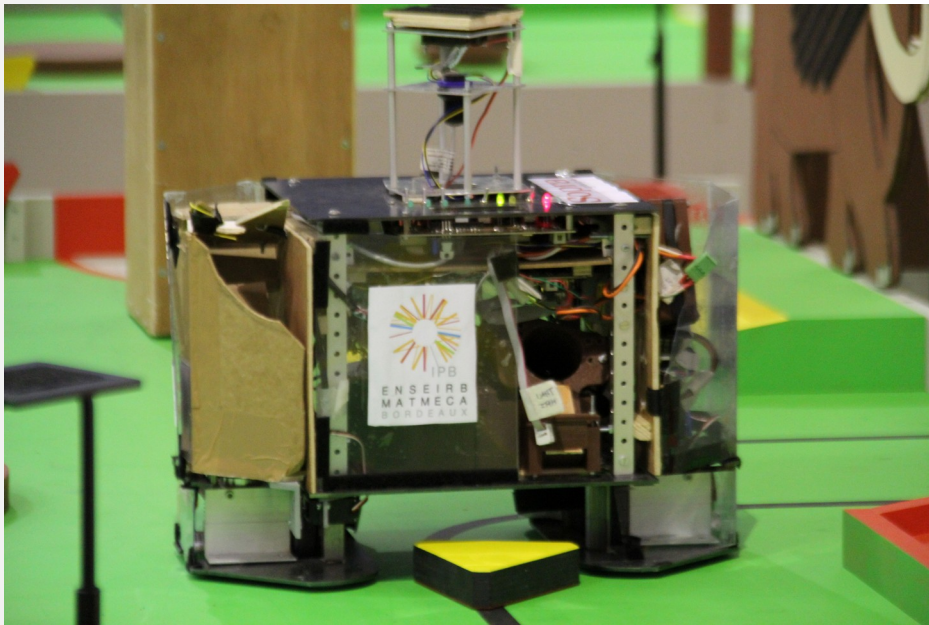


Aversive++

An Open Source C++ library to ease the development of robotic systems on microcontrollers

Robots++



Developers++



- Master degree in computer science from ENSEIRB-MATECA
- Participate to Eurobot contest
- Hired as research engineer at INRIA
- Won Boost Your Code contest

Features++

- Multiplatform Hardware Abstraction Layer
- Simulator : SASIAE

- Design control systems (Filter API)
 - PID, Bang-Bang, etc...
- Actuator Sensor Abstraction (Device API)
 - Motor, servomotor, encoder, distance sensor
- “std::stream like” library (Stream API)
 - UART, I2C, SPI, TCP, ...

Usecase++

```
// Encoder devices
Encoder<s32> left_enc("leftEnc", &le);
Encoder<s32> right_enc("rightEnc", &re);

// Motor devices
Motor<s32> left_motor("leftMot", &lm);
Motor<s32> right_motor("rightMot", &rm);

// COMM
UartStream<0> io("io");
```

define the system's input and output...

Usecase++

```
PidFilter id = PidFilter::identity();

PidFilter left_pid;
PidFilter right_pid;

DiffFilter left_diff;
DiffFilter right_diff;

MotorController left_cmot(left_motor, left_enc, id, left_diff, left_pid);
MotorController right_cmot(right_motor, right_enc, id, right_diff, right_pid);

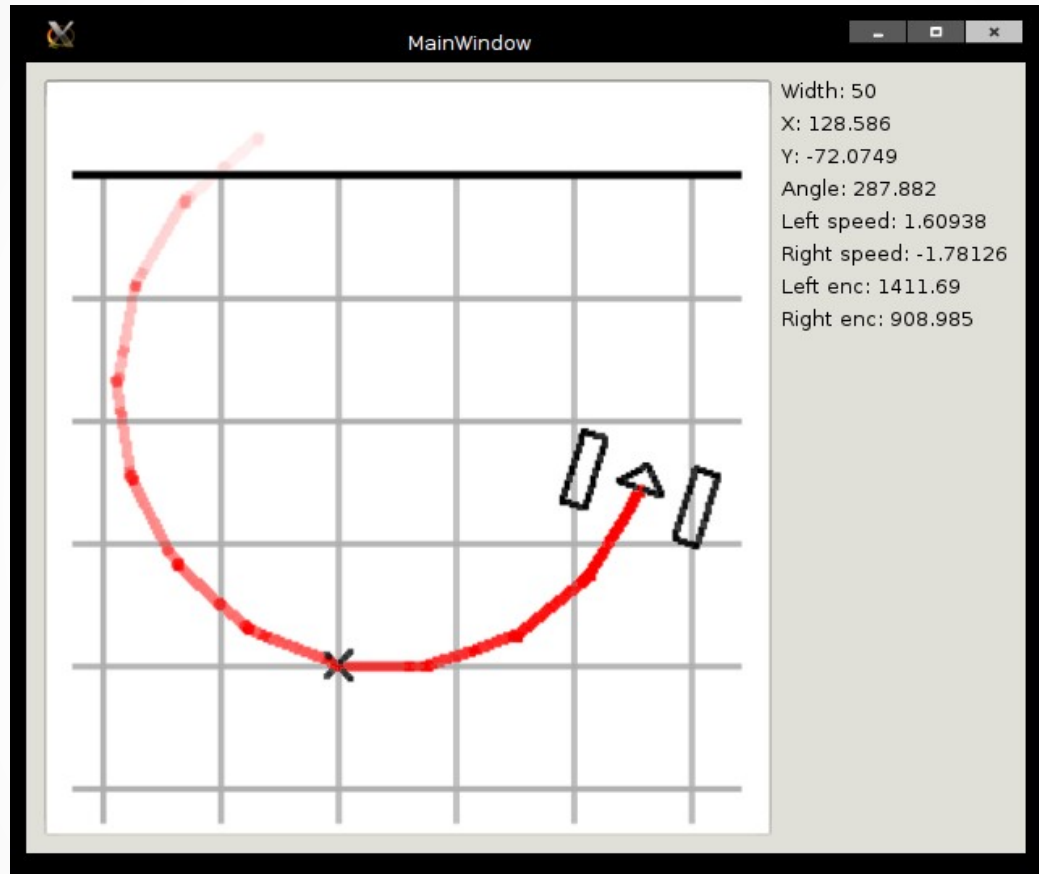
Odometer odo(right_enc, left_enc);

PidFilter dist_pid;
PidFilter angle_pid;

RobotController robot(right_cmot, left_cmot, odo, id, id, dist_pid, id, id, angle_pid);
```

build your control system...

Usecase++



```
robot.setValue(Vect<2, s32>(dist, angle));
```

and just use it !

Contribute++



GitHub

<https://github.com/AversivePlusPlus/AversivePlusPlus>



Forum

<http://forum.aversiveplusplus.com/>



Wiki

[https://github.com/AversivePlusPlus/AversivePlusPlus/
wiki](https://github.com/AversivePlusPlus/AversivePlusPlus/wiki)



Bug tracker

[https://github.com/AversivePlusPlus/AversivePlusPlus/
issues](https://github.com/AversivePlusPlus/AversivePlusPlus/issues)

aversiveplusplus.com

Conclusion++

Thanks for your attention !

Any question ?

NO

