"An Embedded Low-Power Control System for Autonomous Sailboats"

J. Cabrera-Gámez, A. Ramos-de-Miguel, A. C. Domínguez-Brito, J.D. Hernández-Sosa, J. Isern-González, E. Fernández-Perdomo

Instituto Universitario SIANI (www.roc.siani.es),
Departamento de Informática y Sistemas (www.dis.ulpgc.es)
Universidad de Las Palmas de Gran Canaria (www.ulpgc.es), Spain

in: Bars F., Jaulin L. (eds) Robotic Sailing 2013. Springer, Cham. DOI: 10.1007/978-3-319-02276-5_6

BibTeX:

```
@inproceedings{cabrera_gamez_et_al_2014_irsc_2013,
  author="Cabrera-G{\'a}mez, J.
  and Ramos de Miguel, A.
  and Dom{\'i}nguez-Brito, A. C.
  and Hern{\'a}ndez-Sosa, J. D.
  and Isern-Gonz{\'a}lez, J.
  and Fern{\'a}ndez-Perdomo, E.",
  editor="Bars, Fabrice Le
  and Jaulin, Luc",
  title="An Embedded Low-Power Control System for Autonomous Sailboats",
  booktitle="Robotic Sailing 2013",
  year="2014",
  publisher="Springer International Publishing",
  address="Cham",
  pages="67--79",
  abstract="This work presents a small and affordable autonomous sailboat platform designed to be transported and opera
  isbn="978-3-319-02276-5",
  doi="10.1007/978-3-319-02276-5_6"
}
```

An Embedded Low-Power Control System for Autonomous Sailboats*

J. Cabrera-Gámez[1,2], A. Ramos de Miguel[1], A.C. Domínguez-Brito[1,2], J.D. Hernández-Sosa[1,2], J. Isern-González[1] and E. Fernández-Perdomo[1]

[1] Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería (IUSIANI). `E-mail address: jcabrera@iusiani.ulpgc.es`
[2] Dept. Informática y Sistemas, Universidad de Las Palmas de Gran Canaria, SPAIN.

1 Abstract

This work presents a small and affordable autonomous sailboat platform designed to be transported and operated by one or two people without any special means. The sailboat is based on a RC One Meter class vessel equipped with a low power 8-bit microcontroller board and a set of navigation sensors (compass, GPS, wind vane, ...) and a 868 MHz RF module.

It has been designed to serve as a low cost replicable testbed platform for research in autonomous sailing. The embedded control system makes the sailboat completely autonomous to sail a route determined as a sequence of waypoints, adapting its sailing point dynamically to wind conditions. The control system is completed with an off-board base station that permits to monitor and control the boat or defining a new route. The system is characterized by its long autonomy and robustness in case of communication failures.

2 The Sailboat

Autonomous sailboats have a large potential as high speed vehicles of virtually unlimited autonomy for environmental monitoring and sampling. Depending on their net displacement and dimensions, they can accept scientific payloads that maybe too large or power demanding to be integrated in other types of autonomous marine vehicles.

However, the development of autonomous sailboats is complex in terms of needed infrastructure and experimental costs. One way to overcome or reduce these limitations is to resort to a scaled down vessel, following a popular rule

*This work has been partially funded by the Canary Government and FEDER funds under ACIISI ProId2010/0062

among small ship builders that states that the overall cost of a vessel goes proportional to the cube of its length.

In accordance with that vision, in this paper we present a small autonomous sailboat that has been based on a commercial RC boat and low cost or legacy off the shelf components. The motivation behind this approach has been to get an affordable open experimental platform which could serve as test bed for the development of navigation algorithms for sailboats.

2.1 The vessel

The sailboat has been based on a carbon fiber One Meter class vessel with mainsail and foresail (LOA[4]: 100 cm; beam: 24.5 cm; draft: 14 cm; sail area: 0.61 m^2; displacement: 4.3 kg; mast height: 1.6m) and the first prototype has been named *ATIRMA, Autonomous TIRMA* after a memorable canary sailboat.

A ONE Meter Class sailboat was selected because it combines optimally good sailing capacities, cost, ample space under deck with easy access, extra payload capacity and dimensions that ease its operation and transport on a normal car.

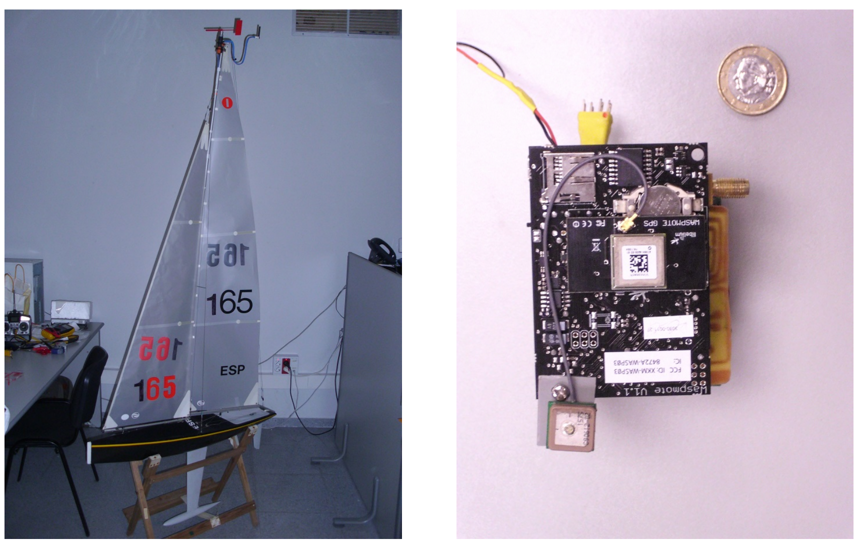

Fig. 1: The ATIRMA sailboat and the on board electronics.

The sailboat employs two analog RC servos as actuators for the rudder and sail's sheet. They are powered from a six NiMH rechargeable AA cells

[4]Length Over All

(1.2V, 2700mAh) connected in series as a 7.4V battery pack with a capacity of approximately 20 Ah. It is equipped with a custom-made wind vane situated on top of the mast for sensing the apparent wind direction but not wind speed.

2.2 The hardware

The vessel's electronics is made up of the following main components:

- An 8-bit microcontroller board
- An XBee PRO 868MHz RF module
- A GPS receiver
- Electronic compass with inclinometers
- Wind vane
- Current sensor

Microcontroller

The sailboat controller is a commercial credit-card size board based on a AT-mega1281 microcontroller running at 8MHz. The microcontroller integrates 8KB of SRAM for data, 128 KB of FLASH memory for program and 4 KB EEPROM [3]. The board provides several UARTs, an I^2C bus, a micro SD card reader, a real-time clock, a three-axis accelerometer and several other sensors for measuring, for example, the board temperature or the battery level.

This board is ready to accept external hardware modules like a GSM/GPRS modem, a GPS receiver or different XBee RF communication modules. It has 5V and 3.3V on-board regulators. It is powered from a 3.7V 6000 mAh Li-Ion battery and consumes 9 mA under normal operating conditions. Suitable photovoltaic panels can be connected directly to the board to recharge the main battery.

RF radio module

For this prototype we have based all communications with the sailboat on XBee 868 Pro RF modules. These modules operate at the 868 MHz ISM band using only one channel. The bandwidth is 24 Kbps and the communications can be encrypted. The nominal range using a 4.5 dB dipolar antena in LOS conditions and free field is 40 km, but more realistic estimations are in the range of 10 km. It is possible to adapt the transmission power in five levels till a maximum of 350 mW. It works at 3.3V and its current consumption is 500mA in transmission and 65mA in reception [1].

GPS receiver

The board is prepared to accept an A1084 20 channel GPS receiver with an external antenna. This receiver is based on the SiRF III chipset and supports the NMEA0183 and SiRF binary serial protocols. We use the binary protocol to configure the receiver (elevation mask, signal strength mask, messages rates, ...) and rely on NMEA RMC and GGA messages for obtaining information about position, altitude, hdop, ground speed, course and time. It has an accuracy of less than 10 meters. It is powered by the on board 3.3V regulators and consumes 26mA [2].

Table 1: Power demands of system components.

Component	Volt(V)	Current(mA)	Power(mW)
Microcontroller	3.3	9	29.7
GPS	3.3	26	85.8
XBee 868 PRO	3.3	65-500	330
TCM2-50	5	20	100
MA3 encoder	5	16	80
ACS712 board	5	7	35
		Electronics total	660.5
	Electronics battery	3.7V - 6000mAh	22200 mWh

Component	Volt(V)	Current(mA)	Power(mW)
Rudder servo	5	10-500	100
Sail winch	5	10-800	500
		Actuators total	600
	Actuators battery	7.4V - 2700 mAh	19980 mWh

Electronic compass

The electronic compass is a legacy TCM2-50 board [4]. Basically, it provides tilt-compensated heading information and instantaneous pitch and roll angles over a RS232 interface. The board temperature and raw readings from three magnetometers can be also obtained. The compass readings are tilt and roll compensated till 50 degrees. The TCM2-50 can't operate for heeling or pitching angles over that limit.

This board is connected to one of the microcontroller TTL serial ports using a simple level converter circuit. The TCM2-50 has a maximum update rate of 20Hz. It is powered at 5V and consumes 20mA.

Wind vane

The wind vane has been custom built from an Optimist wind vane attached to a US Digital's MA3 miniature absolute magnetic encoder [5]. The encoder is installed in an aluminum enclosure with a floating cap on top of the mast and connected to one of the analog inputs of the microcontroller. It allows to detect the direction of apparent wind but not its speed. It works at 5V and consumes 16mA.

Current sensor

The current consumption at the actuator that controls the sails' sheet is measured by means of an ACS712 board [6]. The instantaneous current consumption is read as a voltage at a microcontroller's analog input. The ACS712 board integrates two potentiometers to adjust the intensity range being sensed and the acceptable levels of output voltage. This reading is used as an indirect measure of wind pressure in the sails. It is powered from 5V and consumes 7mA.

A summary of power demands of the main components of the system, along with the capacity of both batteries, is detailed in Table.1. The power consumption reflected in the table for the radio or the actuators are time averages based on laboratory and field measurements under normal operating conditions.

3 Control system

An external base station, a laptop equipped with a XBee USB adapter board, is used to communicate with the microcontroller on board the sailboat over the 868 MHz RF link. Both systems communicate regularly at a predefined but modifiable frequency. Using this radio link, the vessel can be monitored and controlled from the base station.

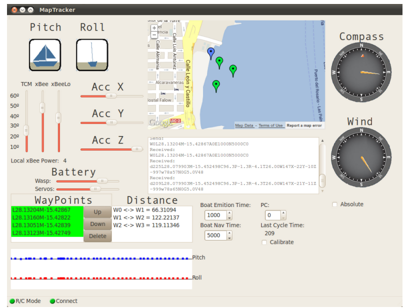

Fig. 2: Graphical user interface at base station.

3.1 Software architecture

The main software elements are the base station control application and the software that runs on the 8-bit microcontroller on board the sailboat. The base station is a Linux application with a Qt front end that relies on the libXBee [7] library to support the radio communications using the XBee radio modules. Using the graphical user interface (GUI) it is possible to add, edit or delete a sequence of waypoints, just by clicking on a Google map (see 2), to define a route for the sailboat.

The interface displays telemetry data received from the sailboat relative to sensor readings or position, bearing and speed of the sailboat. It is also possible to modify some thresholds and parameters like the frequency at which the telemetry packets are remitted or the minimum frequency at which the bearing selection function must be invoked.

Initialization

The initialization of the system is carried out normally with the vessel at shore, but could be done remotely as far as the radio link may reach. In this phase, the operational state of all onboard subsystems are verified and some sensors are calibrated, namely the wind vane and the inclinometers. The calibration steps can be omitted using the base station interface.

First radio communication, SD logging and battery levels are checked and afterwards on board regulators are switched on. Then the GPS receiver is configured to and the elevation and signal strength masks are configured in order to minimize noise in GPS readings. Once the GPS is configured, a first valid fix is awaited and then it will wait 20 seconds to stabilize the GPS measurements.

An optional final stage in initialization deals with the calibration of some sensors offsets. It requires to keep the sailboat in a horizontal position with 0 of pitch and roll and the wind vane pointing forward. This is done only once at the beginning of the experiment but can be avoided and previously calibrated offsets will be recovered from the EEPROM.

At the conclusion of this stage the sailboat will be in remote control mode and it will start sending telemetry data through the radio every 5 seconds by default.

Control loop

Once the initialization has been accomplished, it will start the main control loop that has two possible modes of operation. In the autonomous mode the sailboat's control system selects the best bearing to arrive to the active waypoint. Alternatively, the remote or teleoperation mode permits to take full control of the sailboat from the base station. In both modes the telemetry is kept active.

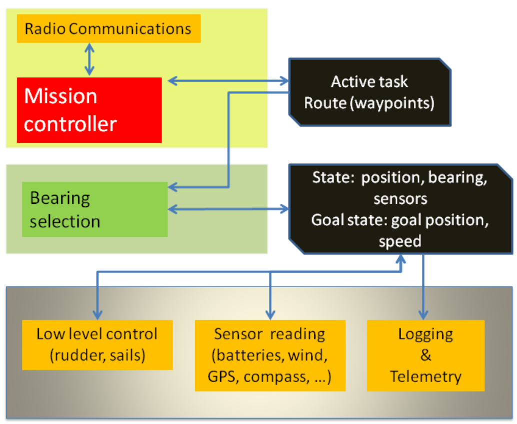

Fig. 3: On-board control architecture

Remote control mode

While the boat is in remote control mode it follows the sail and rudder position commands sent from the base station using a wireless game pad connected to the laptop running the base station application. In this mode, short radio packets are sent to the vessel at a frequency of 10 Hz. The transmission rate can't be too high because in this mode, the on board sensors are sampled, logged and telemetry packets are sent at the specified frequency to the base station.

Autonomous mode

In the autonomous mode the navigation is fully under the control of the on board microcontroller. The control is organized around three levels of control. It is a layered architecture that shares with that presented in [?] a similar assignment of competences to some modules, although the one used here lacks a strategic long term routing module.

At the highest, the route controller simply manages the list of waypoints that define a route and selects the active waypoint. When the sailboat is inside the radius of precision of the waypoint, the route controller will change the active waypoint to the next one in the route. The list of waypoints is treated as a cyclic route by default, so when the last waypoint is reached, it will start again with the first one and the route is repeated.

At the next level, a bearing selection algorithm [10] is used to obtain the best (i.e. fastest) bearing to reach the active waypoint, given the current wind direction and boat position and heading. This control level is runs at an adjustable frequency but can be triggered also by a sudden wind roll. Note that as far we lack an estimation of wind speed on board, we run this algorithm using only the apparent wind.

At the lowest level, a fuzzy controller runs at the highest frequency of the control system and sets the sail and rudder positions to keep the sailboat under control on the bearing determined by the bearing selection algorithm. This controller is an adapted version of the controller described in [11], implemented using the EFLL library [8]. The main difference between the controller described in that paper and ours is that last one's outputs are absolute positions for sail and rudder while Stelzer et al. propose an incremental control system, i.e. outputs of the fuzzy control system are changes to the current settings.

The sailboat can transit into autonomous mode if a prolonged failure of radio communications is detected or because this control mode is explicitly commanded from the base station through a radio packet with the format $WxxLxxMxxAxxExx$. The preamble W identifies this packet as an autonomous mode command packet; the L and the M fields indicate the latitude and the longitude of the active waypoint; A indicates how far (in meters) can be the sailboat off the line that connects the current position of the boat and the waypoint (it is equivalent to the PC parameter in [10]), finally, E indicates the emission period for telemetry messages.

When in this mode, the base station sends every 5 seconds short messages to verify the radio link. If these packets are not received at the vessel for 20 seconds, the active waypoint is deactivated and substituted by the coordinates that identify the "Home Point" and the sailboat will try to arrive to that point autonomously. This constitutes the Return To Home" or RTH behavior that has proved a valuable capability during field tests. This situation can be reverted from the base station as soon as the radio link is reestablished. In that moment, new waypoints and navigation parameters can be transmitted to the boat.

Robust radio connectivity

Loss of radio connectivity is something that may happen easily during sailing due to a variety of reasons and it is important to endow the sailboat with some recovery and continuity strategies to deal with these situations.In order to increase the robustness of radio communications on this uncertain scenario, the XBee radios are used in API mode and all exchanged messages have been limited in extension to make them fit within the payload of XBee API frames. Basically, this constraint reduces the complexity of recovering partially lost packets as all messages involve a single radio frame.

Accordingly, at the lowest level, the XBee radio modules have been programmed to resend automatically dropped or incorrect radio packets for a number of times. The loss of telemetry packets is not critical because they are still logged on the micro SD card available on board. More critical is the loss of command packets sent from the base station and these packets must be acknowledged explicitly from the sailboat. Otherwise, they are resent.

Sensor sampling

Sensors are sampled at different rates depending on its potential rate of change and the temporal cost of a new reading in order to reduce the mean sampling time and hence, the duration of a control cycle.

The GPS sensor available on board has a maximum update rate of 1 Hz and it does not make sense to read it faster because it will deliver old estimates. Even, while reading at the nominal rate of 1 Hz, timestamps of new readings must be checked against the timestamp of the last delivered message to verify that it is indeed a new reading. If that is not the case, a new reading is attempted.

The compass board is programmed to produce a continuous flow of readings at a specific frequency (10 Hz approx.). This approach reduces the cost of reading from the TCM board. Each data packet contains the compass bearing, pitch and roll angles, the temperature of the board and, eventually, an error code. Error codes appear normally when the magnetometers have become saturated or the pitch and roll angle limits have been exceeded. In those cases, these measurements are discarded. Compass packets may accumulate and overflow the microcontroller serial buffer if it is not read fast enough. This is not a problem because the serial buffer is circular and all messages but the last one are discarded. It is important to know that the GPS receiver and the compass are connected to two different serial ports that are in fact multiplexed on the same microcontroller's UART. This implies that it is not possible to receive continuously and simultaneously data from both devices.

The navigation is critically dependent on the adequate sampling rate of the set of on board sensors. With the limited computing power available and high temporal cost of sampling some sensors, a multi rate, smart sampling strategy is necessary. Sensors like the wind vane and the compass have a high update rate and the reading cost is very small. On the other side the GPS has a low update rate and interrogating the GPS receiver takes about 60 msecs. To deal with this situation, two strategies have been implemented. Firstly, fast sensors, and in particular the wind vane, are sampled several times within a single control cycle and filtered to produce better estimates of these magnitudes. Secondly, a Kalman filter is used to produce estimates of the position (lat, lon), orientation and speed. This filter helps to reduce the impact of some noisy GPS positions that may show up sporadically.

Sensors readings are monitored and they may trigger some alarms. For example, a sudden roll in wind direction over a predefined threshold will trigger the execution of the bearing selection algorithm during the next control loop. Also, battery readings are checked against low level thresholds and if low battery alarms are triggered they are notified to the base station.

All collected sensor data are packed and logged on board on a micro SD. A fraction of the logged packet is transmitted to the ground station as a telemetry packet at a predefined frequency. As commented previously, this frequency can be changed from the base station.

With the current hardware, the temporal cost of executing one control cycle is dominated by the temporal cost of the actions carried out during one control cycle. The subtasks that have the higher temporal cost are reading the GPS (60 msec), reading the compass (30 msec) and preparing and logging the telemetry packets (40 msec). Taking into account that some subtasks do not execute in every cycle, the shortest cycle time takes approximately 150 milliseconds and the largest 250 milliseconds.

4 Experiments

Several field trials have been carried out with the ATIRMA on the quiet waters of Las Palmas de Gran Canaria's port bay. An interesting achievement has been the potential power autonomy of the sailboat. We have carried out sailing tests over more that 8 hours in which the sailboat has been sailing continuously and have registered the evolution of remaining capacity. We have never exhausted any of the batteries, even though the evolution of the capacity of each battery was dependent on the experimental conditions (wind intensity and frequency of communications) present during the tests. We plan to test the power autonomy in a close future extensively but currently our rough estimate is that 8 hours of operation under the parameters described in this paper consume approximately 20% of each battery capacity.

The foreseen autonomy exceeds of one day and it could be extended substantially adding supplementary batteries or with the installation of small and lightweight photovoltaic panels that the microcontroller is ready to accept. It must be noticed that with the current setup the servos are adjusted almost continuously and all navigation sensors are always on. If necessary, the implementation of some simple energy conservation strategies could reduce the power consumption even more and extent the autonomy significantly. For example,the winch controlling the sails is responsible for the largest part of the power consumed at the actuators side. A large amount of this power is wasted holding the position of the servo under the pressure of the wind and adjusting the sails to a new position. The substitution of this type of actuator by an actuator that could maintain the position without consuming power would have an appreciable impact in terms of energy conservation.

During these experiments the range of radio communications were tested using on board omni-directional dipole antennae of 4.5 and 0dBi gains. In all cases, the radio link was maintained over the full area of the bay (500 m approx.).

A video of one of the first trials at sea is available from URL [9]. During this video, both control modes, remote control and autonomous, have been exercised. While sailing in open water, away from swimmers or other vessels, the sailboat was on autonomous mode. Remote control was turned on occasionally when it was close to shore and/or bathers had to be avoided.

5 Discussion and conclusions

This work has been motivated by the necessity of developing a small and affordable autonomous sailboat platform that could be transported and operated by one or two people without any special means. In agreement with those objectives, this paper has described the design of a low cost autonomous sailboat whose development has been based on a standard RC One Meter class vessel and off the shelf low power hardware components.

The main features of the described system are its flexibility as experimental platform, its large power autonomy and its robustness in case of communication failures. The amount of space and displacement available in a One Meter class sailboat severely restricts the volume and weight of the sensors and control electronics that can be installed on board. These restrictions have an impact in terms computing power and number and type of the sensors that can be installed on board.

The main limitations of the control system described in this paper are its scarce computing power and reduced RAM memory. These restrictions have demanded a careful analysis and design of the control system to make it ”fit” within the microcontroller memory and processing power, trying - at the same time - to reduce the span of a control cycle as much as possible.

The system described in this paper is very similar in scope to that described in [14], where it was presented a control system for autonomous sailboats based on a 50 MHz (64KB RAM) Cortex-M3 ARM7 processor board.

The main difference between both systems, aside from the smaller computing power and memory of our system, is that our sailing control system is completely embedded on the on board processor, while in [14] the sailboat controller executes in a laptop outside of the boat.

Perhaps the most fragile element of the whole system is the wind vane, as noted by many others [15]. Wind vanes with movable mechanical parts are intrinsically prone to failure. Whilst commercial solutions exist for full scale sailboats, they are unpractical for a boat of small dimensions. Some solutions have been explored and described in the literature but a truly robust design is still to be achieved. An alternative design for a wind sensor (direction and intensity) has been described in [13].

6 Future work

In the near term, future work will address the substitution of the GPS and compass board with more capable and up-to-date versions of these sensors and the incorporation of ultrasound sensors for aerial obstacle detection and avoidance. On the long term, we would foresee to replicate the ATIRMA and tackle the problem of route planning for cooperative surveying by a group of sailboats.

References

1. XBee 868 Pro specifications. `http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/point-multipoint-rfmodules/xbee-pro-868\#specs`
2. A1084 GPS receiver hardware manual. `http://ec-mobile.ru/user\_files/File/Tyco/A1084\_HM\_V1.0.pdf`
3. Libelium's Waspmote manual. `http://www.libelium.com/v11-files/documentation/waspmote/waspmote-technical\_guide\_eng.pdf`
4. PNI's legacy TCM2.5 electronic compass manual. `http://www.pnicorp.com/download/347/99/TCM2.52.6Manualr09.pdf`
5. US Digital absolute encoder MA3. `http://www.usdigital.com/products/encoders/absolute/rotary/shaft/ma3`
6. ACS712 product page. `https://www.sparkfun.com/products/8883`
7. LibXBee library. `http://code.google.com/p/libxbee/`
8. EFLL fuzzy logic library. `https://github.com/zerokol/eFLL`
9. ATIRMA video. `http://www.youtube.com/watch?v=JoCVoFabJMg`
10. Stelzer, R., Pröll, T.: Autonomous sailboat navigation for short course racing. Robotics and Autonomous Systems, **56**, 604-614(2008).
11. Stelzer, R., Pröll, T., John, R.I.: Fuzzy Logic Control System for Autonomous Sailboats. FUZZ-IEEE2007, pp 97–102(2007).
12. Stelzer, R., Jafarmadar, K.: "A Layered System Architecture to Control an Autonomous Sailboat", in Proceedings of TAROS 2007, Aberystwyth, UK.

13. Alvira, M., Barton, T.: Small and Inexpensive Single-Board Computer for Autonomous Sailboat Control, Robotic Sailing 2012, pp 105-116, Springer, 2013.
14. Koch, M., Petersen, W.: Using ARM7 and *mu*C/OS-II to Control an Autonomous Sailboat Robotic Sailing 2011, pp 101-112, Springer 2012.
15. Neal, M. , Sauze,C. , Thomas, B. and Alves, J. C.:Technologies for Autonomous Sailing: Wings and Wind Sensors, in proceedings of the 2nd IRSC , Matosinhos, Portugal, July 6th-12th 2009, pages 23-30, 2009.