Robot

Generated by Doxygen 1.8.13

Contents

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 2

Class Documentation

2.1 Battery Class Reference

Public Member Functions

- Battery ()

  Construct a new Battery:: Battery object.
- float getVoltage ()

  Returns the current battery voltage.

2.1.1 Member Function Documentation

2.1.1.1 getVoltage()

```
float Battery::getVoltage ( )
```

Returns the current battery voltage.

Returns

float Battery voltage.

The documentation for this class was generated from the following files:

- Battery.h
- Battery.cpp

2.2 Bluetooth Class Reference

Public Member Functions

- void **rename** (const char ∗name)
- void **control** ()

The documentation for this class was generated from the following files:

- Bluetooth.h
- Bluetooth.cpp

2.3 Button Class Reference

Public Member Functions

- uint8_t **pressed** ()
- void **wait** ()
- void **press** ()

The documentation for this class was generated from the following files:

- Button.h
- Button.cpp

2.4 Distance Class Reference

Public Member Functions

- void **calibrate** (int16_t duration)
- float **getDistance** (uint8_t sensor)
- void **readCalibrated** ()
- void **readCalibrated** (uint8_t i)
- void **readRaw** ()
- void **readRaw** (uint8_t i)

Public Attributes

- int16_t **sensorsRaw** [6] = {0, 0, 0, 0, 0, 0}
- int16_t **sensorsMin** [6] = {1023, 1023, 1023, 1023, 1023, 1023}
- int16_t **sensorsMax** [6] = {0, 0, 0, 0, 0, 0}
- float **a** [6] = {0.0, 0.0, 0.0, 0.0, 0.0, 0.0}
- float **b** [6] = {0.0, 0.0, 0.0, 0.0, 0.0, 0.0}
- float **sensorsCalibrated** [6] = {0, 0, 0, 0, 0, 0}
- uint32_t **lastUpdate** = 0

The documentation for this class was generated from the following files:

- Distance.h
- Distance.cpp

2.5 Filters Class Reference

Public Member Functions

- void **MadgwickQuaternionUpdate** (float ax, float ay, float az, float gx, float gy, float gz, float mx, float my, float mz)
- void **MahonyQuaternionUpdate** (float ax, float ay, float az, float gx, float gy, float gz, float mx, float my, float mz)

Public Attributes

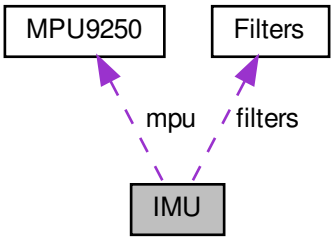
- float **pitch**
- float **yaw**
- float **roll**
- float **q** [4] = {1.0f, 0.0f, 0.0f, 0.0f}

The documentation for this class was generated from the following files:

- Filters.h
- Filters.cpp

2.6 IMU Class Reference

Collaboration diagram for IMU:



Public Member Functions

- float **getRoll** ()
- float **getPitch** ()
- float **getYaw** ()
- float **getAX** ()
- float **getAY** ()
- float **getAZ** ()
- float **getGX** ()
- float **getGY** ()
- float **getGZ** ()
- float **getMX** ()
- float **getMY** ()
- float **getMZ** ()
- void **calibrate** ()

Public Attributes

- Filters **filters** = Filters()
- MPU9250 **mpu** = MPU9250(Wire, 0x68)

The documentation for this class was generated from the following files:

- IMU.h
- IMU.cpp

2.7 Line Class Reference

Public Member Functions

- void **calibrate** (int16_t duration)
- float **getPosition** ()
- float **getAngle** ()
- float **setNoiseThreshold** (int16_t threshold)
- float **setColor** (uint8_t color)
- uint16_t **getSensor** (uint8_t sensor)
- float **getLength** ()
- void **setLength** (float length)
- void **readCalibrated** ()
- void **readRaw** ()

Public Attributes

- int16_t **sensorsRaw** [8] = {0, 0, 0, 0, 0, 0, 0, 0}
- int16_t **sensorsMin** [8] = {1023, 1023, 1023, 1023, 1023, 1023, 1023, 1023}
- int16_t **sensorsMax** [8] = {0, 0, 0, 0, 0, 0, 0, 0}
- int16_t **sensorsCalibrated** [8] = {0, 0, 0, 0, 0, 0, 0, 0}
- int16_t **linePrevious** = 0
- int16_t **noiseThreshold** = 500
- uint8_t **color** = BLACK
- float **length** = 0.082
- uint32_t **lastUpdate** = 0

The documentation for this class was generated from the following files:

- Line.h
- Line.cpp

2.8 Motor Class Reference

Public Member Functions

- **Motor** (uint8_t motor)
- float **getVoltage** ()
- void **setVoltage** (float u)
- float **getSpeed** ()
- void **setSpeed** (float w)
- void **setSpeedGains** (float kp, float ki, float kd)
- void **setSpeedILimit** (float limit)
- void **resetI** ()
- float **getDistance** ()
- void **resetDistance** ()
- float **getDistanceRad** ()
- float **getDiameter** ()
- void **setDiameter** (float diameter)
- void **CAPT_ISR** (uint8_t motor)
- void **OVF_ISR** ()

The documentation for this class was generated from the following files:

- Motor.h
- Motor.cpp

2.9 MPU9250 Class Reference

Public Types

- enum **GyroRange** { **GYRO_RANGE_250DPS**, **GYRO_RANGE_500DPS**, **GYRO_RANGE_1000DPS**, **GYRO_RANGE_2000DPS** }
- enum **AccelRange** { **ACCEL_RANGE_2G**, **ACCEL_RANGE_4G**, **ACCEL_RANGE_8G**, **ACCEL_RANGE_16G** }
- enum **DlpfBandwidth** {
  DLPF_BANDWIDTH_184HZ, **DLPF_BANDWIDTH_92HZ**, **DLPF_BANDWIDTH_41HZ**, **DLPF_BANDWIDTH_20HZ**,
  DLPF_BANDWIDTH_10HZ, **DLPF_BANDWIDTH_5HZ** }
- enum **LpAccelOdr** {
  LP_ACCEL_ODR_0_24HZ = 0, **LP_ACCEL_ODR_0_49HZ** = 1, **LP_ACCEL_ODR_0_98HZ** = 2, **LP_ACCEL_ODR_1_95HZ** = 3,
  LP_ACCEL_ODR_3_91HZ = 4, **LP_ACCEL_ODR_7_81HZ** = 5, **LP_ACCEL_ODR_15_63HZ** = 6, **LP_ACCEL_ODR_31_25HZ** = 7,
  LP_ACCEL_ODR_62_50HZ = 8, **LP_ACCEL_ODR_125HZ** = 9, **LP_ACCEL_ODR_250HZ** = 10, **LP_ACCEL_ODR_500HZ** = 11 }

Public Member Functions

- **MPU9250** (TwoWire &bus, uint8_t address)
- int **begin** ()
- int **setAccelRange** (AccelRange range)
- int **setGyroRange** (GyroRange range)
- int **setDlpfBandwidth** (DlpfBandwidth bandwidth)
- int **setSrd** (uint8_t srd)
- int **readSensor** ()
- float **getAccelX_mss** ()
- float **getAccelY_mss** ()
- float **getAccelZ_mss** ()
- float **getGyroX_rads** ()
- float **getGyroY_rads** ()
- float **getGyroZ_rads** ()
- float **getMagX_uT** ()
- float **getMagY_uT** ()
- float **getMagZ_uT** ()
- float **getTemperature_C** ()
- int **calibrateGyro** ()
- float **getGyroBiasX_rads** ()
- float **getGyroBiasY_rads** ()
- float **getGyroBiasZ_rads** ()
- void **setGyroBiasX_rads** (float bias)
- void **setGyroBiasY_rads** (float bias)
- void **setGyroBiasZ_rads** (float bias)
- int **calibrateAccel** ()
- float **getAccelBiasX_mss** ()
- float **getAccelScaleFactorX** ()
- float **getAccelBiasY_mss** ()
- float **getAccelScaleFactorY** ()
- float **getAccelBiasZ_mss** ()
- float **getAccelScaleFactorZ** ()
- void **setAccelCalX** (float bias, float scaleFactor)
- void **setAccelCalY** (float bias, float scaleFactor)
- void **setAccelCalZ** (float bias, float scaleFactor)
- int **calibrateMag** ()
- float **getMagBiasX_uT** ()
- float **getMagScaleFactorX** ()
- float **getMagBiasY_uT** ()
- float **getMagScaleFactorY** ()
- float **getMagBiasZ_uT** ()
- float **getMagScaleFactorZ** ()
- void **setMagCalX** (float bias, float scaleFactor)
- void **setMagCalY** (float bias, float scaleFactor)
- void **setMagCalZ** (float bias, float scaleFactor)

Public Attributes

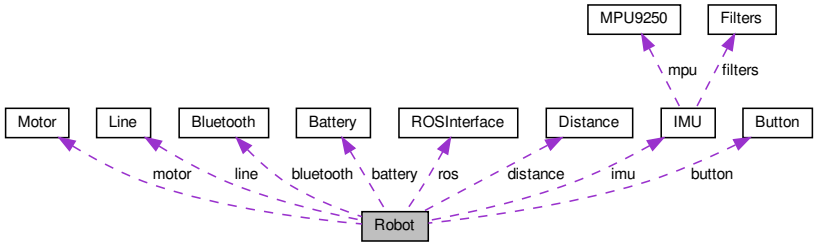
- float **_ax**
- float **_ay**
- float **_az**
- float **_gx**
- float **_gy**
- float **_gz**
- float **_hx**
- float **_hy**
- float **_hz**
- float **_t**

The documentation for this class was generated from the following files:

- MPU9250.h
- MPU9250.cpp

2.10 Robot Class Reference

Collaboration diagram for Robot:



Public Member Functions

- void **stop** ()
- void **go** (float distance)
- void **go** (float distance, float v)
- void **turn** (float angle)
- void **turn** (float angle, float w)
- void **drive** (float v, float w)
- void **beep** (int16_t frequency, int16_t duration)
- void **setWidth** (float width)
- float **getSpeed** ()
- float **getAngularVelocity** ()

Public Attributes

- Button **button** = Button()
- Battery **battery** = Battery()
- Motor **motor** [2] = {Motor(LEFT), Motor(RIGHT)}
- Line **line** = Line()
- Distance **distance** = Distance()
- IMU **imu** = IMU()
- Bluetooth **bluetooth** = Bluetooth()
- ROSInterface **ros** = ROSInterface()

The documentation for this class was generated from the following files:

- Robot.h
- Robot.cpp

2.11 ROSInterface Class Reference

Public Member Functions

- void **subscribe** ()
- void **publishBatteryState** ()
- void **publishOdometry** ()
- void **publishJointState** ()
- void **publishIMU** ()
- void **publishMagneticField** ()

The documentation for this class was generated from the following files:

- ROSInterface.h
- ROSInterface.cpp