

REDI



Armando Sousa
asousa@fe.up.pt

Contribuidores:

Hugo Costa, Susana Neves, Filipe Lopes, Diogo Rafael Santos,
Sara Paiva, Teresa Eduarda Sousa, Filipa Barros
Estudantes MIEEC + MIEIC / DEEC + DEI / FEUP

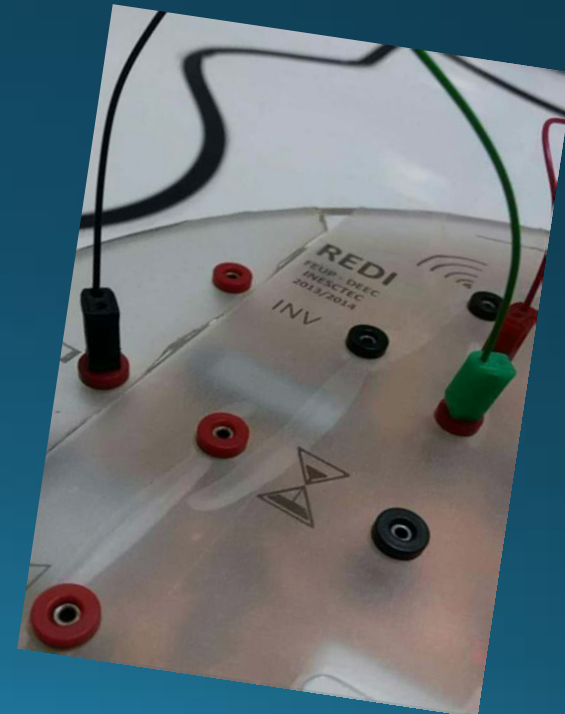
REDI

- Introdução à Robótica
- Introdução à “programação”
- Computação física e visual
- Desenvolver o interesse pelas áreas das Ciências e Engenharias
- Aproximação com a tecnologia
- Desmistificação



Porquê o REDI?

- Ensinar a lógica de programação de uma forma simples e divertida;
- Robot criado por estudantes da FEUP;



REDI:Conceitos

- “Programação” = definição de comportamentos
- Comportamentos através de de conexões físicas
- Controlador (CPU)
- Sensores
- Atuadores
- Robot em sistema Real



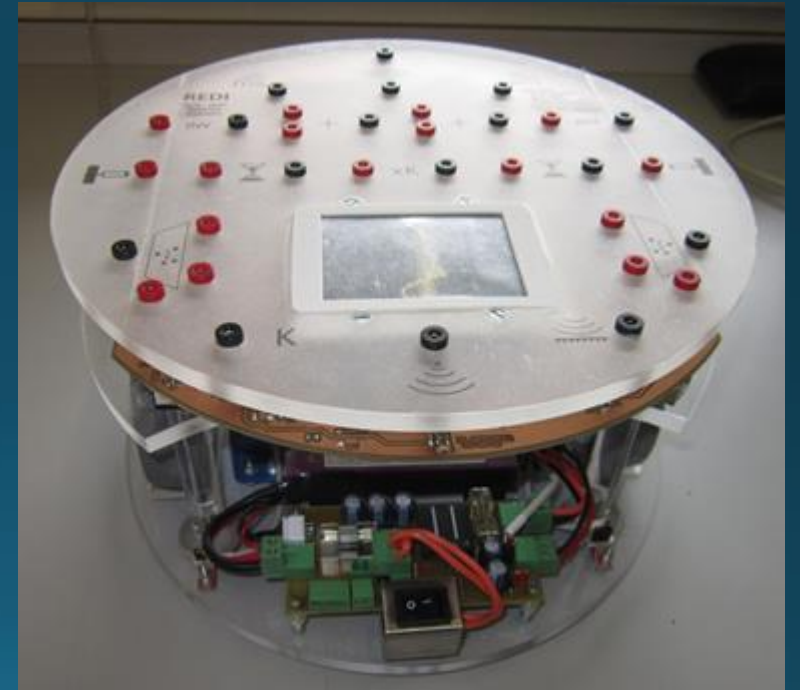
Sensores e Atuadores

- um exemplo vivo

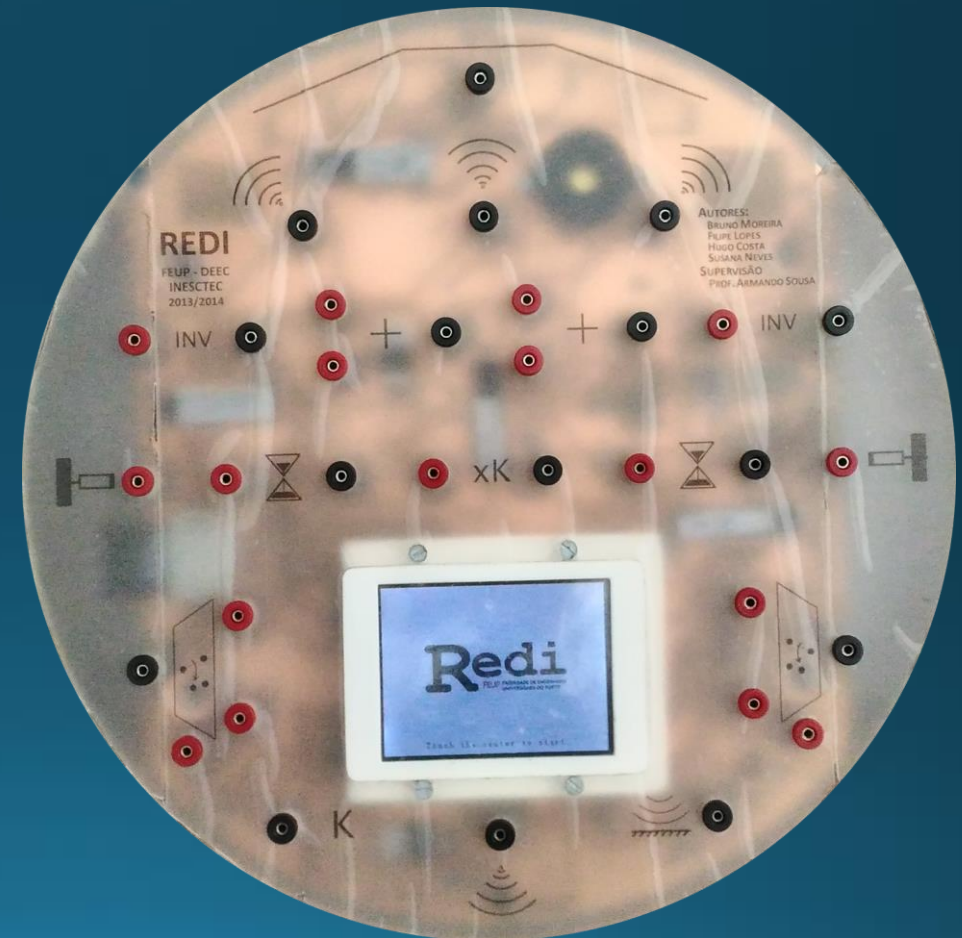
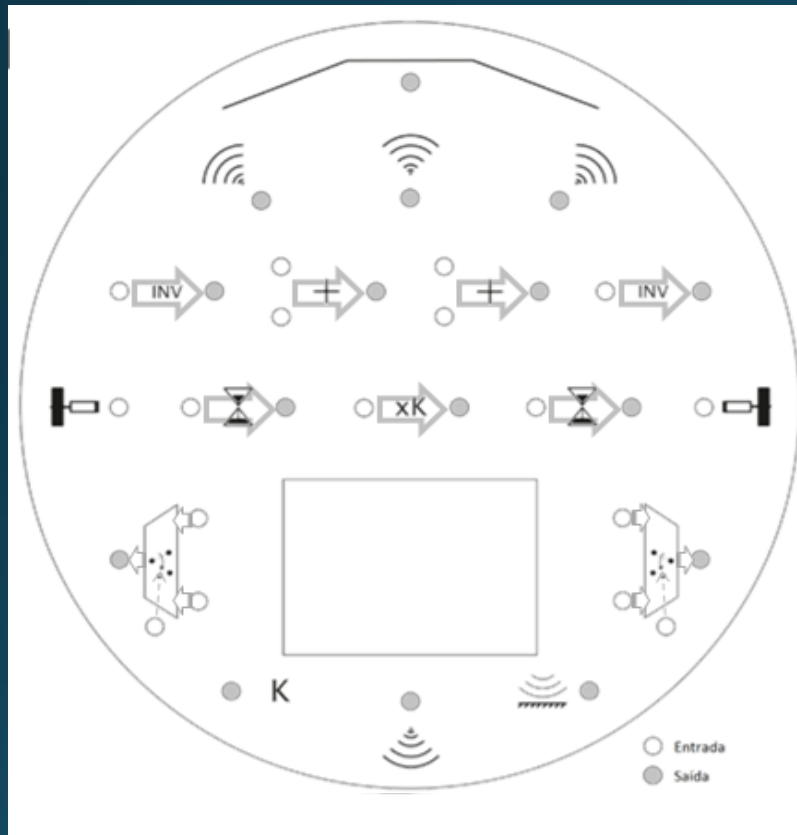


Robot

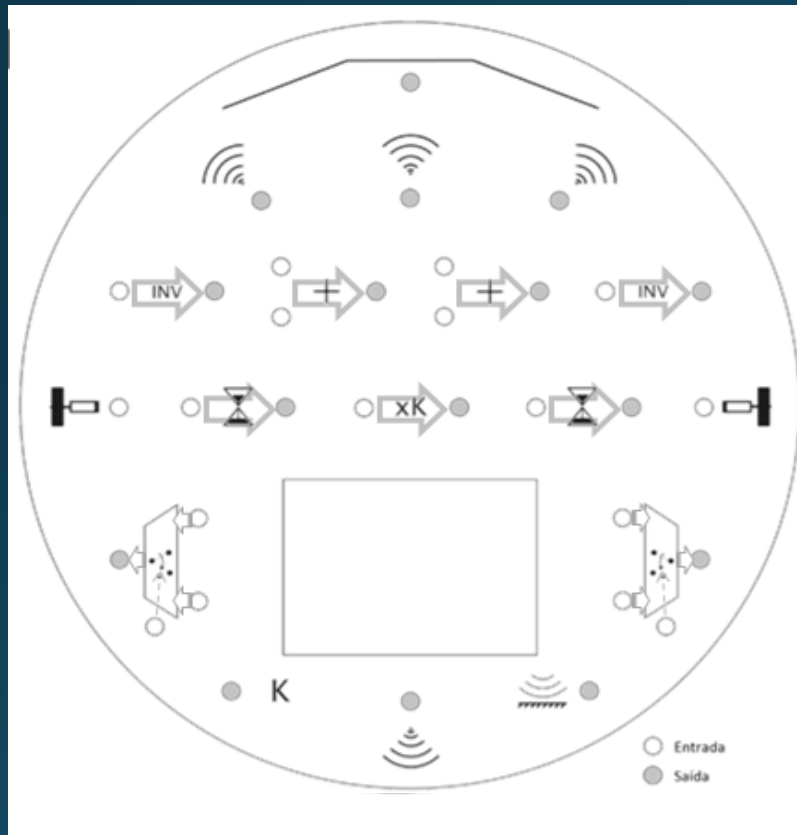
- Structure;
- 2 Electrical Motors;
- Sensors;
- Battery;
- Define behaviour
“Programming”;



Robot e blocos



Robot e blocos



Sensors

Atuators

Invert

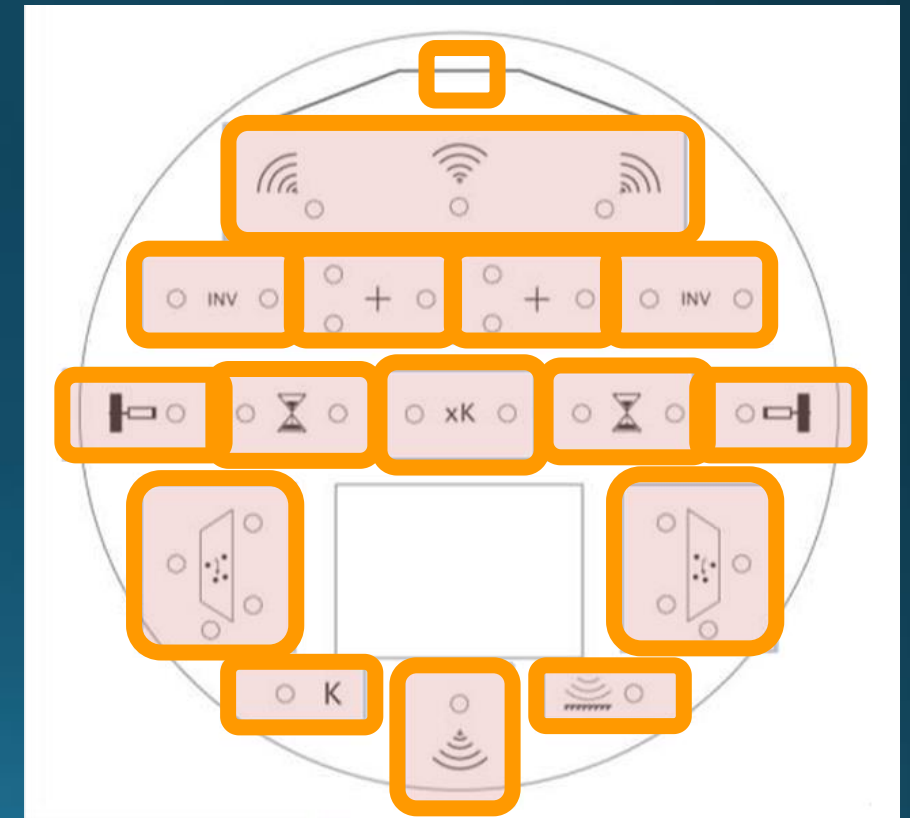
Add

Timer

Multiplexor

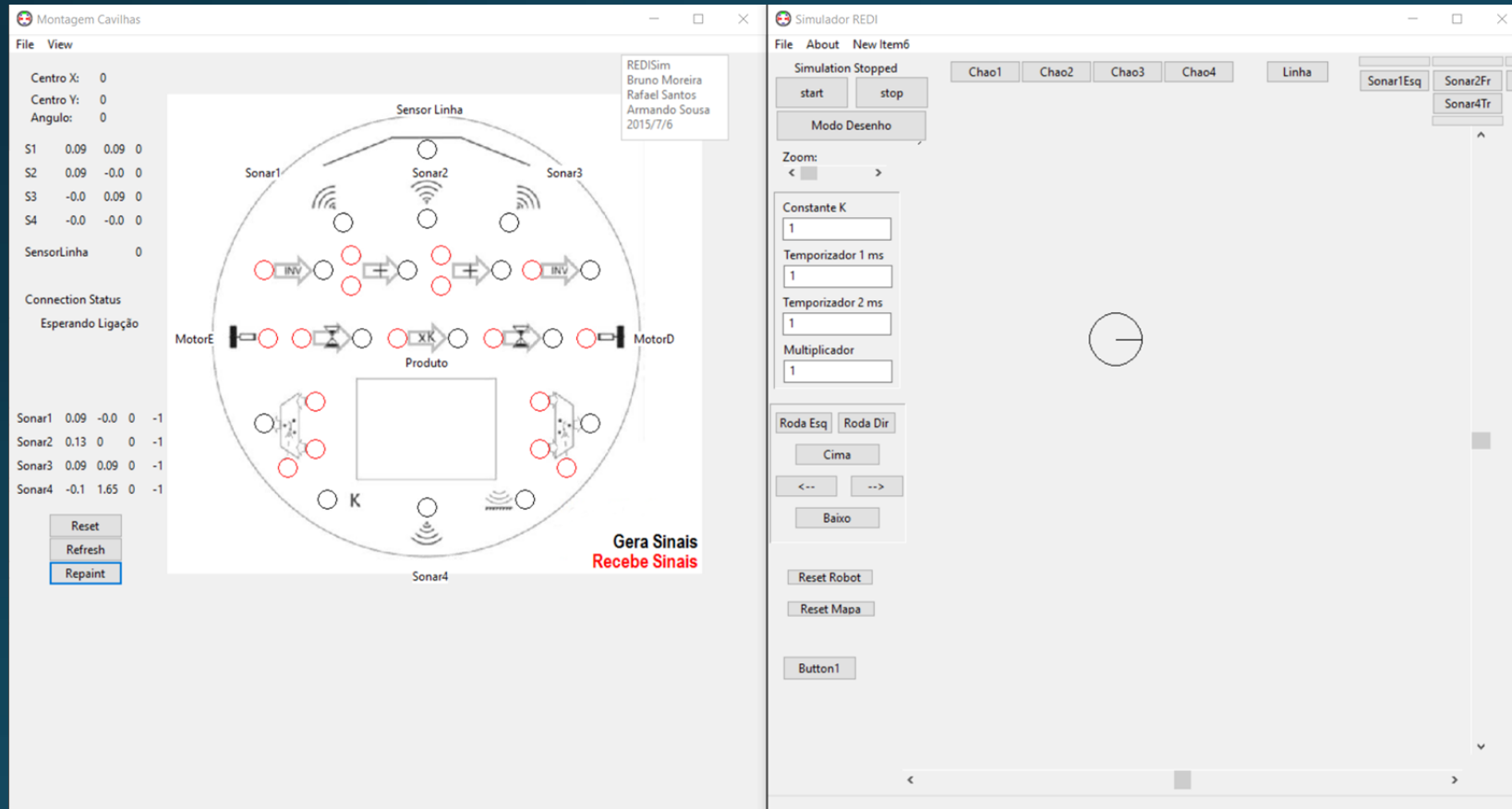
Multiply

Constant



Simulador

- Simula a interface e o comportamento do robot;



[https://fe.up.pt/
asousa/redisim.exe](https://fe.up.pt/asousa/redisim.exe)

Desafios

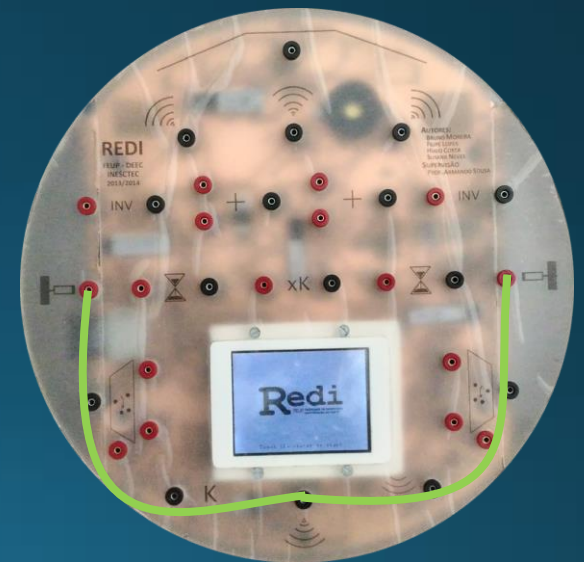
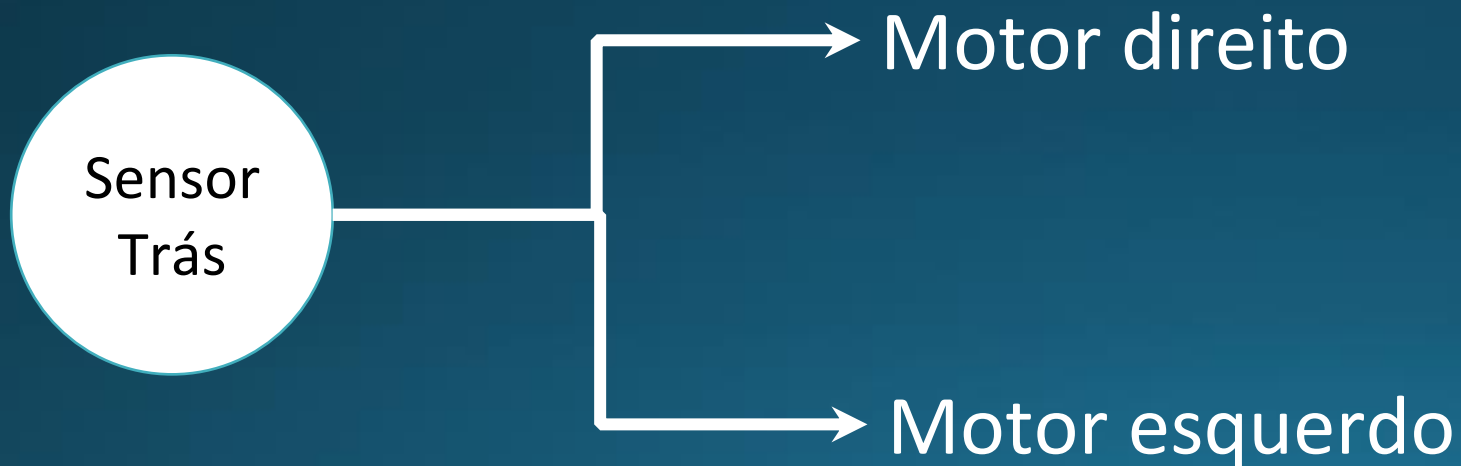


Listagem dos desafios

1. Run from back
2. Magnet
3. Run from front
4. Turn
5. Robot in the middle
6. Keep Distance
7. Avoid obstacle
8. Follow Walls
9. Follow Line
10. Go and avoid
11. Keep Inside

Desafio: robot fuge (de trás)

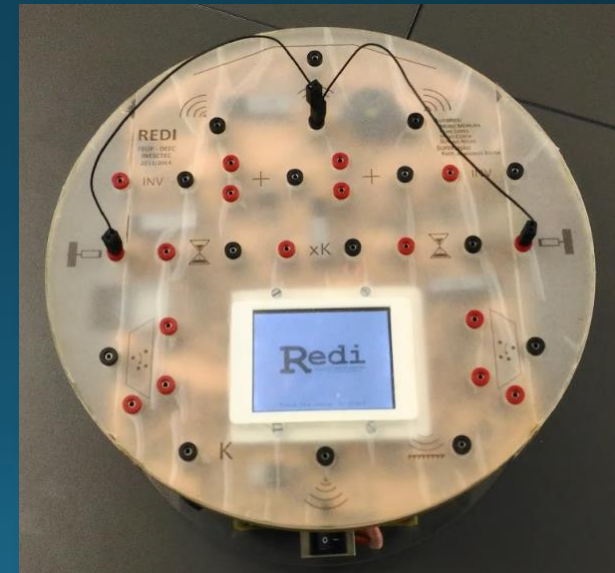
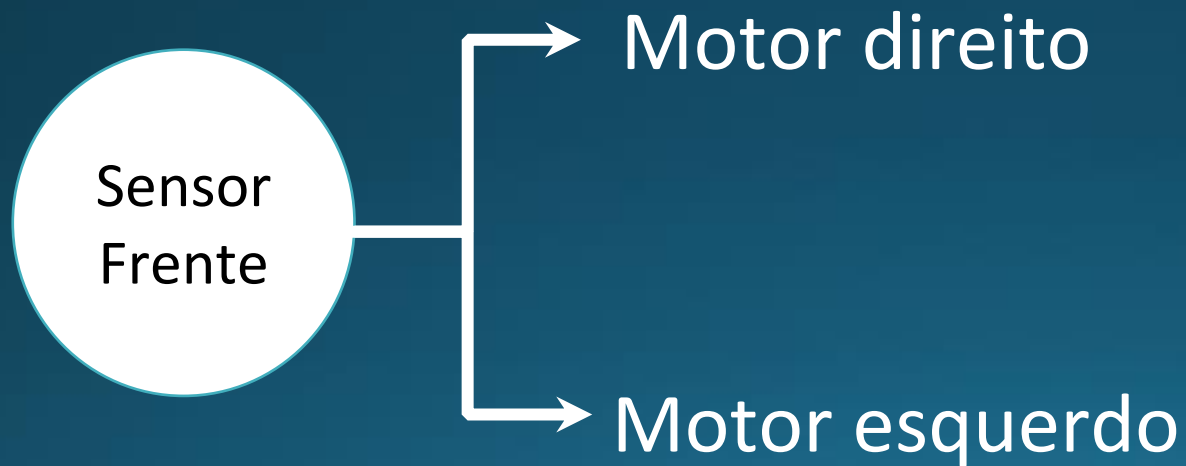
- Quando existe algum obstáculo atrás do robot, este deve afastar-se;
- Quando não existir nenhum obstáculo à frente deve ficar parado.



Motor_Right = Sensor_Back;
Motor_Left = Sensor_Back;

Desafio: Robot Íman

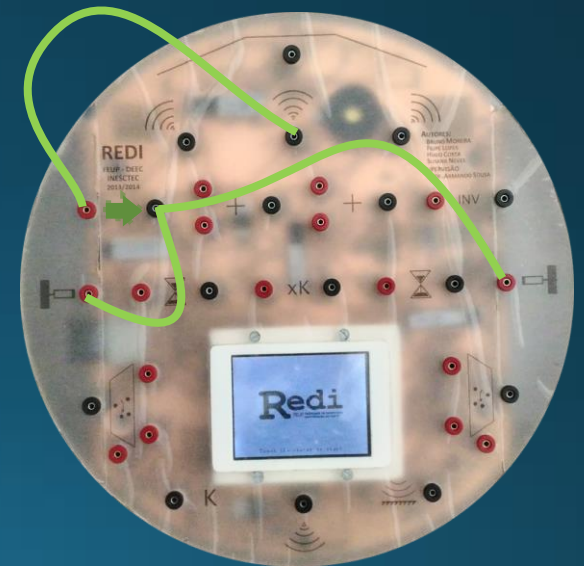
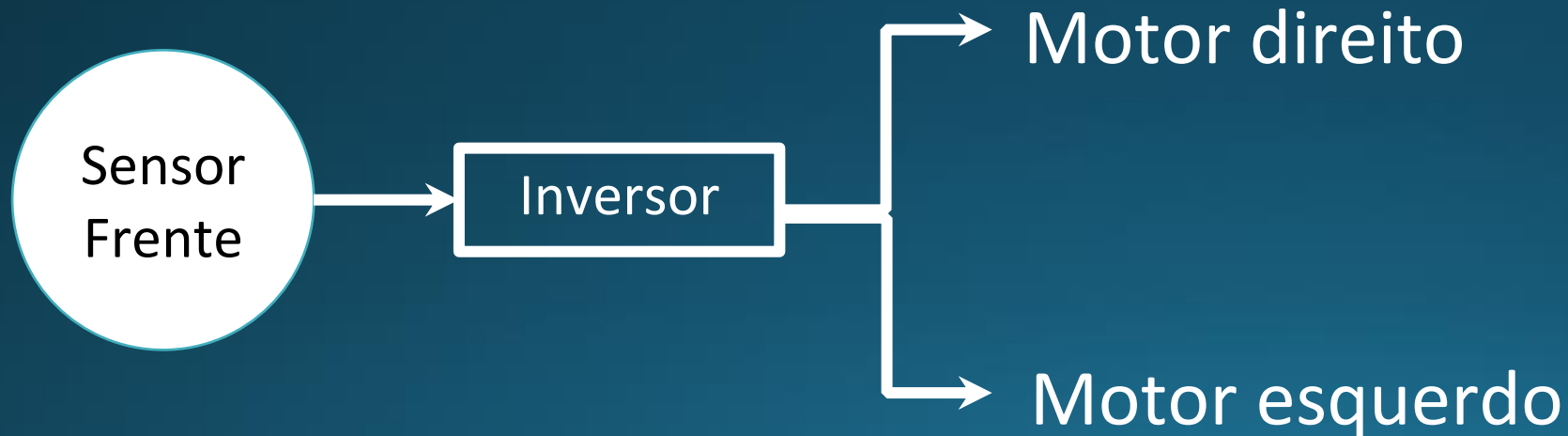
- Quando existir algum obstáculo à frente do robot este deve avançar para ele;
- Quando não existir nenhum obstáculo o robot deve ficar parado .



Motor_Right = Sensor_Front;
Motor_Left = Sensor_Front;

Desafio: robot foge da frente

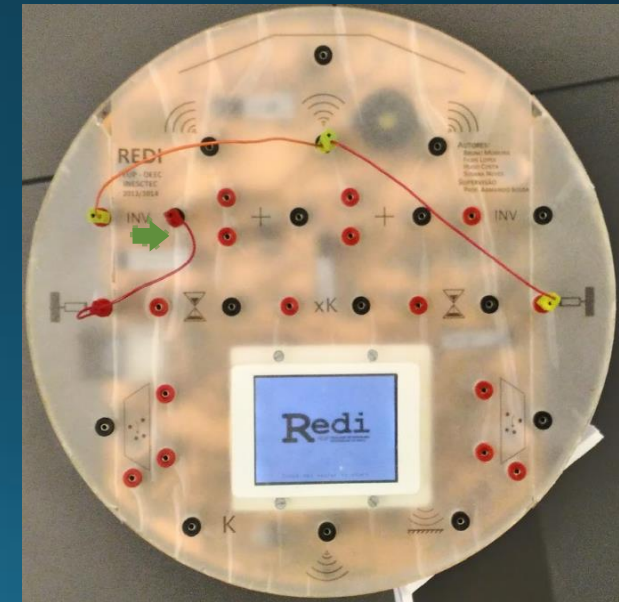
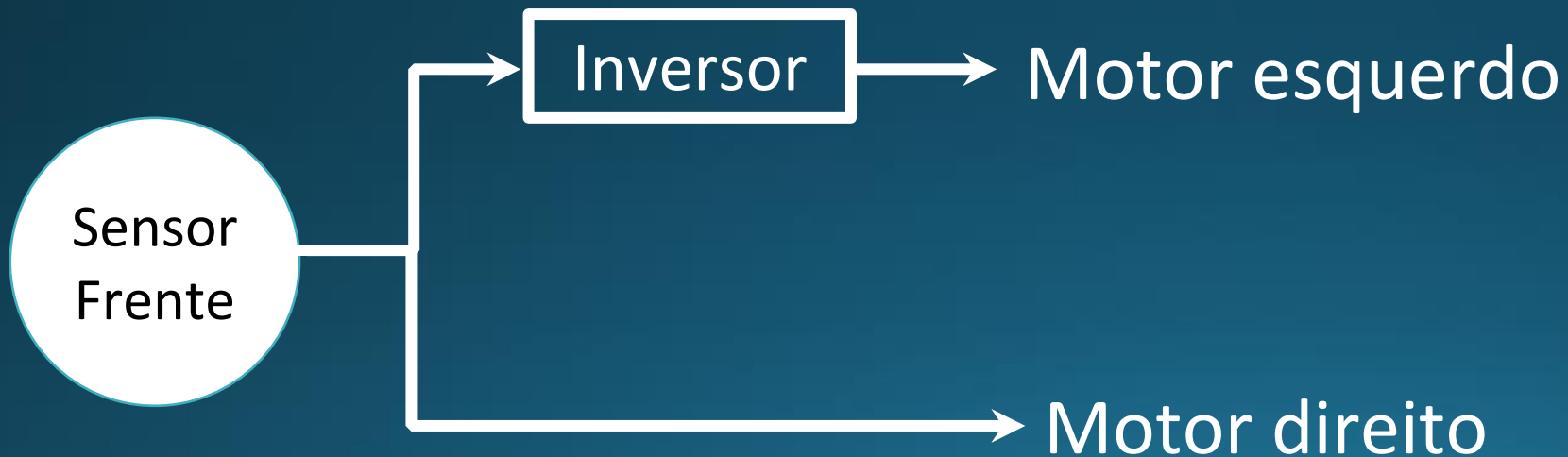
- Quando existe algum obstáculo à frente do robot este deve afastar-se dele;
- Quando não existir nenhum obstáculo à frente deve ficar parado.



Motor_Right = - Sensor_Front;
Motor_Left = - Sensor_Front;

Desafio: robot que se vira

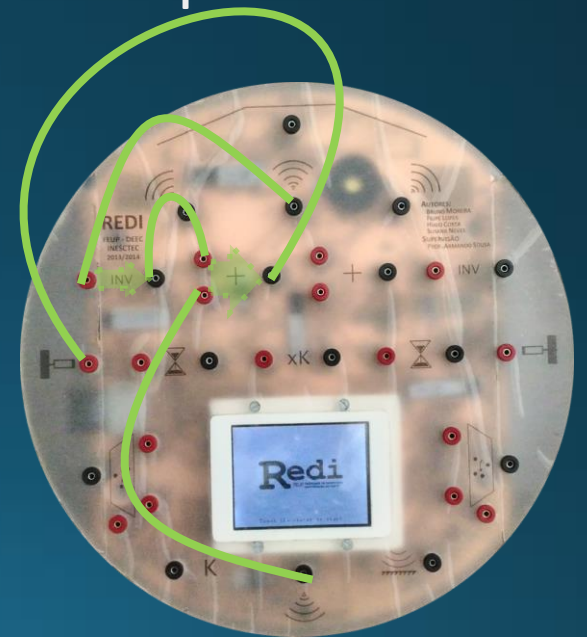
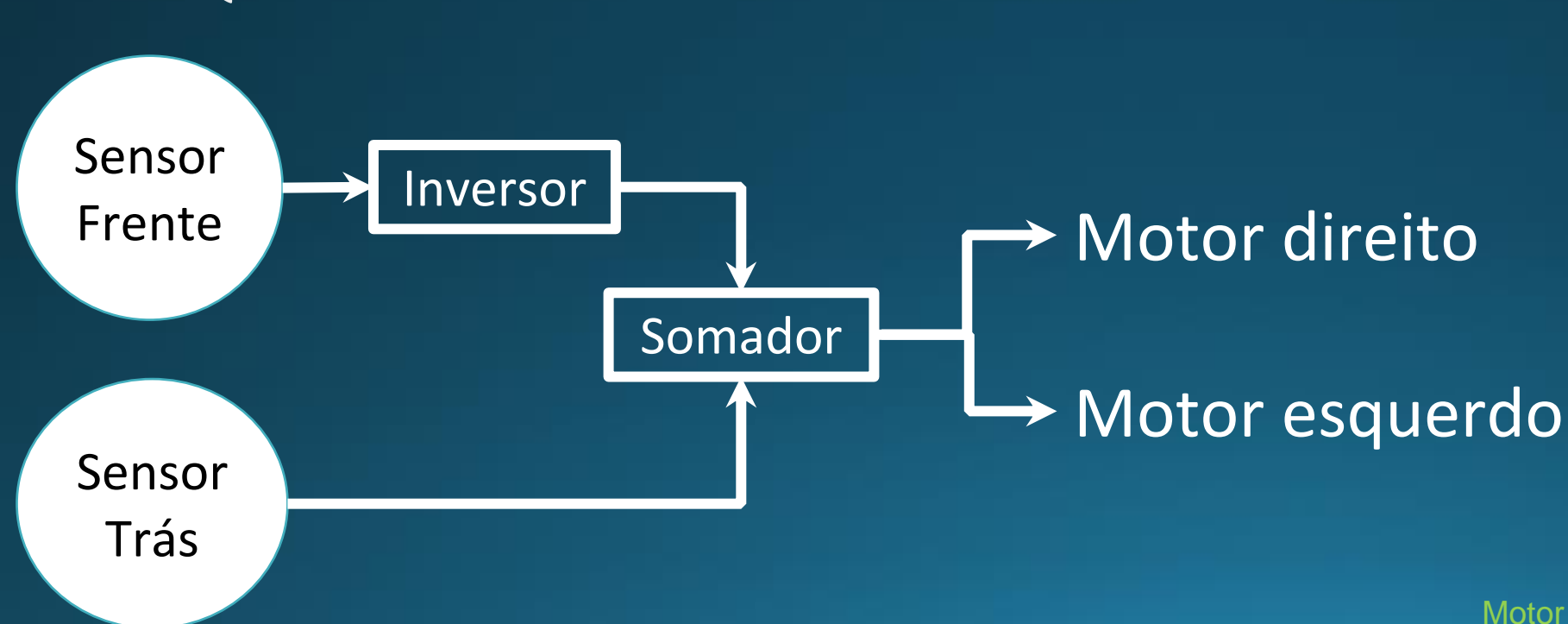
- Quando existir algum obstáculo à frente do robot este deve rodar sobre si próprio;
- Quando não existir nenhum obstáculo à frente do robot este deve ficar parado.



Motor_Right = - Sensor_Front;
Motor_Left = Sensor_Front;

Desafio: robot no meio

- Quando existir algum obstáculo à frente e outro atrás do robot, este deve colocar-se no meio destes dois, mantendo uma distância igual entre si e cada obstáculo;
- Quando não existir nenhum obstáculo o robot este deve ficar parado.

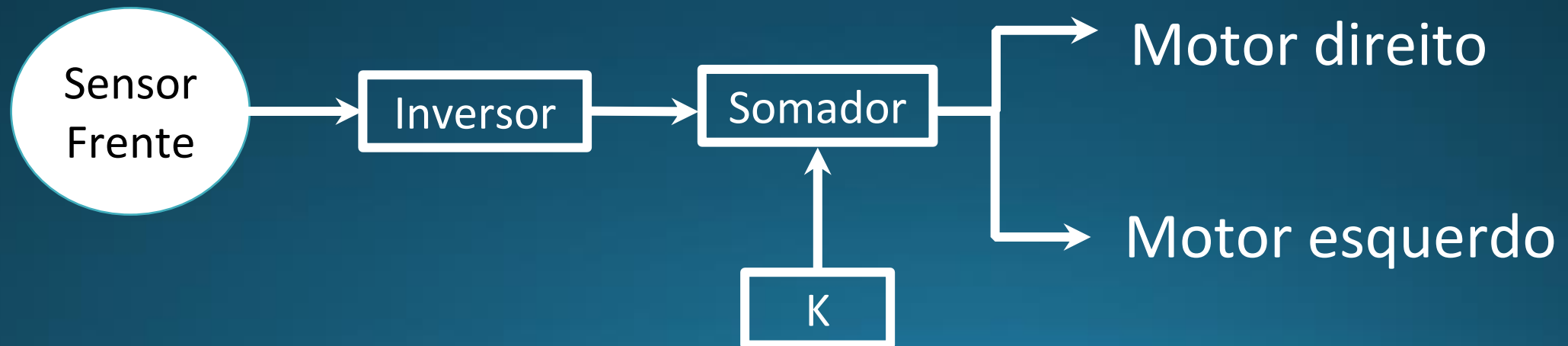


$$\begin{aligned} \text{Motor_Right} &= \text{Sensor_Back} + (- \text{Sensor_Front}); \\ \text{Motor_Left} &= \text{Sensor_Back} + (- \text{Sensor_Front}); \end{aligned}$$

Desafio:

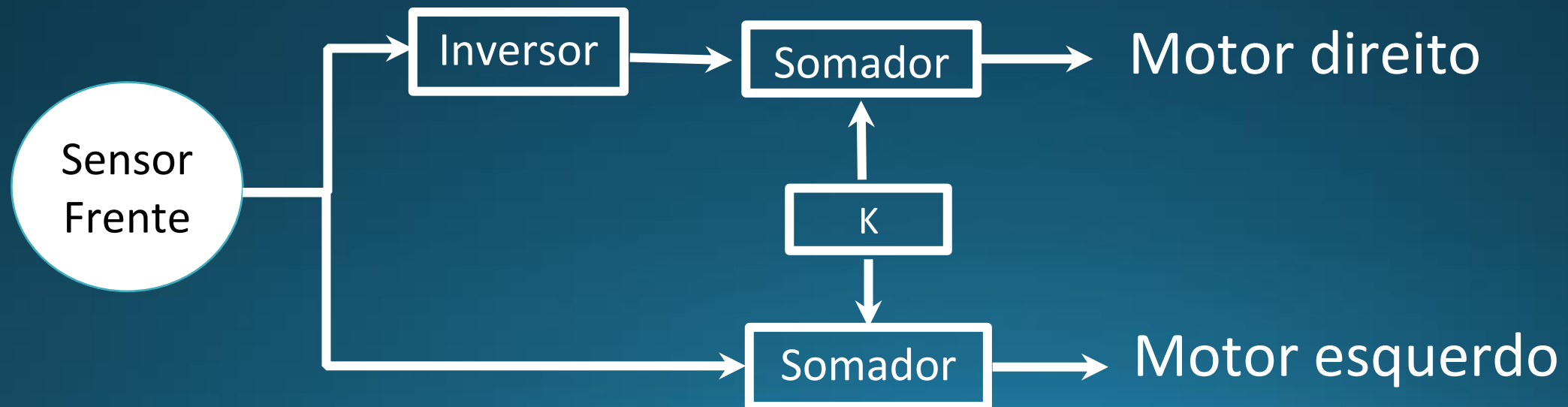
robot que mantém distância de obstáculos

- Quando existir algum obstáculo à frente do robot este deve parar a uma distância fixa deste;
- Quando não existir nenhum obstáculo, deve andar em frente a velocidade constante.



Desafio: robot que evita obstáculos

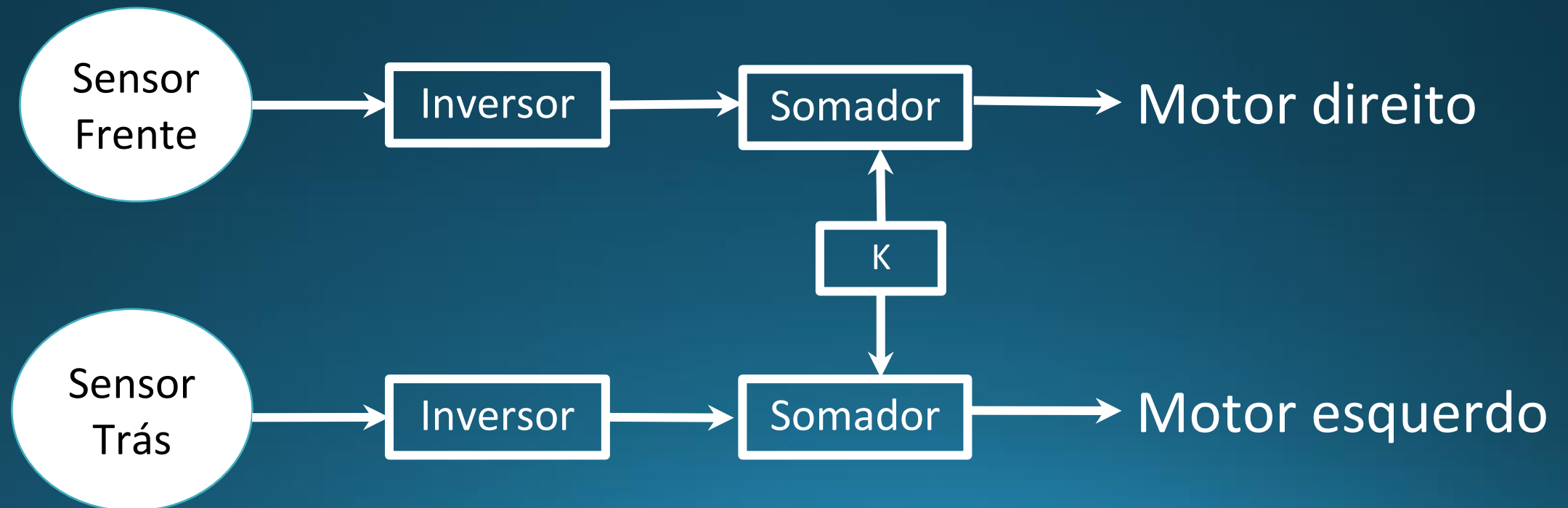
- Quando existir algum obstáculo à frente do robot, este deve desviar-se, curvando para a esquerda;
- Quando não existir nenhum obstáculo este deve andar em frente a velocidade constante.



Desafio:

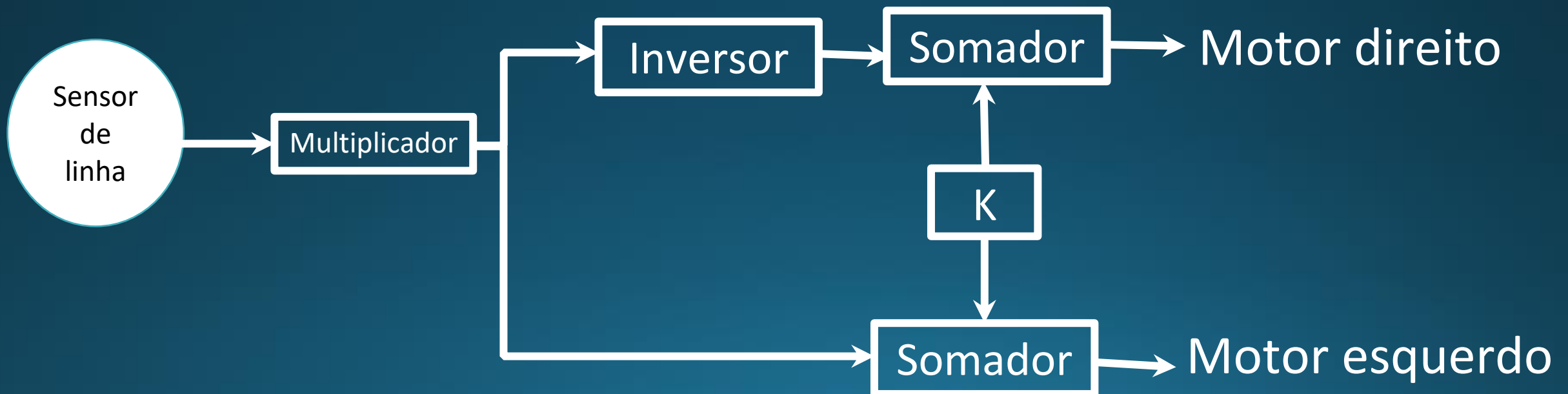
robot segue caminho entre as paredes

- Quando existirem paredes o robot deve mover-se em velocidade constante sem embater nas paredes;
- Quando não existir o robot deve seguir com velocidade constante.



Desafio: robot segue uma linha

- Quando existir alguma linha o robot deve seguir o caminho desenhado por ela;



E agora?



Universidade Júnior Robótica 2015

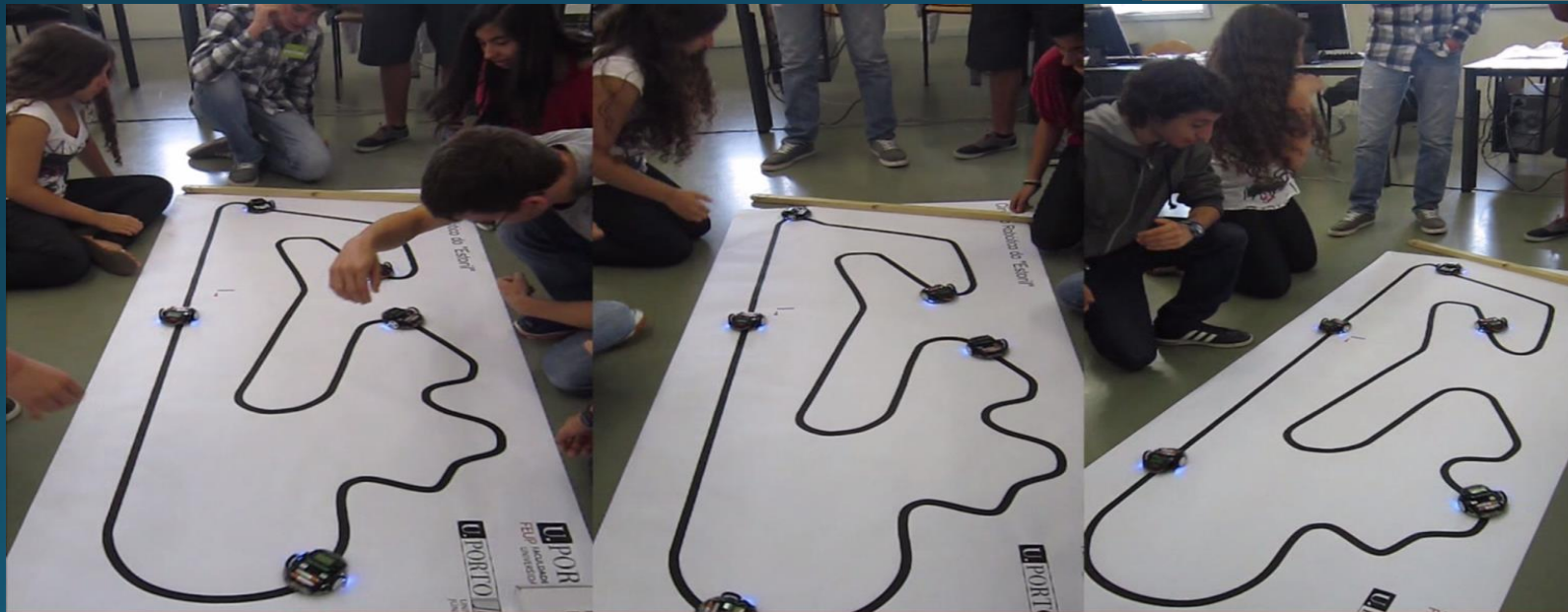
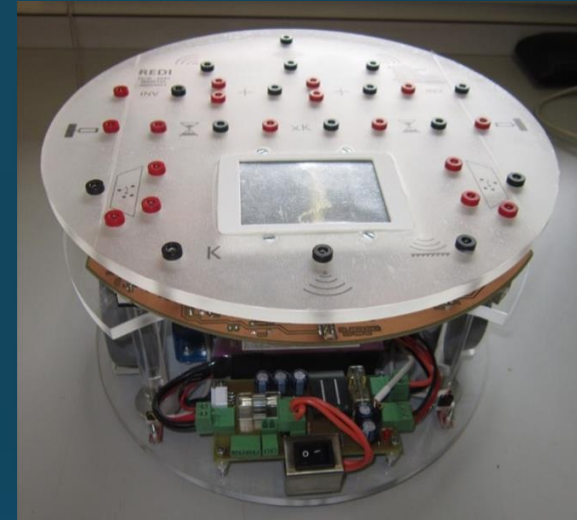
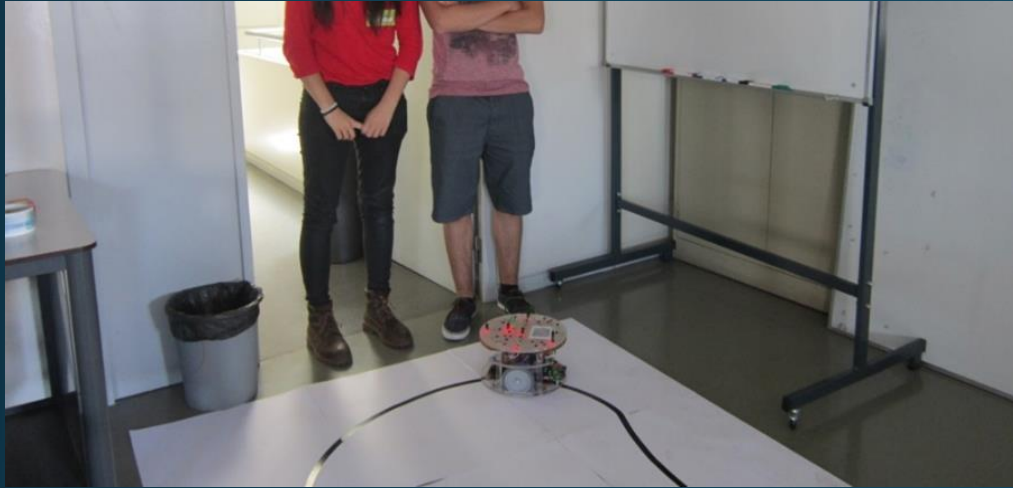




Foto: Jorge Aroso - olhares.sapo.pt

U. PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO