Actimetre System

Last update: 2024-05-04

Summary Overview

Actimetre is short for "Activity Metre". It is intended to measure the activity of lab animals, typically rats in cages, using commodity electronic components. For a hardware cost of less than 10€ per cage, the system can record 3 to 6 axes of activity (acceleration XYZ and gyroscope XYZ) at up to 8kHz.

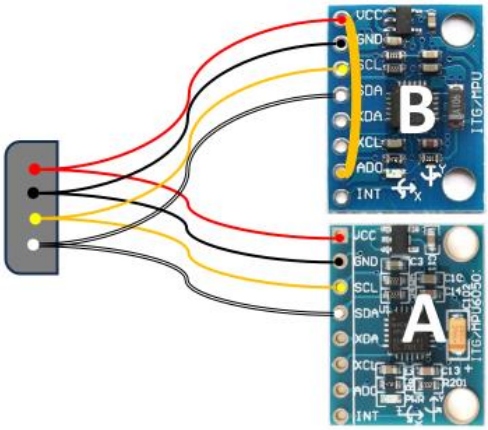
At the base of the system is the MPU-6050/6500 sensor which measures XYZ-axis acceleration and XYZ-axis rotational speed at up to 8kHz. At the next level is a ESP32-S3 which has 2 I2C ports and two execution cores. It can therefore control up to 4 sensors (2 addresses on 2 ports) while communicating over WiFi, connected to the Actiserver, which is an OrangePi Zero2 running Debian Linux. Actiserver also provides the WiFi AP service to the ESP32-S3, as well as being the data repository. It is connected via Ethernet to the central dashboard which helps monitor and manage the system. The whole system can theoretically contain up to 35000 sensors.

The entire system is fairly robust to disruption, except the Central component which is assumed to be highly available. Installation and configuration requires some amount of technical knowledge.

Acticentral is a single central server currently running on a Linux PC. It has a fixed IP address. Actiservers communicate with Acticentral via HTTP.

The MPU-6050/6500 sensor

Datasheet here : MPU-6500 | TDK InvenSense.
Invensense says it's not recommended for new designs, but it's the only 6-axis sensor chip sold on a consumer-grade module at low cost.



The AD0 pin can be connected to either GND (0x68) or VCC (0x69) to control the sensor's I2C address. The 0x68 is called the "A" sensor, the 0x69 unit is "B".

We need four lines (VCC, GND, SDA, SCL) to reach the ESP32. One "A" sensor and one "B" sensor forms a pair, and their cables join in a single JST 2.54 4-pin connector.

The cable also has a USB type-A connector for easily attaching/detaching sensors to the Actimetre.

The ESP32-S3 Mini and Zero modules

The Espressif ESP32-S3 chip is described here: ESP32-S3 Wi-Fi & BLE 5 SoC | Espressif Systems.
Espressif provides a board support library for Arduino IDE.

The ESP32-S3 is available to the public in several module forms, among which: The S3 Mini from Lolin, and the S3 Zero from Waveshare have been tested.

The OLED display

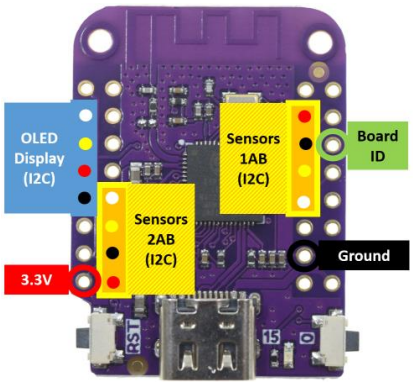
The 0.96 inch 128*64 OLED screen with a SSD1306 controller on I2C interface is a ubiquitous component. It can be found from many places from AliExpress.com. Be sure to use the newer, more compact version with yellow and blue lines.

The Actimetre module

An Actimetre is composed of:

- a ESP32-S3 module (either Lolin or Waveshare)
- a 0.96-inch diagonal 128x64 OLED I2C display
- two 4-pin JST 2.54mm ports for connecting sensors

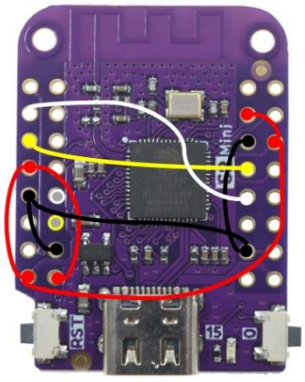
Note that these components are directly soldered on the S3 mini module, so the Actimetre doesn't require additional PCB. Each Actimetre draws less than 250mA at most, so 4 of them can be powered by a low-cost USB 2.0 hub and a 1A power supply.



The display is on the "up" side, which is the side of the module with the USB connector. The connectors to the sensors (I2C, 4-pin JST 2.54) will be on the opposite side. This connector carries VCC, GND, SCL, SDA signals. The display shares the I2C port with Sensor pair "1" (I2C port #0).

There are 9 lines to connect, preferably by wrapping. The JST connectors must be soldered because the legs don't have enough length for wrapping. The screen will be soldered last and will cover all the wires.

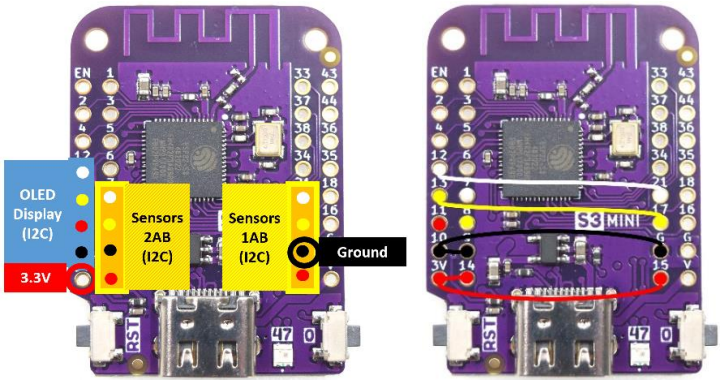
Cabling information for S2x board.



```
          o EN   1 x                    x 40   39 x
    :sda: (i)3   2 x              :vcc:{d}38   37 x
OLED:scl: (j)5   4 x           Sens:gnd:{w}36   35(bd)
Disp:vcc:(ac)7   6 x             #0:scl:{j}34<  33 x
    :gnd:(vz)9   8 o :sda:          :sda:{i}21<  18 x
          x 11  10 o :scl:Sens            x 17   16 x
          x 12  13{v}:gnd:#1           (wz)GN1 GN2 x
        (ab)3V3 14{c}:vcc:               X 15 VBUS X
```

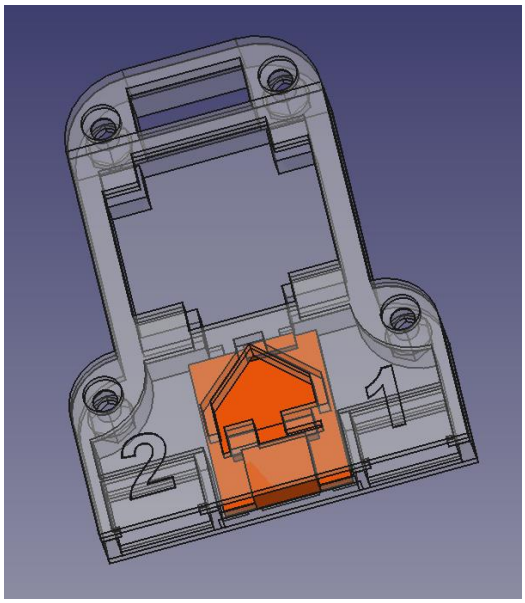
Cabling information for S3i board. The I2C#0 connector's position and orientation are different.

```
          X EN   1 x                    x 33   43 x
          x 2    3 x                    x 37   44 x
          x 3    5 x                    x 38   36 x
    :sda: (i)12  6 x                    x 34   35 x
 I2C:scl: (j)13  7 o :sda:   Port:sda:{i}21<  18 x
Disp:vcc:(ac)11  8 o :scl:     #0:scl:{j}17<  16 x
    :gnd:(yz)10  9{y}:gnd:Port   :gnd:{z}GN1 GN2 x
         (ab)3V3 14{c}:vcc:#1     :vcc:{b}15 VBUS X
```



Newbox

We found that soldering directly on the module made it difficult to service, so a new packaging of Actimetre lays out the components flat in a thin and wide box, and uses the Waveshare ESP32-S3 Zero for smaller footprint.



Actiserver

The Actiserver can be any Linux-based SBC (single-board computer) with both Ethernet and WiFi. It must support WiFi AP (which some USB dongles don't). 1GB of RAM is recommended. The software has been tested on Raspberry Pi3 (1GB), Pi4 (2GB), CM4102000; NanoPi NEO3 (2GB) with a RTL8812 USB WiFi dongle; Orange Pi Zero2 (1GB) and Zero3 (1GB). The Orange Pi Zero3 (1GB) has the best cost/performance ratio and supports an external antenna for better reception.

Actiserver software was written in Kotlin using IntelliJ IDEA. It relies on OpenJDK 17's JVM and therefore is portable across a wide range of systems.

Communication protocols

Communication between Actimetre and Actiserver is a simple socket connection. The Actiserver acts as WiFi AP, but doesn't provide a bridge to its Ethernet connection. Since an Actimetre will know the Actiserver's local IP address (192.168.4.1 by default), it can address it directly. Actiserver software listens on port 2883 at that address, and accepts socket connections. If a socket connection is broken for any reason, the Actimetre reboots by itself, and Actiserver knows the Actimetre is down.

Once a socket is established, Actimetre starts the communication by sending an initial 13-byte packet containing its board type, its MAC address, what sensors are present, and the SW version.

Byte no.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Board type ASCII character 0							
1	Board type ASCII character 1							
2	Board type ASCII character 2							
3	MAC address byte 0							
4	MAC address byte 1							
5	MAC address byte 2							
6	MAC address byte 3							
7	MAC address byte 4							
8	MAC address byte 5							
9	2B is 6500	2A is 6500	2B present	2A present	1B is 6500	1A is 6500	1B present	1A present
10	Version string ASCII character 0							
11	Version string ASCII character 1							
12	Version string ASCII character 2							

Actiserver responds with the unique Actim ID that is assigned to this Actimetre (two bytes, MSB first), as well as the current UTC time in 32-bit epoch seconds since 1970 (four bytes, MSB first). This allows Actimetre to timestamp all its data.

Once Actimetre enters its main loop, it sends a data packet at the sampling frequency it is set to. The default frequency is 1kHz for the S3. This frequency can be changed by pressing the button on the

Actimetre. A data packet is composed of a 8-byte header containing the millisecond timestamp and the sampling frequency, followed by payloads of 12 bytes per measurement. The payload is made of 16-bit signed integers for each of the sensor's output: Accelerometer XYZ axes and Gyroscope XYZ axes.

Byte no.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Seconds since boot, bits 16-23, see [Note3]							
1	Seconds since boot, bits 8-15							
2	Seconds since boot, bits 0-7							
3	I2C port	I2C address	Number of samples in payload					
4	RSSI (0~7)			Sampling mode		Frequency code		
5	is 6500			[Note4]	Microseconds, bits 16-19			
6	Microseconds, bits 8-15							
7	Microseconds, bits 0-7							

- Seconds since boot, this 24-bit unsigned value maxes at 0xFEFFFF (decimal 16711679, representing about 193 days). The Actimetre does a cleaning reboot after 183 days.
- Sampling mode: all axes (0 or 3), only accelerometers (1), only gyroscopes (2).
- Frequency code: 0=100, 1=500, 2=1000, 3=2000, 4=4000, 5=8000
- Microseconds is taken for the last sample of the payload. So if the payload contains 10 samples (byte 3 bits 0-5 contains 10), the time of the first sample of the payload would be 9 sampling periods earlier than this.

That data is decoded by Actiserver who appends each reading in a text format to the appropriate file in the repository. Data repository is in `/media/actimetre` and the file name encodes the Actim ID as well as the start time of the data capture. Actiserver can be configured to start a new file when a certain size is reached (default 1GB) or when a certain time has elapsed (default 24 hours). See the installation instructions for details on how to configure Actiserver.

Actiserver, when it boots up, seeks connection to Acticentral, which is at `actimetre.fr`. Acticentral provides a HTTP CGI entry point, `https://actimetre.fr/cgi/acticentral.py`, that is used for all communication. Note that there is no open back-channel to Actiserver. All communication from Acticentral to Actiserver is handled in the response message to GET or POST requests.

Request	Request type	URL parameters	POST data	Response
Actiserver active (deprecated)	POST	`action=actiserver` `serverId=`*server ID*	Full data on status of Actimetres connected to this Actiserver	Registry data
Actiserver active (current)	POST	`action=actiserver` `serverId=`*server ID*	Full data on status of Actimetres connected to this Actiserver	`OK` or [Note1]
Fetch Registry	GET	`action=registry` `serverId=`*server ID*		Registry data
Fetch Projects DB	GET	`action=projects` `serverId=`*server ID*		Projects data
New Actimetre	GET	`action=actimetre-new` `mac=`*MAC of Actimetre* `version=`*SW version of Actimetre*		Unique (based on MAC) Actimetre

		`serverId=`*server ID* `bootTime=`*boot time of Actimetre*		ID assigned to this Actimetre
Actimetre offline	GET	`action=actimetre-off` `serverId=`*server ID* `actimId=`*ID of Actimetre*		None
Actimetre removed	GET	`action=actimetre-removed` `serverId=`*server ID* `actimId=`*ID of Actimetre*		None See [Note2]
Actimetre report	POST	`action=report` `serverId=`*server ID* `actimId=`*ID of Actimetre*	Plain text message as sent by Actimetre	OK See [Note3]

The full data Actiserver sends in the POST request contains information on itself, including: its configured server ID, its machine type, SW version number, and disk status. It also contains information on all the Actimetres currently connected to this server, including: MAC, board type, SW version, last boot time, sensor configuration, sampling frequency, disk usage, RSSI and connection quality. This information is used by Acticentral to display the Dashboard.

[Note1] Response `!` means Registry or Project data have changed, and Actiserver needs to fetch new data using Fetch Registry and Fetch Projects queries. Response `+A:C` with A=actimId and C=byte to send command C to specific Actimetre. Command 0x10 to simulate button press, 0xF0 to reboot. 0x20 to clean up Actimetre data by calling SYNC_EXEC on remaining data files.

[Note2] This query is issued after all data files for the given Actimetre have been synchronized, i.e. no more data files exist on this Actiserver.

[Note3] When Actimetre sends a Report packet (instead of a data packet), Actiserver relays this information to Acticentral so that the latter can display it on the Dashboard. A Report packet is flagged by the value 0xFF in byte 0, and bits 0-5 of byte 3 contains the number of bytes of payload to interpret as text. Other values in the header are irrelevant.

[Note4] This bit indicates the packet is a Detailed Report message. Bytes 0-2 and 5-7 are accurate, as well as bits 6-7 of byte 3. Byte 4 is meaningless. Bits 0-5 of byte 3 is the size of message, divided by 4. This size includes the null-terminator, which is guaranteed to be present, and zero-padded to the 4-byte boundary.

The Acticentral Registry contains the complete list of all Actimetres known to the system, with their corresponding MAC address and unique ID. This data is JSON-encoded and sent to Actiserver as a response string to the POST request. The Projects database is a dictionary matching actimId with projectId. Actiserver needs this information to store data files in the proper subdirectory.

Acticentral publishes monitoring information on its HTTP server at `https://actimetre.fr`. This data is refreshed every 15 seconds, based on the latest data received from the Actiservers. Note that Acticentral does not store any sensor data.

Acticentral software is mostly a large Python 3.9 script that is called via HTTP CGI for handling requests from Actiserver and actions on the HTML page. The same script is also triggered by a systemd timer for periodic refreshes. HTTP is served by Apache HTTPD 2.4.

Appendix

Recap total BOM for 40-sensor system with 20 Actimetres and 4 Actiservers

Prices are taken from the lowest found on AliExpress.com, including delivery to France. Unit: EUR(€). Acticentral cost is not taken into account.

Component	Cost for system	Cost per sensor	Note
Sensor (MPU-6500)	60.20	1.51	In batches of 10
I2C Cable (2x1m 4P)	19.04	0.48	1m per sensor
USB type-A pair	18.24	0.46	1 pair per sensor
S3 Zero module	125.32	3.13	
0.96" I2C OLED display	29.00	0.73	1 per Actimetre
JST 2.54 4P connectors	4.11	0.10	1 pair per Actimetre
Power supply (1A)	5.83	0.15	1 per 4 Actimetres
USB 2.0 hub 4-port	8.95	0.22	1 per 4 Actimetres
USB cable (A to C)	17.48	0.44	1 per Actimetre
Zero3 (1GB)	84.40	2.11	5 units (1 per 4 Actimetres)
64GB microSD (A1 spec)	24.00	0.60	
Power suppy (2A)	10.00	0.25	1 per Actiserver
USB cable (A to C)	7.85	0.20	1 per Actiserver
APX to SMA cable	5.01	0.13	1 per Actiserver
External antenna	10.35	0.26	1 per Actiserver
Total cost excl. assembly	429.78	10.74	

Limitations and Future work