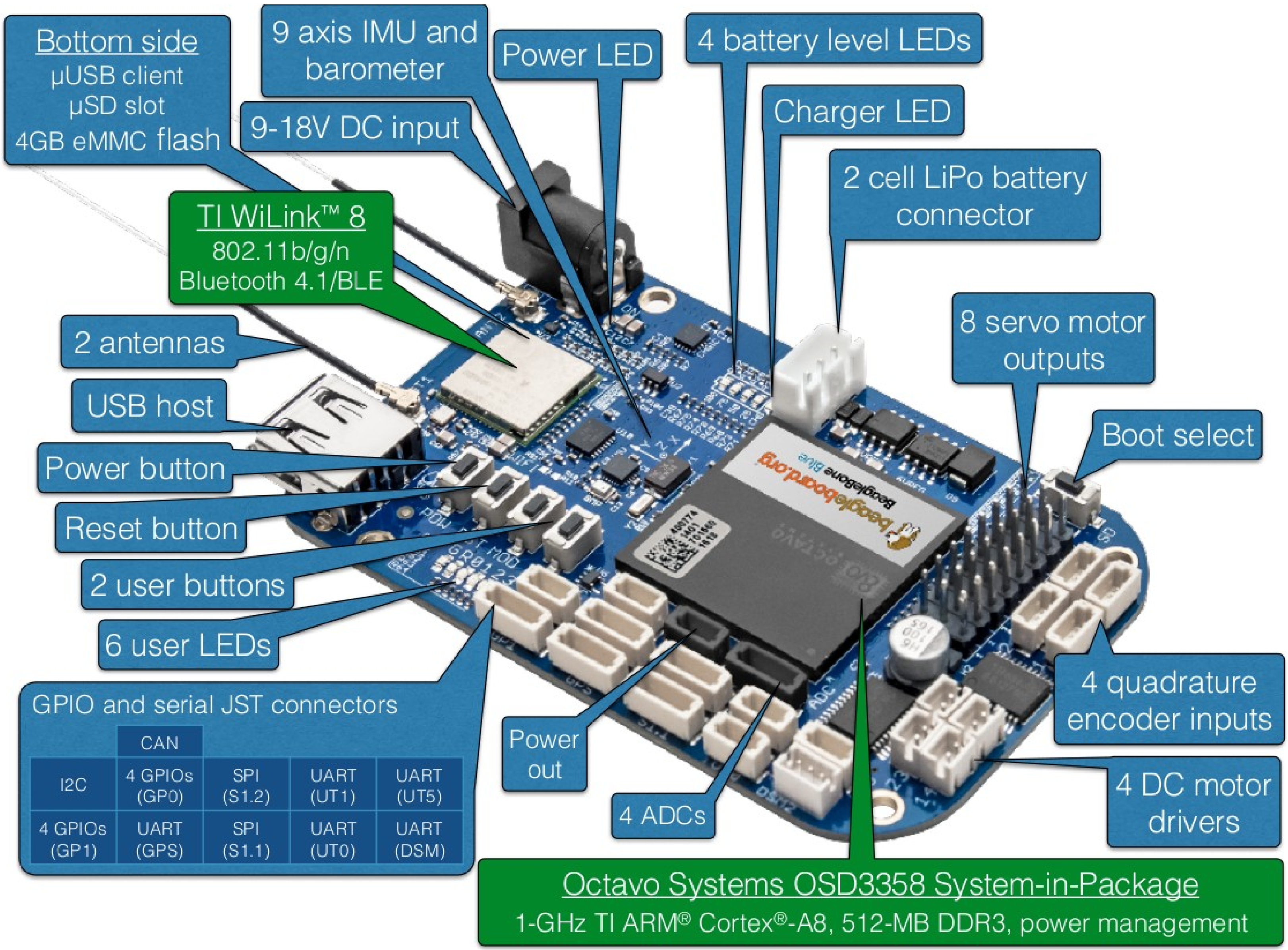


Speed motor equation

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = \frac{1}{r} \begin{pmatrix} 1 & 1 & -(L_1 + L_2) \\ -1 & 1 & -(L_1 + L_2) \\ 1 & -1 & -(L_1 + L_2) \\ -1 & -1 & -(L_1 + L_2) \end{pmatrix} \cdot \begin{pmatrix} u \\ v \\ omega \end{pmatrix}$$

$$\begin{pmatrix} u \\ v \\ omega \end{pmatrix} = \frac{r}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -\frac{1}{L_1 + L_2} & \frac{1}{L_1 + L_2} & -\frac{1}{L_1 + L_2} & \frac{1}{L_1 + L_2} \end{pmatrix} \cdot \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix}$$

The BeagleBoneBlue

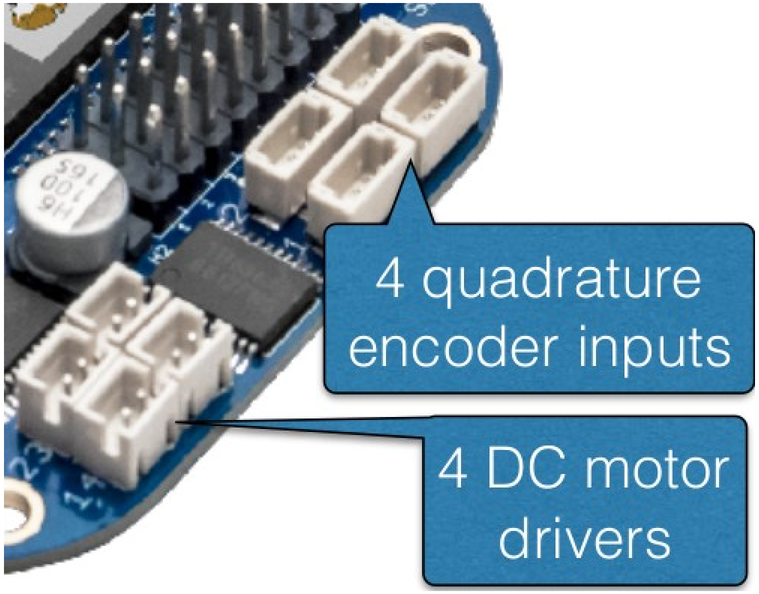


POLYTECH MONTPELLIER

MUSE
MONTPELLIER UNIVERSITY OF EXCELLENCE

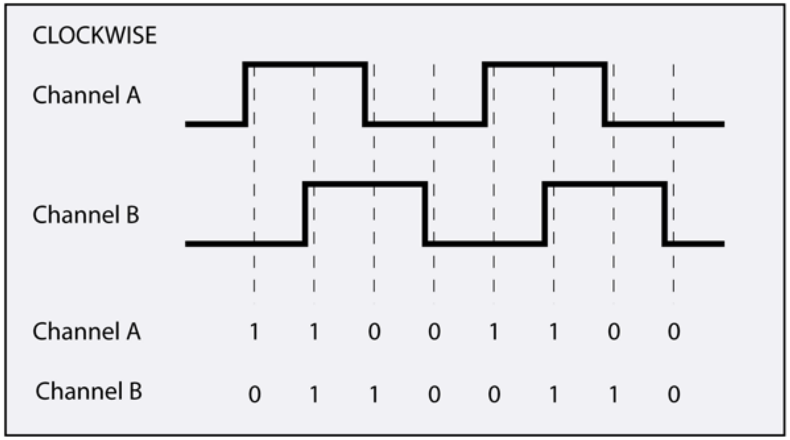
UNIVERSITÉ MONTPELLIER

The BeagleBoneBlue

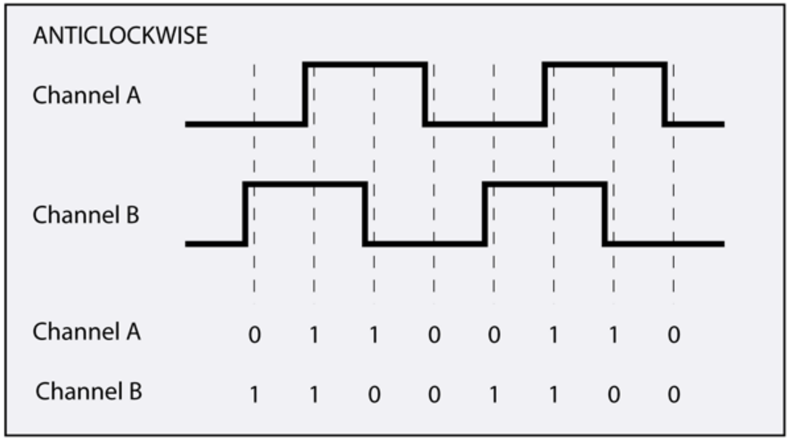


Code example in tutorial, and in the BeagleBone...

Quadrature encoders



Counter up



Counter down

http://www.creative-robotics.com/quadrature-intro

POLYTECH MONTPELLIER

MUSE MONTPELLIER UNIVERSITY OF EXCELLENCE

To Work !

- Choose your robot
- Follow the tutorial
- And make them move correctly