

Sauc-Er? An Autonomous Underwater Vehicle

H. L. R. Ong; L. Morgan; R. Bhat; K. Athaide; S. Ahmed; J Gutierrez; D. Mearns; D. Slipper; N. Carter
Department of Engineering, University of Leicester, Leicester, LE1 7RH, United Kingdom

{hlro1, lmm28, rb179, kfa1, sa133, jg104, ddum1, ds151, njc33}@le.ac.uk

ABSTRACT

This paper details the design of Sauc-Er? – an unmanned autonomous underwater vehicle – developed by students of the Department of Engineering at the University of Leicester for the “Student Autonomous Underwater Vehicle – Europe” (SAUC-E) competition.

Keywords

autonomous underwater vehicle submarine acoustics position location vision identification

Abbreviations

6DOF	Six degrees of freedom
ACON	Acoustics controller (Sauc-Er? hardware)
ADC	Analogue to digital converter
API	Application programmer’s interface
APRO	Acoustics processor (Sauc-Er? software)
AUV	Autonomous underwater vehicle
AV	Autonomous vehicle
CF	Compact Flash
CHK	Checksum (Sauc-Er? packet field)
CMD	Command (Sauc-Er? packet field)
CPRO	Communication processor (Sauc-Er? software)
DAT	Data (Sauc-Er? packet field)
DST	Destination (Sauc-Er? packet field)
E ² PROM	Electrically erasable programmable read only memory
GPIO	General purpose input/output
HDR	Header (Sauc-Er? packet field)
I ² C	Inter-integrated circuit
I ² S	Inter-integrated sound
ID	Identifier (Sauc-Er? packet field)
IMU	Inertial measurement unit
MCON	Main controller (Sauc-Er? hardware)
MEMS	Micro electro-mechanical systems
MPRO	Main processor (Sauc-Er? software)
NPRO	Navigation processor (Sauc-Er? software)
PCB	Printed circuit board
PCON	Propulsion controller (Sauc-Er? hardware)
PMON	Power monitor (Sauc-Er? hardware)
PPRO	Propulsion processor (Sauc-Er? software)
SRC	Source (Sauc-Er? packet field)
USB	Universal Serial Bus
VNC	Virtual Network Computing (a software)
VPRO	Vision processor (Sauc-Er? software)

1. INTRODUCTION

Autonomous vehicles (AVs) are fully self-contained mobile platforms that are either remotely controlled or pre-programmed to perform specific tasks. In general, AVs are capable of making decisions – such as maintaining a minimum distance from other object or negotiating a course – based on its sensor readings, without human intervention.

Although AVs are usually synonymous with the military (e.g. the Predator unmanned aerial vehicle), they are also used, or could be used, for geographical exploration (e.g. charting, finding mineral deposits, crop surveying) and environmental studies (e.g. pollution monitoring, ice sheet), to say the least. Hence, AVs have significant scientific and humanitarian benefits as well.

This paper details the design of Sauc-Er?, an autonomous underwater vehicle (AUV) which will compete in the 2006 Student Autonomous Underwater Challenge-Europe (SAUC-E). The rest of the paper gives an overview of Sauc-Er?, describes its hardware and software design, the competition strategy and the testing procedures.

2. OVERVIEW

Sauc-Er? has a hemispherical clear acrylic chassis with its propulsion system, hydrophones and drop target system connected to the exterior, as shown in Figure 1 (not to scale, some parts simplified/not shown for clarity).

The propulsion system consist of two horizontal bidirectional propulsion units situated to the left and right of the body (90° and 270° with respect to its front), and four vertical propulsion units situated at 45°, 135°, 225° and 315°. The horizontal propulsion units allow Sauc-Er? to move in all horizontal directions as well as rotate on the spot (clockwise and anti-clockwise) by controlling the force exerted by each propulsion module. The vertical propulsion units raise, lower and stabilise Sauc-Er?. These have less thrust as compared to the horizontal units.

Sauc-Er?’s two hydrophones are mounted along the rim of its chassis on the same plane as the propulsion system at 0° and 180°. The hydrophones are used to detect the location of the beacon. Its drop target system is mounted at the base, which also serves as a flat resting surface.

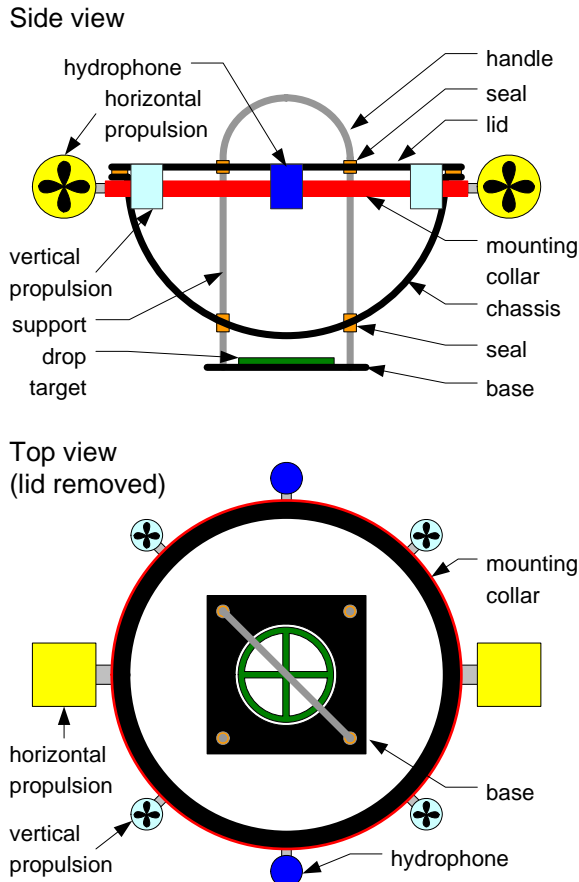


Figure 1: Side and top view (no internal components)

Sauc-Er?'s controllers (main, propulsion and acoustics), motor drivers, imaging system, power supply and navigation system are all located within its chassis, as shown in Figure 2. The hydrophones, propulsion units and solenoids are connected to the acoustics controller, motor drivers and drop target driver respectively via waterproof connectors. The major systems are described in detail in the next section.

3. HARDWARE DESIGN

Sauc-Er?'s design and implementation was divided amongst the members of the team according to their discipline. The team's design paradigm was to minimize mechanical complexity at the expense of speed to improve the chances of successfully completing the tasks. The team purchased off-the-shelf components and ready-made modules, whenever possible, to maximize the time available for software development. In addition, testing and debugging is an integral part in Sauc-Er?'s development right from the beginning. Where feasible, Sauc-Er? is over engineered with at least a 50% safety margin.

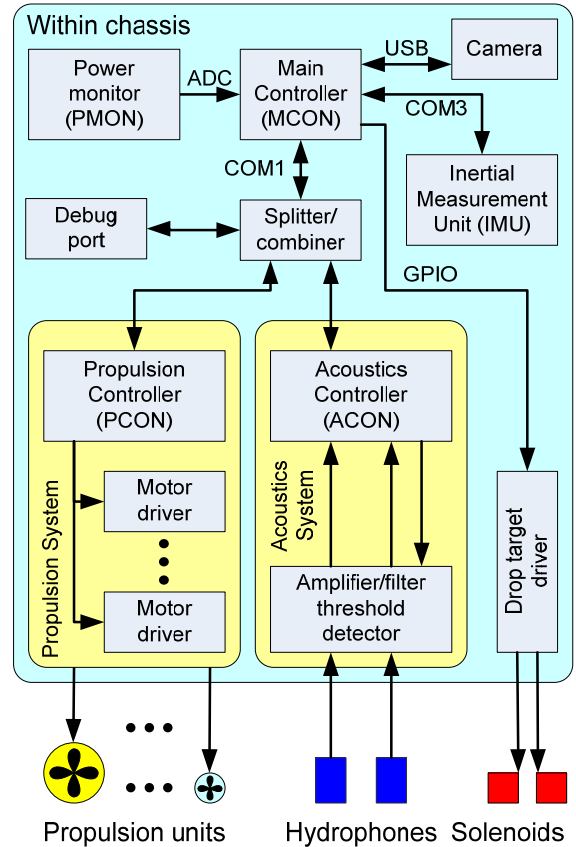


Figure 2: Hardware block diagram

3.1 Requirements

Sauc-Er?'s requirements were set by the team in order to start designing the system, as shown in Table 1.

Table 1: Sauc-Er?'s requirements

Item	Requirement
Minimum displacement	10kg
Minimum top velocity	1m/s
Maximum angular velocity	90°/s
Minimum battery life	30 minutes
Minimum power supply	5V, 2A
Minimum power supply	12V, 5A
Minimum positive buoyancy	1%
Maximum positive buoyancy	10%

3.2 Weight

Sauc-Er?'s displacement is approximately 16.8kg. Ballast (lead weights) are added at the base to improve stability and control its buoyancy to within 10%. This minimises the power required by the vertical propulsion units during

descends and maintaining current depths. The ballast and batteries – the heaviest components – are situated at the base of the chassis to achieve a low centre of gravity necessary for stability.

The estimated weight (maximums) of the various components is shown in Table 3. Sauc-Er? is about 700g lighter after dropping the drop target.

Table 2: Estimate maximum component weight

Device	Weight (kg)
Chassis	1.5
Propulsion units	3.0
Batteries	1.5
Support structure	4.0
Electronics	2.0
Drop target	0.7
TOTAL	12.7

3.3 Chassis

The chassis shape was the responsibility of the entire team with each member presenting a suitable design. The inverted hemisphere was chosen due to the large internal volume, high strength, lowest drag (in all directions except rising) and low centre of gravity.

Sauc-Er?’s chassis is a 400mm diameter, 4mm thick clear acrylic hemisphere with a 20mm flange along the rim. Eight 20mm cube spacers between the flange and chassis wall holds the mounting collar in place. The mounting collar – a 20mm broad metal strip perforated with 4mm holes 12.7mm (1/2 inch) apart that circumvents the rim – is for mounting the horizontal and vertical propulsion units, hydrophones and any other devices that may be necessary in the future. The hole dimension and spacing ensure compatibility with Meccano parts which may be essential for quick fixes.

The chassis lid is a 440mm diameter, 4mm thick, clear acrylic plate bolted to the internal support structure (see Figure 1) and held along the rim with metal C-clips. The propulsion units and hydrophones wires are connected through the lid via waterproof connectors. Rubber gaskets (seals) are used to watertight all the joints.

Sauc-Er?’s estimated drag force with respect to velocity, as shown in Figure 3, was calculated based on a sphere with the following equation:

$$F_d = \frac{C_d \rho v^2 A}{2}$$

where F_d is the drag force, C_d the drag coefficient, ρ the density of water (taken as 1000kgm^{-3}) and A the surface area. The drag coefficient is derived from based on the Reynolds number. Since Sauc-Er? is a hemisphere, Figure 3 can be considered as the upper limit of the drag force likely to encounter.

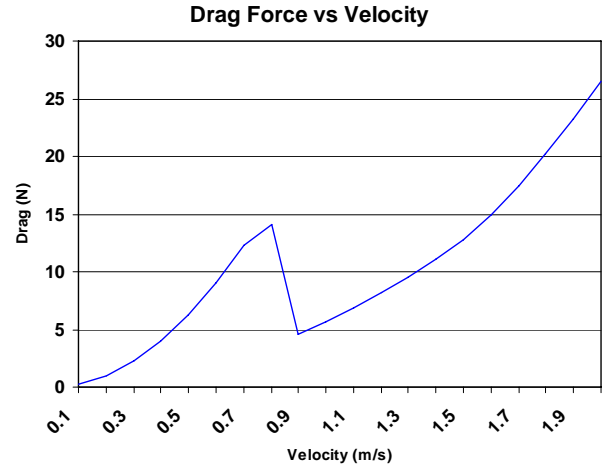


Figure 3: Estimated drag force of Sauc-Er?

The drag force increases to 14N at 0.8m/s before falling sharply to 5N at 0.9m/s due to the transition from laminar to turbulent flow. Sauc-Er?’s actual drag force may be lower than estimated since it is a hemisphere and the units attached (hydrophones, base and propulsion) to the outer surface may induce turbulent flow at lower velocities. Figure 3 also implies that if Sauc-Er? achieves the minimum top velocity of 1.0m/s, it is able to travel for at least 1.5m/s.

3.4 Support Structure

The backbone of the support structure consist of four threaded 6mm diameter metal rods that starts at the base and ends above the lid. These rods – located at the corners of an imaginary 152.4mm (6 inches) square – have rubber seals that tightly clamp the chassis and follow its curvature to prevent leaks. Silicone sealant is also used to reduce the chances of leakage. Sauc-Er?’s handle is connected above the lid to two of these rods.

The vertical rods are traversed with Meccano plates at various heights, as shown in Figure 4. These plates – used to mount/hold the batteries, circuit boards and the camera – are easily adjustable and can be quickly adapted to mount new modules. The lowest plate holds the lead weights used as ballast, in place.

Sauc-Er?’s base is a 200mm square, 2mm thick aluminium plate with a 102mm diameter hole in the middle. The base houses the drop target, which falls

through the hole when required, and the mechanism holding it in place.

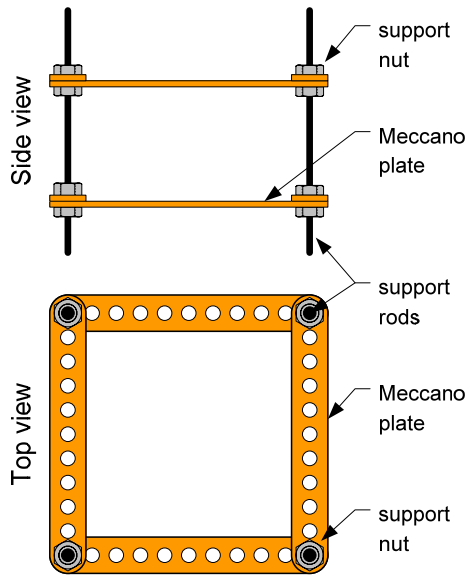


Figure 4: Internal support structure

3.5 Drop Target System

The drop target is a 100mm diameter, 5mm thick circular metal disc with cut outs (see Figure 5), held in place beneath the chassis by the drop target mechanism. The cut outs significantly reduce the drag of the drop target allowing it to fall more or less vertically. The dimple lowers the drop target’s centre of gravity for added stability and also reduces the chances of it sliding off the target at the bottom of the pool.

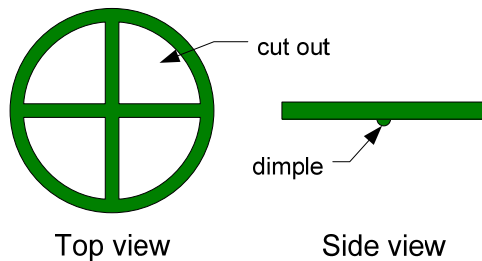


Figure 5: Drop target

The drop target is firmly held in place by two latching solenoids on both sides and a crossbar above it, as shown in Figure 6. The armature of the solenoid which fits into holes in the drop target retracts, when the solenoid is activated, releasing it. This arrangement means Sauc-Er? does not need to be powered to attach the drop target and the solenoid only needs to be briefly powered to release the drop target.

The drop target driver, as shown in Figure 2, is basically a relay being driven by the main controller’s (MCON, see Section 3.12) general purpose I/O (GPIO) via a transistor.

This relay provides galvanic isolation for the solenoids and could be powered from its own batteries since the solenoids are not waterproofed. The entire driver is constructed on its own printed circuit board (PCB) and connected via pluggable terminal block connectors.

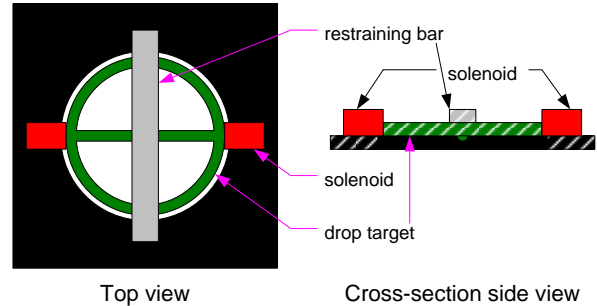


Figure 6: Drop target mechanism

3.6 Power Supply System

The estimated maximum power consumption of Sauc-Er?’s various systems (in Watt-Hours) is shown in Table 3. In reality, the consumption would be significantly less as these are based on a 100% duty cycle.

Table 3: Estimate maximum power consumption

Device	Consumption (Watt-hours)
H. propulsion units	21.6
V. propulsion units	24.0
Main Controller	2.0
Other controllers	0.1
Camera	0.5
Inertial Measurement Unit	0.5
Hydrophones	0.5
Miscellaneous	0.5
TOTAL	49.7

Sauc-Er?’s power is supplied by four 6V 1.2Ah sealed lead acid batteries arranged in a series-parallel formation to give 12V at 2.4Ah (28.8 Watt-hours). Even when operating at its maximum capacity, Sauc-Er? has enough power to last at least 30 minutes. Four lower capacity batteries rather than two higher ones are used to maintain craft stability. The batteries can be recharged *in-situ* by connecting a lead acid charger to a connector on the chassis and setting the power and battery mode switches to “on” and “charge” respectively, as shown in Figure 7.

Sauc-Er? is powered when the power and battery mode switches are “on” and “run” respectively. The 12V output

from the batteries (Vbat) drives the propulsion units and main controller directly (the latter has its own on-board voltage regulators), and is stepped down to 5V for all other devices. This step down is accomplished in two stages. The battery voltage is first stepped down by a DC-DC converter to 6V. A DC-DC converter is used due to their high efficiencies. The second stage regulates the voltage down to two 5V sources with low-noise linear voltage regulators. This arrangement dramatically reduces the high frequency noise generated by the DC-DC converter as well as insulating both 5V sources from each other (“5Vdig” is for digital systems and “5Vana” for analogue systems).

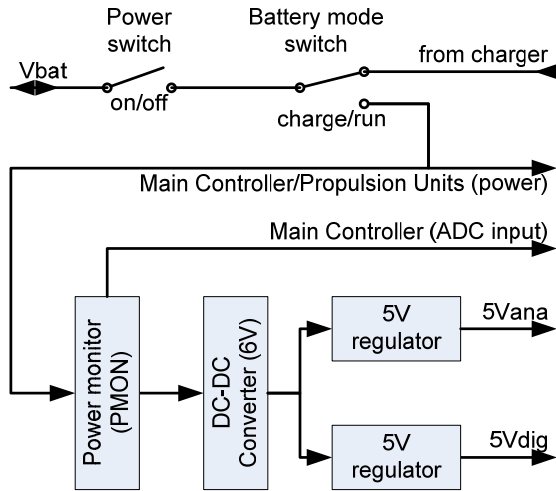


Figure 7: Power supply block diagram

The MCON reads the battery voltage and current drawn with a power monitor (PMON) circuit to ensure Sauc-Er? is operating within safe limits. Current monitoring is achieved by an IC (Zetex ZXCT1009) that proportionally converts the current into a voltage. Voltage monitoring involves creating a 1/4th mirror of Vbat with a potential divider. Both signals are then read by the Main Controller via its analogue-to-digital converter (ADC).

The power supply system is mounted on a single PCB. The batteries, switches and power lines connect to the PCB via pluggable terminal block connectors.

3.7 Propulsion System

Horizontal propulsion is achieved with two 2150RPM (revolutions per minute, no load) motors (MFA/COMO 940D) each connected to a 50mm diameter, 72mm pitch propeller (MOCOM 2024M) and housed in a watertight casing. Each horizontal propulsion unit must generate at least 7.5N of thrust to overcome the laminar flow drag force which peaks at 0.8m/s, as shown in Figure 8 (blue is the drag force, red is the force of one propulsion unit, green is the force for two propulsion units). The maximum propeller slip (i.e. the ratio of the actual linear

distance travelled in one revolution to the propeller’s pitch) possible was 30% before the propeller’s thrust cannot overcome the drag at 0.8m/s.

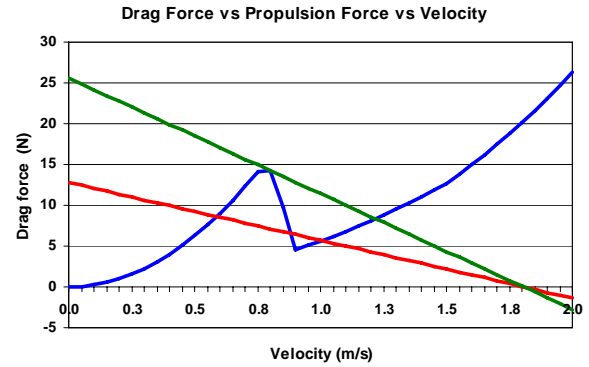


Figure 8: Drag force and propulsions force with respect to velocity at 30% propeller slip

The estimated thrust generated by each propeller shown in Figure 8 does not consider many forms of losses such as cavitation, drag and actual propeller working area, but does consider propeller slip. The following equations were used to derive Figure 8. The subscripts i, p and e denote the inlet, propeller and exit.

$$F_p = \dot{m}_e v_e - \dot{m}_i v_i$$

where F_p = propeller thrust

\dot{m}_x = mass flow rate (x = i for inlet, e for exit)

v_x = velocity (x = i for inlet, e for exit)

Since water is incompressible, $\dot{m}_e = \dot{m}_i$, hence:

$$F_p = \rho v_p A (v_e - v_i) \text{ as } \dot{m}_e = \rho v_p A$$

where ρ = density of medium

v_p = velocity of propeller

A = area of propeller

By assuming that $v_p = (v_e + v_i)/2$, the equation is reduced to:

$$F_p = 2\rho v_p A (v_p - v_i)$$

Vertical propulsion is achieved with four motors (MFA/COMO 918D) each connected to a 39mm diameter, 50mm pitch propeller (MOCOM 1816M) and housed in a watertight casing. The four propulsion unit must produce a minimum of 16.5N ($16.8\text{kg} \times 9.81\text{m/s}^2 \times 0.1$), assuming 10% positive buoyancy, to lower Sauc-Er?. Hence, each motor must produce at least 5N of thrust.

Each propulsion unit (horizontal and vertical) have its own motor driver, which are all identical. The motor driver – capable of handling up to 4A continuously – controls the polarity and voltage supplied (using pulse-width modulation, PWM) to change the propulsion motor’s direction and power output.

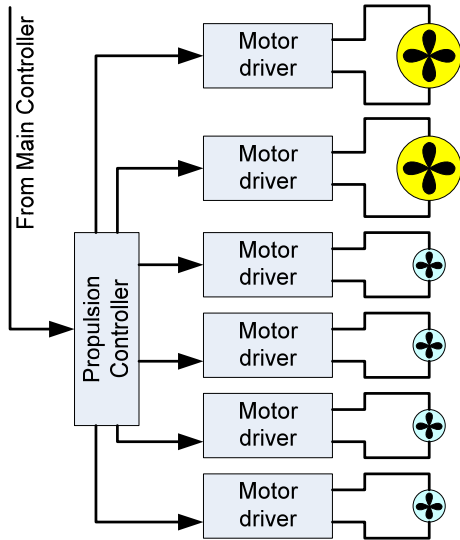


Figure 9: Propulsion controller and motor drivers

The horizontal and vertical propulsion units are controlled by the propulsion controller (PCON), as shown in Figure 9. The PCON is a PIC16F876 microcontroller that is responsible for generating the necessary PWM and direction signals for all six channels according to the commands sent by the Main Controller via the 5V “RS232-like bus” (see Section 3.11). The PWM frequency and duty cycle resolution is 5kHz and 11 steps (0% to 100% in 10% steps) respectively. The PWM’s duty cycle is gradually – rather than abruptly – increased to the desired value when commanded to reduce the wear and tear, vibration and power consumption of the propulsion motors. The propulsion controller reverses the motor’s rotation for a short duration when commanded to stop.

The PCON biases the power supplied to the horizontal and vertical propulsion motors; hence taking their physical differences into consideration. The biasing coefficients are determined experimentally and stored in the electrically erasable programmable read-only memory (EEPROM) built into the microcontroller. When Saucer? is moving forward, the PCON will apply a higher duty cycle to the weaker horizontal propulsion unit. Power biasing is also automatically applied to the vertical propulsion units.

The PCON is an open-loop controller, although power biasing reduces propulsion errors. It receives commands

from the MCON which reads the inertial measurement unit (IMU, see Section 3.8) to form a close-loop system. The commands accepted by PCON are shown in Table 4. These commands, apart from Init, are acknowledged.

Table 4: PCON commands

Command	Description
Init	Initialises PCON (equivalent to a soft-reset)
E_Read	Reads the EEPROM
E_Write	Writes the EEPROM
H_DC	Sets the duty cycle of each horizontal motor (no biasing)
H_Move	Move forward/backward by enabling the horizontal motors (with biasing)
H_Rotate	Rotate on the spot by enabling the horizontal motors in opposite directions (with biasing)
H_Stop	Stop the horizontal motors
V_DC	Sets the duty cycle of each vertical motor (no biasing)
V_Move	Move up/down by enabling the vertical motors (with biasing)
V_Stop	Stop the vertical motors

The PCON and motor drivers have their own PCBs. They use pluggable terminal blocks for quick, reliable connections rather than the more compact multi pole connects which usually requires soldering or crimp tools. The PCON is connected to the splitter/ combiner (see Figure 2) and derives its power from it. The motor drivers are directly powered by the batteries (for the motors) and the “5Vdig” supply (for the driver IC).

3.8 Navigation System

Navigation is by means of a solid state, 6-degrees-of-freedom (6DOF) IMU from Spark Fun Electronics (Sense-6DOF-v2). This IMU consist of three micro electro-mechanical system (MEMS) accelerometers and three MEMS rotational sensors (rate gyroscopes) arranged in an orthogonal manner. The accelerometers and rotational sensors are capable of sensing up to 6g and 150°/sec respectively. It communicates with the MCON via RS232 and is powered by the 5Vdig line.

The MCON reads the IMU’s translational and rotational forces and temperature at 100Hz. The three translational and three rotational forces are converted to absolute translation and rotation with respect to its origin using

quaternions. This algebraic system, which can be implemented as matrices, is less processing intensive as compared to Euler angles and do not suffer from gimbal lock.

Ideally the navigation system requires laser gyroscopes (or equivalent) for accuracy since positional errors accumulates over time. These sensors however, are beyond the team’s budget; hence a compromise between cost and accuracy is taken. Sauc-Er? copes with the limitations in these ways:

- The linear and angular velocities have been limited (see Table 1) to increase the perceived sampling resolution and reduce the chances of the sensors saturating.
- The temperature reading is used to reduce temperature-induced sensor errors.
- Moving Sauc-Er? in predefined paths that return to the origin and determining IMU errors based on the difference in displacement between the start and end positions of the run. The average difference of multiple runs are used reduce the IMU errors.
- Zeroing the coordinate system when Sauc-Er? reaches a target to reduce accumulative errors (assuming the location of all targets are known). All unaccomplished targets will have their coordinates zeroed based on the last accomplished target.

3.9 Vision System

The vision system uses a fixed focal length, USB (universal serial bus) Logitech QuickCam Pro 5000 webcam pressed against the wall of the chassis. Due to its wide viewing angle (approximately 90°) its position is fixed (no servo mechanism) to the internal support with Meccano but manually adjustable.

The webcam supports the USB Video Specifications which allows it to return video data in various formats such as MJPEG, MPEG-2, DV and raw data. Sauc-Er? uses the MJPEG format due to the lower bandwidth requirements and converts it to bitmap before further processing. These are done by the vision processor module (VPRO) as discussed in Section 4.5.

3.10 Acoustics System

The final target of the competition is a 200kHz acoustic sounder (acoustic target) that emits pulse periodically. This signal is detected by the matching hydrophones (from the Eagle Cuda 128 fish finder) mounted at the front and rear (0° and 180°), as shown in Figure 1. The signal from each hydrophone is amplified and low-pass

filtered before going through their respective threshold detectors, as shown in Figure 10.

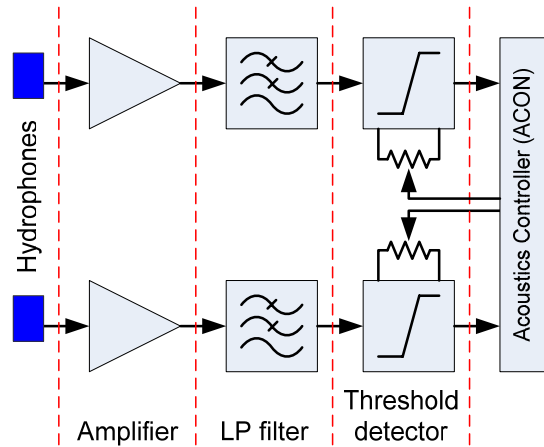


Figure 10: Block diagram of acoustic system

Each threshold detector is a comparator whose trigger level can be controlled with a digital potentiometer by the acoustics controller (ACON). The ACON determines the suitable trigger level by varying the comparator’s trigger level during quiet periods (interval between target’s signals) until it triggers. The ACON then increases the trigger level by a safety margin to reduce false triggering ready for the next incoming signal. The suitable amplifier gain and safety margin would be determined during submersion test at the competition’s facility.

The ACON is a PIC16F876 microcontroller that poll both threshold detectors and starts a 296.4kHz, 8-bit counter whenever a pulse is detected by either of them. This frequency was chosen to give a 5mm spatial resolution since sound travels at 1482m/s in fresh water at 20°C. ACON stops the counter if the other threshold detector pulses, or when the count reaches 100 (the hydrophones are less than 500mm apart), whichever comes first. This window is sufficient to detect the target when it is directly in front of Sauc-Er? (maximum difference in the time of flight of the signal) while providing excellent immunity to echoes (Sauc-Er? has to be within 250mm of the wall to be affected by echoes).

The ACON timestamps the latest valid counter value which can be accessed by issuing a suitable command (see Table 5, these commands, apart from Init, are acknowledged.). In addition, it also identifies the channel that started the counter. Only the latest counter value is stored however, but the timestamp allows the MCON to determine Sauc-Er?’s position when the reading was taken.

The ACON’s counter value represents the difference in flight time of a pulse reaching both hydrophones. The various combinations of counter values are discussed in Section 4.8.

Table 5: ACON commands

Command	Description
Init	Initialises ACON (equivalent to a soft-reset)
E_Read	Reads the EEPROM
E_Write	Writes the EEPROM
A_Threshold	Determines the suitable threshold detector trigger level
A_Read	Reads the latest valid counter value, hydrophone that started the counter and threshold detector trigger level

The ACON is constructed on its own PCB and derives its power from the splitter/combiner, as shown in Figure 2. It uses pluggable terminal blocks for connections.

3.11 Communication System

Sauc-Er? is built upon multiple microcontrollers connected to the main controller to distribute the processing load; hence the communication backbone plays a vital role.

MCON itself has three serial, three USB (two host, one slave), one I²C (Inter-Integrated Circuit) and one I²S (Integrator Interchip Sound) ports. The systems connected to it can be classified as high speed data devices (camera and IMU), medium speed control devices (PCON and ACON), and low speed control devices (PMON and the drop target driver).

The high speed data devices had to be individually connected to the MCON as port sharing may affect their performance, as shown in Figure 2. The webcam connects to one USB host port and the IMU connects to a serial port (COM3). PMON is directly connected to the MCON since the latter has an onboard multi channel ADC. The drop target system is also directly connected to a general purpose input/output (GPIO) pin on the MCON.

The PCON and ACON communicate with the MCON via an “RS232-like bus”. Connecting these systems to a bus has many benefits – especially during the testing stages – such as the ability to observe all the commands issued by the MCON and the replies it receives, control Sauc-Er?’s movement without the MCON (even wirelessly) and the ability to add more controllers without the need to redesign the hardware.

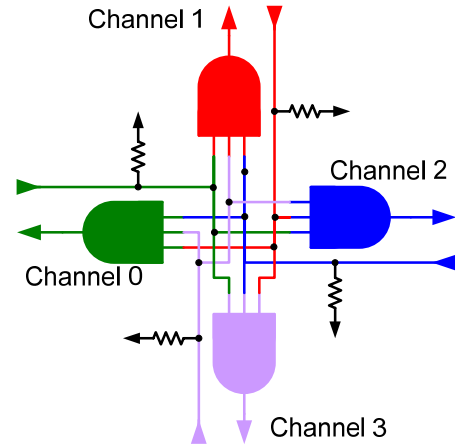


Figure 11: Schematic of a 4-channel splitter/combiner

The “RS232-like bus” is not a true bus: it has to be broken to insert new devices (a splitter/combiner is one). The heart of the “bus” is a splitter/combiner that splits or combines a signal 3 ways (4 channels) using AND gates, as shown in Figure 11. AND gates are used since the serial port level defaults to the high state when idle. When a device connected to – for example – Channel 0 transmits a byte, the low bits (e.g. start bit) will cause the outputs of channels 1, 2 and 3 to go low, hence mirroring Channel 0’s input. The pull-up resistors are vital to prevent the channel inputs from floating, which causes bit errors, when they are not connected to any devices.

The main disadvantage of the “RS232-like bus” is that it does not automatically handle “bus” collisions. This is not a problem in Sauc-Er?’s case, all devices only reply to the MCON when they receive suitable packets.

Table 6: Identifiers for “bus” devices

Device	Identifier (ID)
<i>Broadcast</i>	0
MCON	1
PCON	2
ACON	3
IMU	10
Camera	11
Debug Port	20
<i>Reserved</i>	255

Each device on the “bus” has its own unique byte-wide identifier (ID) stored in the device’s on-board EEPROM at address location 0, as shown in Table 6. The ID value of 255 is reserved since this is the default value for unprogrammed EEPROMs. The ID value of 0 is used as a

“Broadcast ID”, i.e. all devices on the “bus” accept this packet (but will not reply).

The IMU and camera are not connected to the “bus” but had been assigned IDs since the communication bridge (see Section 4.4) can packetise/depacketise packets. Hence, devices connected to the debug port (see Section 3.13) are able to access the IMU and camera.

The “bus” communication has to be packetised to ensure data is routed to the correct device. The 13-byte packet format used in Sauc-Er? is shown in Figure 12.

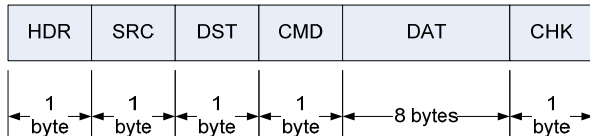


Figure 12: Packet structure

Each packet starts with a unique header byte (HDR), followed by the source device’s ID (SRC, 1 byte), the destination device’s ID (DST, 1 byte), the command (CMD, 1 byte), the data bytes (DAT, 8 bytes) and the checksum (CHK, 1 byte). The header and checksum – a summation of all previous bytes modulo-256 – increases data integrity.

Although it is possible to reduce the packet size by removing the SRC field (all commands come from the MCON under normal circumstances), this is not suitable when it comes to testing and debugging since an external controller connected to the debug port may conflict with the MCON.

The splitter/combiner has its own PCB. Each channel consists of a 4-way pluggable terminal block connector, two for the signals plus one for 5V (digital) and one for ground. By adding the extra power line, devices attached to the splitter/combiner need not require separate power sources. The splitter/combiner’s modularity also means increasing the number of channels simply involves joining one channel from each 4-channel unit together.

3.12 Main Controller (MCON)

The Main Controller is a Keith & Koep Trizeps IV ConXS development board with a 520MHz Intel PXA270 ARM7 microcontroller running Microsoft Windows CE 5.0. This microcontroller’s built-in hardware modules such as serial ports (normal, IrDA, Bluetooth compliant), Ethernet port, USB ports (host and slave), I2C controller, VGA display controller, PWM and GPIOs, makes it ideal for Sauc-Er?.

Sauc-Er? runs on a “headless” (no display) Windows CE image built with Microsoft’s Platform Builder. This image only includes the bare necessities such as an FTP (file transfer protocol) server, a telnet server, memory

card (Compact Flash, Secure Digital, Multimedia Card) access and the relevant C/C++ libraries, to reduce the startup time and processor load. Another image that includes ActiveSync and VNC (virtual network computing, a display emulator) was developed from this image for testing and debugging purposes. The development board’s bootloader provides the user with an easy method to change images.

The ConXS board is directly battery powered since it has its own onboard voltage regulator. A breakout board was made to connect some of the signals (GPIOs, serial ports, etc.) – only available via 0.1 inch/0.05 inch headers – to pluggable terminal blocks for easy, fast, reliable connections. This breakout PCB sits above the ConXS board and uses ConXS’s connectors and headers for support.

The MCON accepts the commands shown in Table 7. These commands – always acknowledged – are used during testing and debugging. The bridge command creates an exclusive link between COM1 and the camera or IMU, as discussed in Section 4.4

Table 7: MCON commands

Command	Description
Init	Initialise Sauc-Er? (equivalent to a soft-reset of the control program)
Bridge	Connect/disconnect COM1 from the camera/IMU
Status	Returns the status of Sauc-Er?

3.13 Debug Port

The debug port is a small 5V module with a Maxim MAX232 IC that converts the “TTL RS232” signals to RS232 levels. This allows a PC or a wireless transceiver to be connected to Sauc-Er?, which is especially important during testing and debugging. The debug port is mounted on its own PCB and uses pluggable terminal blocks as connectors.

4. SOFTWARE DESIGN

Sauc-Er? is controlled by three programs, one each in the ACON, the PCON and the MCON. The latter – the main program that controls Sauc-Er? – is described here.

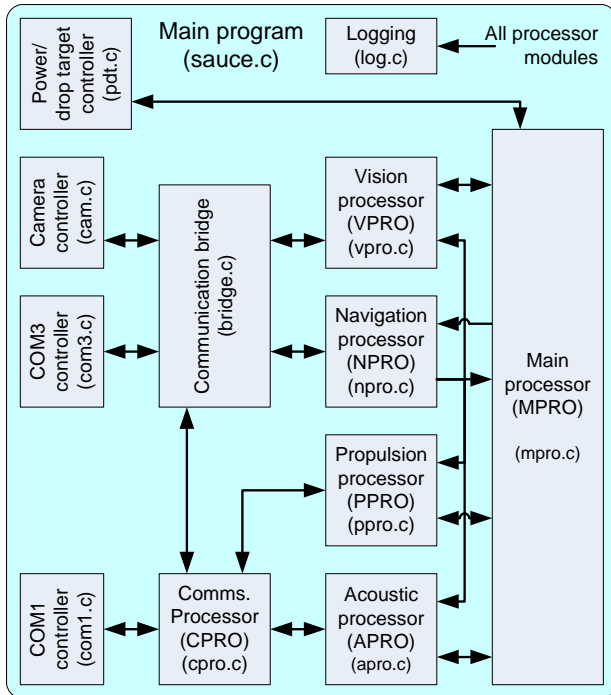


Figure 13: Control program block diagram

Sauc-Er? is controlled by a program written in C running on Windows CE. When Sauc-Er? is powered, Windows CE boots and executes the startup.bat batch file. This behaviour – “hardwired” in the image – allows the team to modify the startup sequence and program(s) without the need to rebuild the image, which takes at least 10 minutes. Startup.bat, sauce.exe (default startup program) and the files that the latter generates are stored on the CF card rather than on ConXS’s FLASH memory for easy modification (the contents of the CF card can also be modified on a PC).

4.1 Container Module

Sauce.exe consists of the main module, sauce.c that acts as the container for the other modules, as shown in Figure 13. All the processor modules (NPRO, VPRO, APRO, PPRO, CPRO and MPRO) are also wrapped with a wrapper function and compiled to form executables (e.g. npro.exe, vpro.exe). In doing so these modules could be individually tested from Windows CE’s command prompt. Although it is possible to call these programs within another (e.g. sauce.exe calling apro.exe), this approach is processing intensive and slow since Windows CE has to be involved in intra-module data transfers.

All the modules have at least an initialising and read/write function as their interface. Sauc-Er?’s container module (sauce.c) is responsible for initialising all the other modules and threading them.

4.2 Interface Controllers

The COM1, COM3, camera and power/drop target controller modules communicate with the “bus”, IMU, camera, PMON and drop target driver respectively via Windows CE, while hiding the intricacies of the Windows’ API (application programmers’ interface) from the processor modules. These modules buffer outgoing and incoming data to/from the hardware where necessary. When data is received, the controller modules raise respective semaphores, which are polled by the processor modules. Double buffering is used to ensure the integrity of read and write access buffers.

4.3 Logging Module

The logging module – not directly involved in Sauc-Er?’s control – is vital for testing and assessing its performance. The logging module buffers and writes timestamped debug information from all processor modules to the CF card in a chronological order. It is responsible for ordering the debug information which may not be received in the correct order due to the multithreaded environment of Windows CE. The debug information is a structured text string which could easily be viewed in a modular and/or chronological order with a custom written program.

4.4 Communication Processor Module

The communication processor module (CPRO) is responsible for packetising and depacketising the packets, checking their integrity, and relaying them to the right processor module. For example, when the acoustics processor (APRO) needs a reading from the ACON, the APRO will send the “raw” command to the CPRO, as shown in Figure 14 (red lines are commands, blue lines are reply). The CPRO packetised the command and assigns the MCON and the ACON’s ID to this packet’s SRC and DST fields (see Figure 12). This packet then travels via the splitter/combiner to all devices connected on the “RS232-like bus”. The ACON processes it and sends a reply packet with the SRC and DST fields being the ACON and the MCON’s ID respectively. This reply packet travels via the splitter/combiner to the MCON’s serial port. The CPRO then depacketise the packet and sends the “raw” reply to the APRO.

The CPRO module works in conjunction with the communication bridge module to accept packets destined for the camera or IMU when it receives the Bridge command (see Table 7). The “bridged” device is selected – and remains so – based on the command parameter value until the “unbridged” command is received (the Bridge command with a different parameter value). When “bridged”, the CPRO depacketise packets from the Debug port and sends the “raw” payload to the camera/IMU, and

packetised “raw” data sent from the camera/IMU to the Debug port.

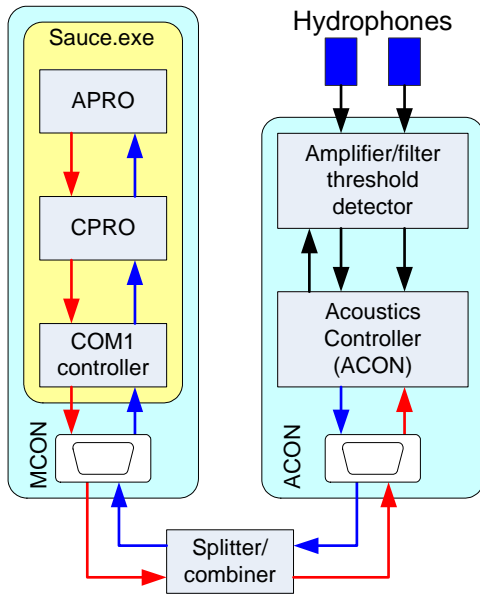


Figure 14: Communication sequence

4.5 Vision Processor Module

The vision processor module (VPRO) is responsible for grabbing and processing a frame once every 0.5s. After saving the grabbed frame in “raw” form in the Compact Flash (CF) memory, the VPRO processes them to determine the location of the visual competition targets.

Image processing starts with the summation of the luminosity of the red and green channels, as shown in Figure 15. The blue channel is not used since the underwater environment is bluish. The resulting image is then blurred (3x3 pixel nonlinear summation) to reduce artefacts due to bubbles and suspended particles followed by applying a sigmoid-shaped transfer function to increase the contrast (difference between light and dark). The last step involves edge detection whereby boundaries are determined based on the difference in luminosity of each pixel with respect to its neighbours. When this difference exceeds a threshold, that pixel is considered an edge: a string of them defines a boundary. The location (center), size, capture time and intensity of all closed boundaries (i.e. forming a closed two dimensional shape) are recorded in the visual record. Only the latest 100 results – accessible to the main processor module – are maintained.

The intermediate processed frames are also stored on the CF card. These information allow the team to analyse the decisions taken by Sauc-Er? to debug and better understand the system.

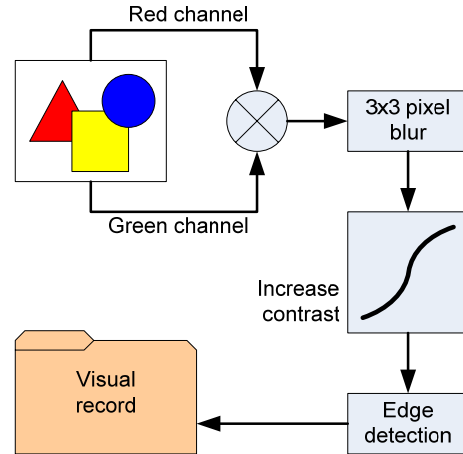


Figure 15: Frame processing flow

4.6 Navigation Processor Module

The navigation processor (NPRO) converts the acceleration and rotational forces into translational and rotational vectors. It then plots the position and orientation of Sauc-Er? in a 3-dimensional Cartesian space as it moves along. Sauc-Er?’s origin (0, 0, 0) is taken as the point that it was turned on.

The NPRO log the coordinates of Sauc-Er? and store the latest 100 readings in a table accessible to the main, propulsion and acoustics processors.

4.7 Propulsion Processor Module

The propulsion processor (PPRO) is responsible for moving Sauc-Er? to the location specified by the main processor. It send commands to the PCON while reading the NPRO as the feedback to its proportional-integral (PI) control system. Sauc-Er? sets a semaphore – polled by the main controller – once it reaches its destination. The PPRO also keeps Sauc-Er? stable and at the desired depth with the information read from the NPRO.

Sauc-Er? travels from its current position to one commanded by MPRO by travelling along the horizontal plane before moving vertically. For example, it first rotates 36.87° anti-clockwise, moves forward 5 units, dives 2 units and finally rotate 36.87° clockwise to get from [10, 4, -2] to [7, 8, -4] when initially pointing along the Y-axis.

4.8 Acoustics Processor Module

The acoustics processor (APRO) is responsible for determining the position of the acoustic target. It sends a read request to the ACON which responds as soon as it receives the latest pulse.

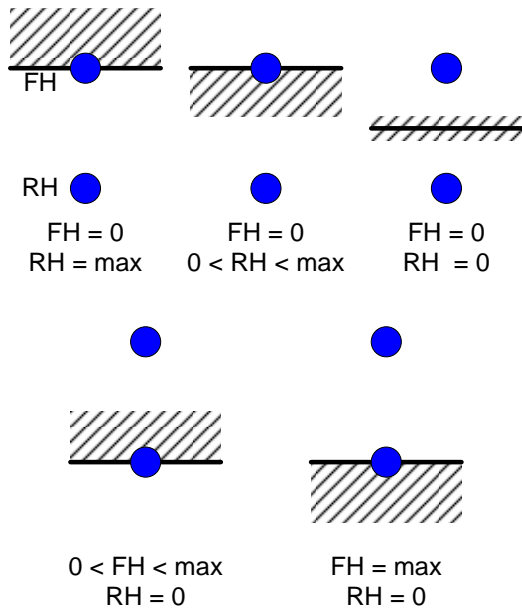


Figure 16: Acoustic target area (shaded) based on hydrophone readings (seen from top)

The APRO estimates the position of each pulse based on the difference between the front (FH) and rear (RH) hydrophones. At any time, one hydrophone will be zero and the other between zero and the maximum possible distance between FH and RH, since the ACON only measures the difference in the time of flight of a signal as discussed in Section 3.10. Hence, the following situations, shown graphically in Figure 16 (target in shaded area), determine the direction (but not distance) of the acoustic target:

1. If $FH = 0$ and $RH = \text{maximum}$, the target is in front of Sauc-Er? but its direction cannot be ascertained.
2. If $FH = 0$ and $0 < RH < \text{maximum}$, the target is somewhere within the volume between the vertical planes perpendicular to the hydrophones. The target is closer to the plane perpendicular to the front hydrophone.
3. If $FH = 0$ and $RH = 0$, the target on the plane which is equidistance from both hydrophones.
4. If $0 < FH < \text{maximum}$ and $RH = 0$, the target is within the same area as that stated in situation 2 but closer to the plane perpendicular to the rear hydrophone.
5. If $FH = \text{maximum}$ and $RH = 0$, the target is behind Sauc-Er? but its direction cannot be ascertained.

For the APRO to determine the direction of the target, Sauc-Er? has to slowly rotate up to 180° while listening.

Once the characteristic fall of one hydrophone from maximum to zero and the rise of the other is detected, Sauc-Er? rotates in the opposite direction to confirm the reading and improve the accuracy. Confirmed readings are recorded in the acoustic record – which stores the 100 newest readings – together with Sauc-Er?'s orientation, position and the ACON's comparator threshold level.

4.9 Main Processor Module

The main processor (MPRO) is responsible for identifying and estimating the location of the targets by applying various algorithms to the visual and acoustic records, and importantly, to execute the competition strategy, discussed in Section 5.

The MPRO reduce false positives in the visual record by analysing contiguous frames in the visual record when Sauc-Er? is still and only accept targets that appear in multiple frames. The accepted results are recorded in the visual target record, which is used by the MPRO when necessary.

The MPRO estimates the location of the acoustic target by triangulating the readings stored in the acoustic record and stores it in the target record as well. It processes readings that are spatially far apart to improve triangulation accuracy.

5. COMPETITION STRATEGY

Sauc-Er? is capable of executing various competition strategies without significant, if any, changes to the software modules apart from the MPRO. Two strategies are potentially feasible: 1) locating all the targets before attempting each of them, and 2) attempting the first target once it has been located even if the next has not been located. The latter strategy, described here, may be more suitable in terms of the competition time and amount of data collected.

The strategy starts with Sauc-Er? first diving to clear the starting gate. This would be a preprogrammed manoeuvre since it is allowed to start immediately behind the starting gate. Sauc-Er?'s starting point will be considered the origin of its location on the 3D Cartesian coordinate space. It continuously record targets once started (i.e. the VPRO and the APRO is working all the time) but the MPRO only uses the recorded information when necessary.

Once passed the start gate, Sauc-Er? will head to the center of the pool, avoiding any targets along the way, before rotating slowly twice (search mode). If the first visual target is not found, it proceeds to another location and carry out another search. This is repeated until the first visual target is found, during which Sauc-Er? should have been able to detect the second visual target and the acoustic target.

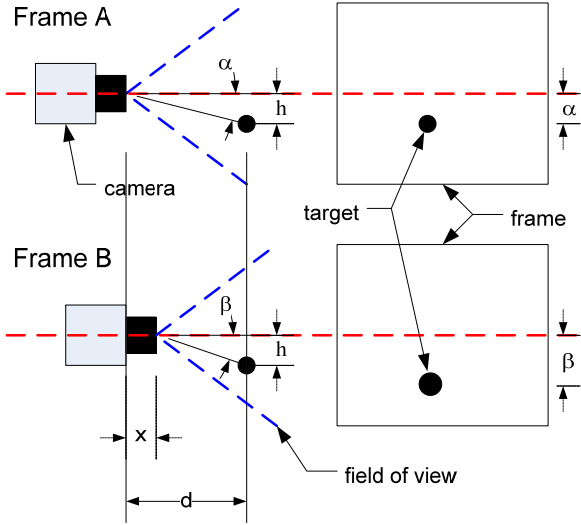


Figure 17: Calculating object distance

When either visual targets is found, Sauc-Er? estimates the distance to the target by analysing contiguous frames while moving horizontally towards it. This is possible since its position in a frame determines its angle (α and β) from the datum, as shown in Figure 17. It is then possible to derive an equation for both frames:

$$\tan \alpha = \frac{h}{d} \text{ and } \tan \beta = \frac{h}{d-x}$$

which can be simplified to

$$d = \frac{x \tan \beta}{\tan \beta - \tan \alpha}$$

as h (distance of the object from the datum) is constant. Since x can be determined from the IMU, the distance the object is from the camera, $d - x$, is calculated.

Sauc-Er? should have at least two hits in its acoustic records after completing the visual targets, which is the minimum required to triangulate the target's location. If

this is not the case, it will have to perform a search at other locations. Once Sauc-Er? reaches the acoustic target, it will surface.

6. TESTING

Testing is central to any successful project which must be developed and started in tandem with the project, rather at a later stage. Hence, every system in Sauc-Er? is tested at every stage of development. The various test that has to be performed is categorised in to three stages: bench test, wet test and competition test; designated as T1.x.y, T2.x.y and T3.x.y respectively, where x is the subcategory and y the test number. These tests are shown in Table 8, Table 9 and Table 10 at the end of the paper.

The tests carried out at each stage is divided into subcategories and designated with second level numbering based on the system being tested (e.g. acoustics system is T1.6.y, T2.6.y etc). This is further refined to the individual tests necessary for each system (e.g. T1.6.1, T1.6.2).

All the tests from one stage should be completed before starting the next stage. Many of the tests listed in each column in each stage could be performed concurrently (e.g. T1.1.1 can be performed at the same time as T1.3.1). A few however (e.g. T2.4.5 must be carried out before T2.8.5), have to be performed in descending order (table-wise). All the tests from one column should be completed before the tests in the adjacent column is carried out. Some cells however (e.g. T1.5.1, T2.2.3) have no test as the systems involved could only be tested after other dependent systems are tested.

7. ACKNOWLEDGMENTS

The authors thank the Ministry of Defence, Herriot-Watt University, the Engineering, Physics and Science Research Council (EPSRC) and the National Oceanography Centre for organizing the competition and for their financial and technical support.

Table 8: Stage 1 tests

Category	T1.x.1	T1.x.2	T1.x.3	T1.x.4	T1.x.5
T1.1.x Chassis	The chassis must be surportable by handle at 15kg without deforming.	The support structure must be securely fastened.	The drop target must be properly released.		
T1.2.x Power System	The voltage outputs must be stable and handle the minimum specified current.	The PMON must correctly monitor the current and voltage.			
T1.3.x Propulsion System	The motor casing must be water-tight. The motor driver must drive motors at various duty cycles in both direction.	The PCON must correctly acknowledge valid packets and ignore corrupted ones.	The PCON must generate the correct output when commanded.	This system must correctly drive the motors when fully connected.	
T1.4.x Navigation System	The IMU must produce the correct inclination/ rotation output.	The IMU drfit error must be characterised.	The orientation algorithms must produce the correct reading.		
T1.5.x Vision System		The webcam must be controllable within Windows CE.	Frames must be captured and saved on the CF card in JPEG format.	Captured frames must be recorded and processed. Boundaries must be identified.	
T1.6.x Acoustics System	The hydrophones must be working and the received acoustic pulse characterised.	The amplifier, filter, threshold detector must be working. ACON must correctly acknowledge valid packets and ignore corrupted.	The ACON must be able to control the threshold detector's trigger level. The threshold detector must trigger correctly.	The ACON must control its counters correctly. It must generate the correct output when commanded.	
T1.7.x Communication Backbone	The splitter /combiner must work on all channels. Unconnected channels must not affect others.	The Wireless transceiver must work on "RS232-like bus". The debug port must be connectable with a PC.	The PC must be able to send commands to MCON, ACON and PCON.		
T1.8.x Main Controller	The Windows CE kernel must be working with serial/USB ports, FTP/telnet server, CF card access.	The MCON must correctly acknowledge valid packets and ignore corrupted ones.	The data log must be correctly written and timestamped. PMON must be correctly read.		

Table 9: Stage 2 tests

Category	T2.x.1	T2.x.2	T2.x.3	T2.x.4	T2.x.5
T2.1.x Chassis	The chassis must be watertight at maximum weight.		Sauc-Er?'s buoyancy must be correct. It must also be stable.	The drop target must be properly released.	
T2.2.x Power System		The batteries and power system must be properly fastened and correctly connected.	The power supply must be stable when powering up each system and all at once.		
T2.3.x Propulsion System	Connectors to the chassis must be watertight.	The propulsion units, motor drivers and PCON must be properly fastened and correctly connected.		Sauc-Er? must move vertically and hover at different duty cycles. The power biases must be calibrated.	Sauc-Er? must move horizontally and rotate correctly at different duty cycles. The power biases must be calibrated.
T2.4.x Navigation System		The IMU board must be properly fastened and correctly connected.	The IMU's position must be calibrated.		The IMU must be working correctly.
T2.5.x Vision System		The webcam must be properly fastened and correctly connected.			The grabbed frames must be stored on the CF card with correct text overlay.
T2.6.x Acoustics System	Connectors to the chassis must be watertight.	The hydrophones and ACON must be properly fastened and correctly connected.		The acoustic pulse must be characterised and the amplifiers, filters and threshold detectors calibrated.	The ACON must correctly choose a suitable threshold and capture acoustic pulses.
T2.7.x Communication Backbone	Connectors for the wireless transceiver must be watertight.	The splitter/combiner must be properly fastened and correctly connected.	The wireless transceiver must be working correctly.		
T2.8.x Main Controller		The ConXS board must be properly fastened and correctly connected.			The MCON must be able to perform close-loop movements with the IMU and PCON with small errors.

Table 10: Stage 3 tests

Category	T3.x.1	T3.x.2	T3.x.3	T3.x.4	T3.x.5
T3.1.x Chassis	The chassis must be watertight at maximum weight and depth.	Fresh silica gel packets must be added.	Sauc-Er?'s buoyancy must be correct. It must also be stable.		
T3.2.x Power System	Batteries must be full charged.	The batteries and power system must be properly fastened and correctly connected.			
T3.3.x Propulsion System	Connectors to the chassis must be watertight at maximum depth.	The propulsion units, motor drivers and PCON must be properly fastened and correctly connected.		Sauc-Er? must move in all directions and hover at different duty cycles. The power biases must be calibrated.	
T3.4.x Navigation System		The IMU board must be properly fastened and correctly connected.	The IMU's position must be calibrated.		Sauc-Er? must be able to run a planned route and return to the origin with minimal errors.
T3.5.x Vision System		The webcam must be properly fastened and correctly connected.		This system must be able to positively identify the location of the competition targets.	
T3.6.x Acoustics System	Connectors to the chassis must be watertight at maximum depth.	The hydrophones and ACON must be properly fastened and correctly connected.		This system must be able to positively identify the location of the acoustic target.	
T3.7.x Communication Backbone	Connectors for the wireless transceiver must be watertight at maximum depth.	The splitter/combiner must be properly fastened and correctly connected.	The wireless transceiver must be working correctly.		
T3.8.x Main Controller		The ConXS board must be properly fastened and correctly connected.		The MCON must be able to deduce which target to attempt first.	Sauc-Er? must be able to carry out all operations autonomously.