ROMA TRE
UNIVERSITÀ DEGLI STUDI

CORSO DI STUDIO IN
INGEGNERIA ELETTRONICA
Laurea in Ingegneria Elettronica

Sensor data acquisition and management in a star tracker for micro and nano satellites

Candidate
Simone Tibollo

Advisor
Prof.ssa Federica Battisti

Supervisor
Ing. Fabrizio Bernardini

Second advisor
Prof. Vincenzo Ferrara

Rome, December, 2019
Academic Year 2018/2019

Index

Abstract (Italian)

Il progetto oggetto di questa tesi è realizzato in collaborazione con il DIET (Dipartimento di Ingegneria dell'Informazione, Elettronica e Telecomunicazioni) dell'Università Sapienza di Roma.
Una prima fase di studio è stata condotta dal Dott. Ing. Sara Scibelli ed è oggetto della pubblicazione "Low-cost stellar sensor for attitude control of small satellites”, presentata al 41st PhotonIcs and Electromagnetics Research Symposium.
Il progetto consiste in un sensore di assetto autonomo, a basso costo, a basso consumo e massa ridotta, che non richiede risorse di calcolo dal computer di bordo del satellite.
Il dominio di applicazione è costituito da piccoli satelliti (micro e nano-satelliti) la maggior parte dei quali non utilizza una misurazione e un controllo dell’assetto di precisione, a causa dei requisiti di costo, massa, volume ed energia eccessivi, difficili da soddisfare in generale quando i satelliti hanno tali piccole dimensioni.
La proposta di progetto prevede l'integrazione di due sensori, in particolare uno Star Tracker basato su APS (Active Pixels Sensor) e un sensore giroscopico di tipo MEMS (Micro Electro-Mechanical Systems), integrati in una singola unità che può essere utilizzata in modalità singola o combinata. Il sensore stellare consente di determinare l’assetto satellitare relativo e assoluto, operando come sensore primario a bassa velocità angolare, mentre per alte velocità angolari verrà utilizzato il giroscopio.

Nell’ambito di questa tesi, partendo dallo studio effettuato in precedenza e dai requisiti di progetto, è stato effettuato uno studio e dimensionamento dei componenti selezionati, seguito da una fase di progettazione.
In particolare sono stati utilizzati un microcontrollore Microchip PIC32, un sensore CMOS APS da 0.3 Mp prodotto dalla ON Semiconductor e un sensore giroscopico MEMS della ST Microelectronics.
Durante questa fase, a partire da uno schema architetturale ad alto livello, sono state progettate le diverse sezioni circuitali.

Lo stadio di alimentazione prevede un'unica tensione di 3.3V, compatibile con tutti i dispositivi che fanno parte del sistema, con una corrente massima assorbita di 225mA. È stato dimensionato un convertitore switching di tipo step-down da 500mA massimi, con un ripple inferiore a 50mV. La scelta di un convertitore di tipo switching è volta a consentire un ampio intervallo di tensione in ingresso (4.8-50V), limitando però la dissipazione termica, che rappresenta un problema importante in assenza di atmosfera.

Per quanto riguarda il microcontrollore è stata effettuata una mappatura dei segnali utilizzati, sulla base della disponibilità e della compatibilità delle interfacce a disposizione, prevedendo un'interfaccia parallela per l'acquisizione del sensore CMOS, accompagnata da alcuni segnali digitali di controllo e da un'interfaccia seriale (I^2C) dedicata alla configurazione dei registri e una seconda interfaccia seriale (SPI) per la lettura del sensore giroscopico.

È stato sfruttato un modulo PLL (Phase-Locked Loop) interno al PIC32 per generare un clock necessario al sensore CMOS, configurabile tramite firmware.

È stata poi prevista un'interfaccia esterna, chiamata DATA I/F, utilizzata sia per lo scambio di dati con il computer di bordo (OBC), sia a scopo di test. È possibile configurare questa interfaccia con 3 diversi protocolli (UART, I^2C o SPI), per massimizzare la compatibilità con i diversi sistemi.

Una seconda interfaccia esterna, definita DEBUG I/F, consiste in una porta ICSP (In Circuit Serial Programming) necessaria per programmare il microcontrollore. In particolari configurazioni (con un hardware compatibile) potrebbe essere possibile utilizzare questa interfaccia anche per riprogrammare il microcontrollore durante la missione, in casi di criticità. Questa caratteristica consente di ottenere un'alta affidabilità, anche in assenza di ridondanza hardware.

Successivamente sono state progettate le porzioni di firmware per il microcontrollore, necessarie per la configurazione iniziale (gestione dei clock di sistema e configurazione delle interfacce) e per l'acquisizione dei sensori utilizzati.

A partire dal circuito progettato si è sviluppato un circuito stampato, in tecnologia SMD (Surface Mount Device) utilizzato per assemblare un prototipo in laboratorio, che è stato oggetto di diversi test preliminari.

Oltre al modulo principale, di dimensioni 50x50mm, è stata progettata anche una scheda adattatore con un fattore di forma e un bus PC104 compatibili con lo standard CubeSat. Questo modulo, oltre ad offrire la possibilità di selezionare i segnali da utilizzare, in base alla disponibilità sul computer di bordo, riporta le diverse interfacce su connettori standard JST, utili sia per l'integrazione in micro e nano satelliti che non utilizzano un bus PC104, sia nelle fasi di test e progettazione.

Nonostante le dimensioni, la massa e i costi contenuti, il sistema qui presentato ha dimostrato avere prestazioni confrontabili con le soluzioni esistenti in commercio.

In particolare, le criticità legate ai tempi di lettura del sensore ottico risultano trascurabili, in quanto questo verrà utilizzato per velocità angolari inferiori a 8.75mdps (millesimi di grado al secondo), che corrisponde alla minima sensibilità del sensore giroscopico.

Gli sviluppi futuri di questo progetto, oltre ad una eventuale ottimizzazione della lettura del sensore ottico, per migliorarne le prestazioni, prevedono l'implementazione di un algoritmo di tracking per utilizzare le immagini acquisite e ricavare i dati di accelerazione angolare e assetto relativo.

Sfruttando la memoria EEPROM disponibile sul microcontrollore (2048KB), potrebbe essere possibile anche memorizzare un database delle principali stelle note per determinare l'assetto assoluto del satellite.

1 Introduction

The project subject of this thesis is carried out in collaboration with the DIET (Department of Information Engineering, Electronics and Telecommunications) of the Sapienza University of Rome.

A first phase of study was conducted by Dott. Ing. Sara Scibelli, with a master's degree from Sapienza University of Rome, and is the subject of the paper "Low-cost stellar sensor for attitude control of small satellites", it was also presented at the 41st PhotonIcs and Electromagnetics Research Symposium [1].

The project aims to design a low-cost, low-consumption, low-mass stand-alone attitude sensor, that requires no computational resources from the satellite's On-Board Computer.

The application domain includes small satellites (micro and nano-satellites), most of which do not have a precision attitude determination and control, due to excessive cost, weight, volume and energy demand requirements, hard to meet in general when satellites have such small dimensions.

The project proposal involves the integration of two sensors, in particular a Star Tracker based on APS (Active Pixels Sensor) and a MEMS (Micro Electro-Mechanical Systems) gyroscopic sensor, integrated into a single unit that can be used in single or combined mode. The star sensor can be used to determine both relative and absolute satellite attitude, operating as the dominant sensor at low angular rate, while for high angular rates, the gyroscope will be used.

During the first design phase carried out by Dott. Ing. Sara Scibelli, with Ing. Fabrizio Bernardini and Prof. Vincenzo Ferrara of La Sapienza University, the foundations of this project have been laid and a general architecture for the sensor has been defined.

The needs of a small satellite in terms of attitude determination and control are simpler than those of larger satellites. This is also due to the fact of the lower resolution of the payloads, or the very different operational requirements.

After deployment, many satellites require to quickly damp the rotational rates and for this purpose a MEMS-based gyroscope sensor can be used. It is useful for

quickly and accurately estimate angular velocities. In this way the control logic and the actuators can reach a more precise and faster aiming.
A gyroscopic sensor can also be used for those satellites that use cold gas attitude control (in particular in the event of a jet failure, which would generate excessive speeds) and for those that need to aim quickly in different directions. The satellites that also carry a propulsion system for the regulation of the orbit can surely use a gyroscopic sensor.
During nominal operations, knowledge of a precise attitude with reference to the celestial sphere may be necessary in different missions. At very low rates (pointing or slow drifting of the attitude) the gyro sensors cannot be used with good results and an optical sensor can offer a good solution, assuming of course sufficient angular separation from the Sun or other bright bodies.
The most difficult condition is for Earth orbiting missions, in which, for example, optical imaging is performed in daylight and therefore one side of the sky is occupied by the planet and the other half is partially affected by the Sun. A small sensor like the one proposed can be arranged in such a way to satisfy mission requirements, by proper analysis of the attitude scenarios.
A first possible application of a stellar optical sensor is to provide images of the sky in parallel to other payload operations (for example, acquisition of images with other purposes). These images, or the star positions on the sensor processed from them, can be used during ground post-processing to correlate the attitude with the data acquired by the payload.
A second application of the optical star sensor is to provide the relative or absolute attitude tracking. In relative tracking mode, the star tracker can be used to provide deviation from an initial attitude by tracking the movements of stars over the sensor. This is the simplest and most effective way to use the sensor. In absolute tracking mode the stars patterns acquired by the sensor are used to determine the attitude of the satellite, providing useful on-board knowledge for particular missions.
The overall aim of this sensor is to provide attitude, delta-attitude and attitude rates to a satellite On-Board Computer, thus freeing the latter from the need to implement these tasks.

2 Field of application

2.1 Satellites classification

During a first analysis of this project, a study was carried out on the field of application of this sensor, in particular the standard features of micro and nano satellites and of the attitude detection and control functions.

Below is a classification of small satellites according to their weight [2]:

- Small satellite: from 100 kg to 500 kg
- Micro satellite: 10 kg to 100 kg
- Nano satellite: 1 kg to 10 kg
- Pico satellite: 0.1 kg to 1 kg
- Femto satellite: less than 100 g

While the pico and femto satellites are not suitable for containing an optical attitude sensor, small class satellites can also be compatible with high-end star trackers, characterized by considerable performance, but also by a large size and mass.

The field of application of this sensor will therefore be concentrated on the micro and nano classes.

2.2 Cubesats design

CubeSat standard is particularly interesting and is used as a reference for this sensor design. It is a platform with a cubic structure of side 10 cm. In addition to the 1U cubic format (10x10x10cm), it is possible, by increasing one of the 3 dimensions, to design CubeSats with 1.5U (15x10x10cm), 2U (20x10x10cm), 3U (30x10x10cm), and so on [3].

These satellites are commonly released with a system called Poly-Picosatellite Orbital Deployer (P-POD), which consists of a sled installed on the launcher, which in one instant releases one or more CubeSats in series. Protrusions beyond the maximum dimensions are allowed by the standard specification, to a maximum of 6.5 mm beyond each side. Any protrusions may not interfere with the deployment rails and are typically occupied by antennas and solar panels [4].

Figure 1 - 1U CubeSat CP1 (left), 3U CubeSat CP10 (right) [Cal Poly]

2.3 Attitude Determination and Control Systems

The attitude is the orientation of a body fixed coordinate frame with respect to an external frame. The Attitude Determination and Control System (ADCS) is made of two different parts: *attitude determination* and *attitude control* [5].

Attitude determination refers to the process of measuring and determining spacecraft orientation. Different type of sensors able to measure the satellite's attitude are presented on the next paragraph.

Absolute attitude determination refers to identifying the orientation of the space with reference to an external reference system, typically the J2000 celestial sphere.

Relative attitude determination refers to identifying the orientation of the space with reference to previous attitude arbitrarily chosen as starting orientation.

In both cases attitude determination can be performed on-board, that is computed and used in real-time in the spacecraft, or be performed in post-processing, as results of on-ground computations from raw data generated on-board the spacecraft.

Attitude control refers to the process of orienting the spacecraft in the given direction. It is accomplished using a wide variety of techniques. The requirements for pointing accuracy, stability, and maneuverability, as well as other mission requirements such as cost, weight, reliability, orbital motion and lifetime are the key parameters which drive the decision of which technique to use. The various techniques can be grouped according to whether the applied torques are passive or active.

In the context of this thesis we will not be interested in the attitude control devices, but only in the attitude determination, for this reason the attitude control systems will not be studied in depth.

2.4 Types of attitude sensors

2.4.1 Relative attitude sensors

Many sensors generate outputs that reflect the rate of change in attitude. These require a known initial attitude, or external information to use them to determine attitude. Many of this class of sensor have some noise, leading to inaccuracies if not corrected by absolute attitude sensors [6].

- Gyroscopes

  Gyroscopes are devices that sense rotation in three-dimensional space without reliance on the observation of external objects. Classically, a gyroscope consists of a spinning mass, but there are also "ring laser gyros" utilizing coherent light reflected around a closed path. Another type of "gyro" is a hemispherical resonator gyro where a crystal cup shaped like a wine glass can be driven into oscillation just as a wine glass "sings" as a finger is rubbed around its rim. The orientation of the oscillation is fixed in inertial space, so measuring the orientation of the oscillation relative to the spacecraft can be used to sense the motion of the spacecraft with respect to inertial space.

- Motion reference units

  Motion reference units are a kind of inertial measurement unit with single- or multi-axis motion sensors. They utilize MEMS gyroscopes. Some multi-axis MRUs are capable of measuring roll, pitch, yaw, integrated with 3 axis linear acceleration. They have applications outside the aeronautical field, such as:
    - Antenna motion compensation and stabilization
    - Dynamic positioning
    - Heave compensation of offshore cranes
    - High speed craft motion control and damping systems
    - Hydro acoustic positioning
    - Motion compensation of single and multibeam echosounders
    - Ocean wave measurements
    - Offshore structure motion monitoring
    - Orientation and attitude measurements on Autonomous underwater vehicles and Remotely operated underwater vehicles
    - Ship motion monitoring

2.4.2 Absolute attitude sensors

This class of sensors sense the position or orientation of fields, objects or other phenomena outside the spacecraft [6].

- Horizon sensor

  A horizon sensor is an optical instrument that detects light from the 'limb' of Earth's atmosphere, i.e., at the horizon. Thermal infrared sensing is often used, which senses the comparative warmth of the atmosphere, compared to the much colder cosmic background. This sensor provides orientation with respect to Earth about two orthogonal axes. It tends to be less precise than sensors based on stellar observation. Sometimes referred to as an Earth sensor.

- Orbital gyrocompass

  Similar to the way that a terrestrial gyrocompass uses a pendulum to sense local gravity and force its gyro into alignment with Earth's spin vector, and therefore point north, an orbital gyrocompass uses a horizon sensor to sense the direction to Earth's center, and a gyro to sense rotation about an axis normal to the orbit plane. Thus, the horizon sensor provides pitch and roll measurements, and the gyro provides yaw.

- Sun sensor

  A sun sensor is a device that senses the direction to the Sun. This can be as simple as some solar cells and shades, or as complex as a steerable telescope, depending on mission requirements.

- Earth sensor

  An Earth sensor is a device that senses the direction to Earth. It is usually an infrared camera; nowadays the main method to detect attitude is the star tracker, but Earth sensors are still integrated in satellites for their low cost and reliability.

- Star tracker

  A star tracker is an optical device that measures the position(s) of star(s) using photocell(s) or a camera. It uses magnitude of brightness and spectral type to identify and then calculate the relative position of stars around it.

- Magnetometer

  A magnetometer is a device that senses magnetic field strength and, when used in a three-axis triad, magnetic field direction. As a spacecraft navigational aid, sensed field strength and direction is compared to a map of Earth's magnetic field stored in the memory of an on-board or ground-based guidance computer. If spacecraft position is known then attitude can be inferred.

2.5 Operating principles of a star sensor

A star sensor comprises an imaging function, a detecting function and a data processing function. The imaging function collects photons from objects in the field of view of the sensor and focuses them on a detecting element. This element converts the photons into an electrical signal that is then subject to some processing to produce the sensor output [7].
Figure 1 present a schematic of this sensor model.

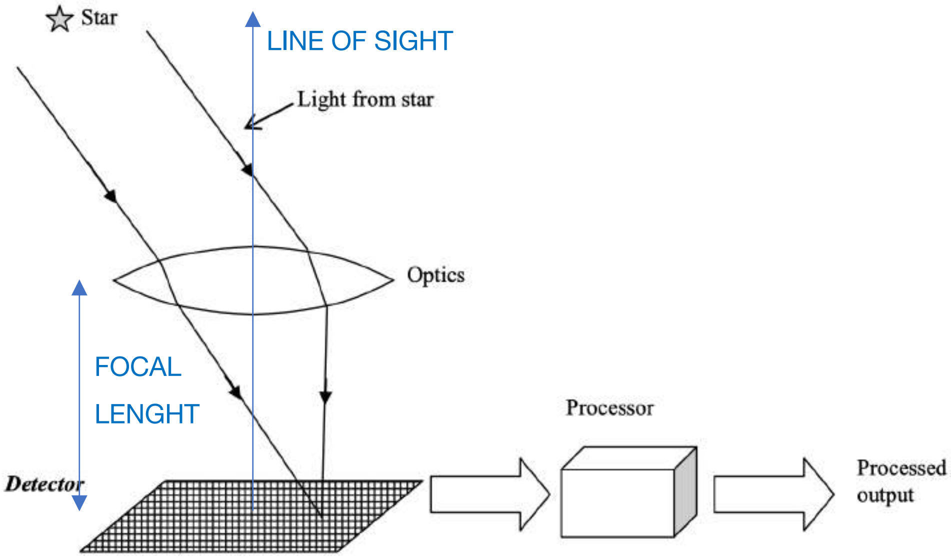


Figure 2 - Generalized Star Sensor model [6]

The ECSS standard defines 3 types of star sensors, defined in function of their capabilities [7]:

1. **Star camera**

   Capabilities:

   a. Cartography

2. **Star tracker**

   Capabilities:

   a. Cartography
   b. Star tracking

3. **Autonomous star tracker**

   Capabilities:

a. Autonomous attitude determination (‘lost in space’ solution);
b. Autonomous attitude tracking (with internal initialization).

The different capabilities, with specific inputs and outputs, are defined below:

Cartography is the capability to scan the entire sensor field of view and to locate and output the position of each star image within that field of view.

Inputs:

1. Only acquisition command

Outputs:

1. star position,
2. measurement date.

Star Tracking is the capability to measure the location of selected star images on a detector, to output the co-ordinates of those star images with respect to a sensor defined reference frame and to repeatedly re-assess and update those co-ordinates for an extended period of time, following the motion of each image across the detector.

Inputs:

1. the initial star position;
2. the angular rate;
3. validity date.

Outputs:

1. the position of each Star Image with respect to a sensor-defined reference frame;
2. focal length if star position on the detector chip is output in units of length;
3. the measurement date.

Autonomous star tracking is the capability to detect, locate, select and subsequently track star images within the sensor field of view for an extended period of time with no assistance external to the sensor.

Inputs:

1. the angular rate;
2. the validity date.

Outputs:

1. the position of each star image with respect to a sensor-defined reference frame;
2. the Measurement date.

Autonomous attitude determination is the capability to determine the absolute orientation of a defined sensor reference frame with respect to a defined inertial reference frame and to do so without the use of any a priori or externally supplied attitude, angular rate or angular acceleration information.

Inputs:

1. Only acquisition command

Outputs:

1. the relative orientation of the defined sensor reference frame with respect to the defined inertial reference frame (usually expressed in the form of a normalized attitude quaternion[1]);
2. the Measurement date;
3. a validity index or flag estimating the validity of the determined attitude.

Autonomous attitude tracking is the capability to repeatedly re-assess and update the orientation of a sensor-defined reference frame with respect to an inertially defined reference frame for an extended period of time, using autonomously selected star images in the field of view, following the changing orientation of the sensor reference frame as it moves in space.

Inputs:

1. the attitude quaternion;
2. the 3-dimension angular rate vector giving the angular rate of the sensor boresight reference frame with respect to the inertial reference frame;
3. the validity date for both supplied attitude and angular rate.

[1] Quaternions are a numerical systems that extends complex numbers for representing orientations and rotations of objects in three dimensions, using 4 numbers, as an alternative to Euler angles.

Outputs:

1. the orientation of the sensor defined reference frame with respect to the inertially defined reference frame (normally in the form of an attitude quaternion);
2. the Measurement date;
3. a validity index or flag, estimating the validity of the determined attitude;
4. measurement of Star Magnitude for each tracked Star Image.

In general, a stellar sensor can be used as a relative attitude sensor, measuring only the angular rates, or as an absolute attitude sensor, if it is able to identify the stars observed on the basis of a catalog stored on board.

2.6 Operating principles of a MEMS-based rate gyro

A vibrating structure gyroscope, defined by the IEEE as a Coriolis vibratory gyroscope (CVG) [8] is a gyroscope that uses a vibrating structure to determine the rate of rotation.

The underlying physical principle is that a vibrating object tends to continue vibrating in the same plane even if its support rotates. The Coriolis effect causes the object to exert a force on its support, and by measuring this force the rate of rotation can be determined.

Vibrating structure gyroscopes are simpler and cheaper than conventional rotating gyroscopes of similar accuracy. Inexpensive vibrating structure gyroscopes manufactured with MEMS technology are widely used in consumer devices.

There are different types of CVG implementation:

Cylindrical resonator gyroscope (CRG)

This type of gyroscope uses metal alloys with attached piezoelectric elements and a single-piece piezoceramic design.

Piezo-electric elements on the resonator produce forces and sense induced motions. This electromechanical system provides the low output noise and large dynamic range that demanding applications require, but suffers from intense acoustic noises and high overloads.

Piezoelectric gyroscopes

A piezoelectric material can be induced to vibrate, and lateral motion due to Coriolis force can be measured to produce a signal related to the rate of rotation.

Tuning fork gyroscope

This type of gyroscope uses a pair of test masses driven to resonance. Their displacement from the plane of oscillation is measured to produce a signal related to the system's rate of rotation.

Wine-glass resonator

Also called a hemispherical resonator gyroscope or HRG, a wine-glass resonator uses a thin solid-state hemisphere anchored by a thick stem. The hemisphere with its stem is driven to flexural resonance and the nodal points are measured to detect rotation.

Vibrating wheel gyroscope

A wheel is driven to rotate a fraction of a full turn about its axis. The tilt of the wheel is measured to produce a signal related to the rate of rotation.

Inexpensive vibrating structure microelectromechanical systems (MEMS) gyroscopes have become widely available. These are packaged similarly to other integrated circuits and may provide either analog or digital outputs. In many cases, a single part includes gyroscopic sensors for multiple axes. Some parts incorporate multiple gyroscopes and accelerometers (or multiple-axis gyroscopes and accelerometers), to achieve output that has six full degrees of freedom. These units are called inertial measurement units, or IMUs.

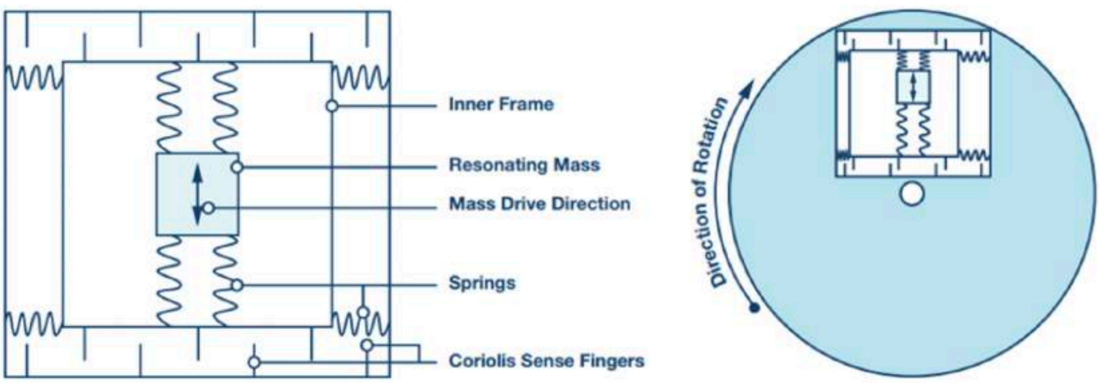


Figure 3 - Schematic of the gyroscope's mechanical structure (left), Coriolis effect in frame and resonating mass [Analog Devices]

3 Requirements

3.1 Performance requirements

The autonomy of the sensor, as well as its generic installation requirements, are the main drivers for both the hardware and software architectures. Of course, the low consumption requirements are also important.

The expected requirements for absolute and relative attitude accuracy are 0.1 degree, but values up to 1 degree can be accepted for most applications. This is based on an evaluation of the accuracy at nadir for a satellite working in a typical 800 km altitude [1]. An estimate of the pointing accuracy on the earth as a function of angular resolution is as reported in Table 1.

Attitude accuracy	**Nadir pointing error**
±1.0°	13.96 km
±0.5°	6.98 km
±0.1°	1.40 km
±0.02°	280 m

Table 1 - Attitude accuracy vs Nadir pointing error

In Table 2 are listed some example of small satellite's cameras. Horizontal and vertical angles of view and respective estimated field of view on the Earth surface (at 800km distance) are highlighted. As you can see, the previous estimated nadir pointing accuracy with ±0.02° attitude accuracy is typically acceptable.

Satellite or instrument	**Camera AoV**	**Sensor resolution (pixel)**	**Earth observation @ 800km**	**Earth resolution @ 800km**
KARI/HiREV	8.81° x 7.05°	3388 × 2712	123 km x 98 km	36 m
APIS/OPTOS	12.16° x 9.73°	1280 x 1024	170 km x 136 km	133 m
UKube-1/C3D	19.48° 15.58°	1280 x 1024	275 km x 220 km	215 m

Table 2 - Example of cubesats camera angle of view, estimated Earth observation field and resolution

For what regards attitude rates an initial requirement is set at 0.001 deg/sec for low rates and 0.1 deg/sec for high rates. Angular rates of a nanosat after deployment is a function of the deploy mechanism (e.g. P-POD, Poly-Picosatellite Orbital Deployer), of attitude dynamics of the upper stage, and of the possible deployment of appendages (e.g. antennae). A typical CubeSat is designed for a maximum angular rate of 0.1 rad/s (5.7 deg/s) for each axis, therefore for a maximum of 10 deg/s (about 1.7 rpm). With a safety factor of 50% this becomes 15 deg/s (or 2.5 rpm). The same applies to micro-satellites, and the 82 kg SumbandilaSAT is an example of satellite with an expected deployment rate of 10 deg/s as per launcher specifications.

Data fusion from both sensors, the optical one and the rate gyroscope one, will enable continuous access to angular rates, while for attitude values, the choice of the operating mode will be predominant.

Regarding the angle of view of the optical sensor, we have examined the characteristics of some commercial star trackers:

- Vectronic VST-41M: 14°x14°
- Leonardo Spacestar: 20°x20°
- Leonardo A-STR: 20°x20°

Most commercial star trackers use a sensor with a square active area. However, we can consider acceptable for our purpose a sensor with an active rectangular area (proportions 4:3), cheaper and easily available on the market.

3.2 Mass and volume

With reference to the nano satellites and in particular to the CubeSats, it is appropriate to design the attitude sensor in order to use a limited percentage of mass and volume available to the satellite, leaving space for other necessary on-board systems, such as batteries or other power systems, on-board computer, radio communication system and scientific instrumentation.

In the case of a CubeSat, a plane of dimensions contained in the 10x10cm structure could be occupied, with a maximum height of 25mm. The critical size will be the height since the protruding part will surely be the lens of the optical sensor, to which particular attention will be paid during the design phase.

The weight should not exceed 100g. As will be indicated in the conclusions, the weight of our prototype is 19g (54g with adapter board).

3.3 Electrical interface requirements

Regarding the power supply of the sensor, all the components will be chosen compatible with the 3.3V voltage standard. With this premise, it will be easy to make the system compatible, for example, with 3.7V lithium batteries. It would be ideal to make the system compatible also with the 28V standard, used in many satellites. It will also be important to keep the current consumption of the sensor at a minimum level, to maximize the autonomy of the host satellite.

The main communication interface with the sensor will be serial and the UART, SPI or I^2C protocols can be used. The main data exchanged through this interface will be simple input commands and attitude variables as output. For this reason, a 115200 standard baud rate is suitable and compatible with most OBC. Through the same interface it could be possible to transfer even the raw image of an entire acquired frame, for example for scientific purposes, but for this a non-negligible transfer time will therefore be required.

An ICSP type programming port (In Circuit Serial Programming) will be accessible on the sensor, both for initial programming and debugging on the ground, and to make possible a reprogramming of the system during the mission, by the on-board computer.

3.4 Reliability requirements

A satellite, being exposed to radiation, is sensitive to failures caused by the involuntary change of memory elements. To avoid this problem, typically, the on-board electronics are designed with a redundancy of microcontrollers, microprocessors, FPGAs and memories.

In the case of micro and nano satellites these techniques are not always used, both because of the reduced volume and mass requirements, and because the missions planned for these satellites generally have a relatively short operating life.

Implementing cyclic algorithms, in the event that an error occurs during a cycle, this can be corrected in the next cycle.

It will also be possible to provide a task to verify the integrity of the memory and critical registers. Techniques of this type are known as EDAC (Error Detection And Correction) and allow a good immunity to errors, in the absence of hardware redundancy.

In any case, as explained in the previous paragraph, this sensor will be equipped with a programming port, which can also be used during the mission by the on-board computer (which can be designed with redundancy) to reprogram the sensor in case of failures. According to the ICSP standard, the host system should be able to communicate via a serial bus (3.3V or 5V clock and data signal) and generate a +12V voltage on the MCLR pin [9].

4 Design considerations

4.1 Architecture

The system will be managed by a 32-bit microcontroller. The interface with the optical CMOS/APS sensor is typically parallel, but a secondary I^2C or SPI interface is used in parallel for the configuration and control of the sensor.

Also the MEMS gyroscopic sensor will be interfaced to the microcontroller via I^2C or SPI.

A power supply stage will be designed to adapt and adjust the voltage required for the different devices, according to the specifications discussed previously and according to the needs of each component.

The microcontroller will be directly connected to the external Data I/F (UART/SPI/I^2C interfaces) and to the Debug I/F (ICSP port).

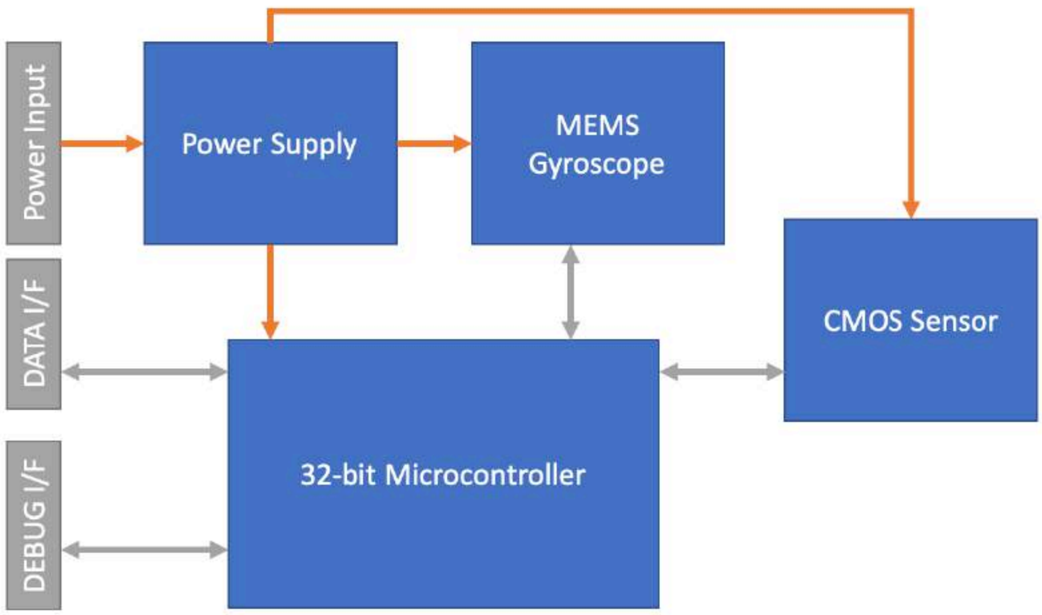


Figure 4 - System architecture

4.2 Component selection

4.2.1 Image sensor

The selection of the image sensor is driven by the typical working logic of a star tracker. In normal operations, the star tracker has to assess which stars are available in the field of view (assuming the angular rates are low enough for a correct acquisition). This full-frame acquisition is used to select a few targets stars that will be tracked to determine relative motion or to establish the absolute attitude at the time of acquisition. After selection the ability to acquire only small windows around the selected target stars will improve the performances of the algorithms used for attitude and attitude rate determination.

These two functions, full-frame acquisition and selected windows acquisition, are therefore the drivers in the selection, to which it is added the technology aspect, the choice of APS technology to simplify interfacing with a microcontroller.

4.2.1.1 First selection

In a first stage of this project we have selected the sensor NOIP1SN1300A (ON Semiconductor), because of its many interesting features, such as 1.3 MP resolution (SXGA standard), 10 bit image depth, up to 210 fps and multiple window readout, with up to 8 user-selected Regions Of Interests (ROI) [10].

This last feature, allows the system to acquire at the same time up to 8 small areas around each clamped star, after a first acquisition of an entire frame, to speed up the star tracking, without the need of full frame acquisition at each step.

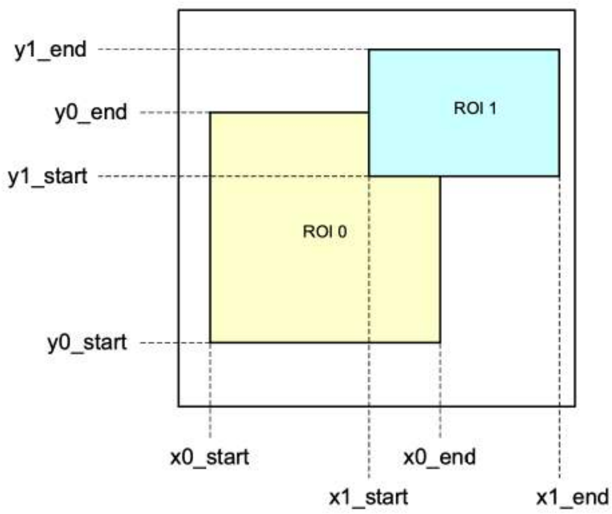


Figure 5 - Multiple window readout

The criticality of this sensor was the handling of the output stream. The data interface of this sensor consists in 4 LVDS lanes with 360MHz clock (or 288MHz with 8 bit down-sampling). This high-speed stream (720 Mbps @ 10 bit or 576 Mbps @ 8 bit) is not manageable with a microcontroller (such as a PIC32), but it requires an FPGA architecture, which increase the complexity and also the size of the system.

We evaluated the possibility to reduce the output data rate of the sensor, accepting a reduction of the frame rate, but the data rate is not a software configurable parameter (configuration registers) and in the sensor datasheet is not explicit the possibility to use a lower system clock than the standard value (72MHz, with 5x or 4x PLL), with a consequential data rate and frame rate reduction.

In addition to the data rate handling problem/issue, if we read a single full frame with a microcontroller, we need the following RAM space:
8 bit option: (1280px)*(1024px)*(8bit) = 10485760bit = 10.4Mb
10 bit option: (1280px)*(1024px)*(10bit) = 13107200bit = 13.1Mb
Standard microcontrollers do not have this RAM available, so an external RAM would be required, with an increase in complexity and dimensions.

4.2.1.2 Second selection

After the first analysis, another CMOS sensor has been chosen, with the same technology (APS), but with a lower resolution and lower output data rate. The chosen sensor is the **MT9V034-D**, 1/3-Inch Wide-VGA CMOS Digital Image Sensor, produced by ON Semiconductor [11].

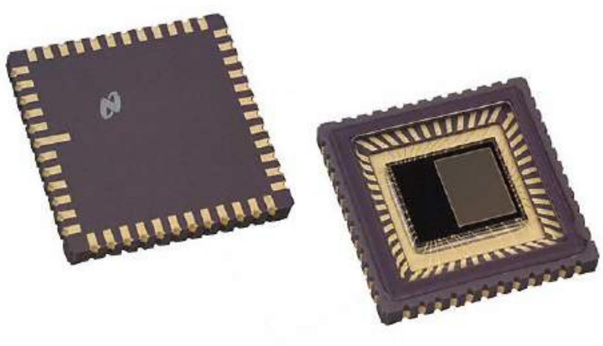


Figure 6 - MT9V034 package top and bottom view

Resolution: this Wide-VGA format CMOS sensor has 752 x 480 active pixels (360,960 pixels). This resolution is sufficient to have an acceptable resolution (as described in the optical considerations of paragraph 4.2.2), while maintaining a limited data flow, manageable with the microcontroller.

Global Shutter Photodiode Pixels: this feature allows simultaneous pixel integration, ideal for capturing moving objects. This feature is advantageous compared to the classic rolling shutter, which provides a slight delay between the acquisition of one line and the next, deforming the image.

RGB Bayer or Monochrome: the selected version of this sensor is of monochromatic type, without RGB Bayer filter.

Data output: the sensor has an output on a 10-bit parallel interface, with a clock signal and two frame and line boundary signals. Only the most significant 8 bits will be used in our application to optimize the memory used.

Two-wire Serial Interface: an I^2C compatible serial interface is available for configuration register configuration and readout.

Window Size: user programmable window is available to acquire any smaller format, with parametric coordinates and window dimensions. It is possible to configure a single window, therefore in the case of acquisitions of different stars it is necessary to make several parametric acquisitions in sequence, taking into account the delay between one and the next.

Automatic Controls: Auto Exposure Control (AEC) and Auto Gain Control (AGC) are available on this sensor, as an alternative to the manual configuration of the gain and exposure.

4.2.2 Optics

The selection of lens for our sensor should starts from the optical considerations, previously discussed in paragraph 3.1.

The relation between the angle of view and the focal length of the lens is defined by the following equation [12]:

$$\theta = 2\tan^{-1}\left(\frac{h}{2f}\right)$$

With:

θ = angle of view

h = frame dimension

f = focal length

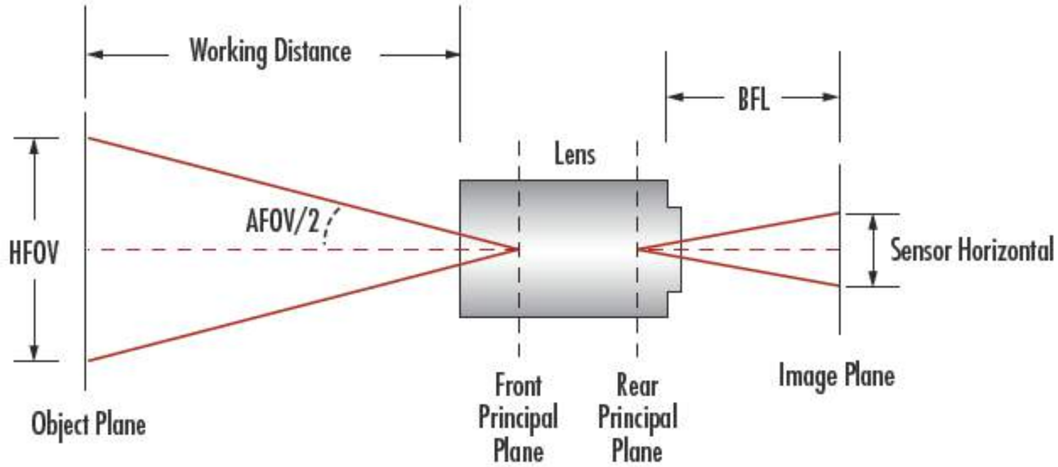


Figure 7 - Optical path

As frame dimension we can use, in the same equation, the diagonal, horizontal and vertical dimension of the active area of our sensor (4.51x2.88mm), to obtain the respective angles of view.

We can calculate the different angles of view from focal length of most common lenses available on the market.

Focal Length	**Diagonal AoV (°)**	**Horizontal AoV (°)**	**Vertical AoV (°)**
12 mm	25.1	21.3	13.7
16 mm	19.0	16.0	10.3
25 mm	12.2	10.3	6.59
35 mm	8.74	7.37	4.71

Table 3 - Focal length vs angle of view

From Angle of View data and sensor resolution (752x480), we can calculate the angle resolution obtained with each lens:

Focal Length	Angular resolution (°)
12 mm	0.028
16 mm	0.021
25 mm	0.014
35 mm	0.010

Table 4 - Focal length vs angular resolution

The most common lens mount for this type of image sensor (1/3”) is the S-Mount standard, also known as M12 (metric diameter of the screw mount).

The chosen lens is a **16mm f/2** lens, manufactured by Edmund Optics (#66-894) [13].

Lens specifications:

Focal Length FL (mm)	16.0
Aperture (f/#)	f/2
Maximum Sensor Format	1/3"
Length (mm)	14.40
Mount	M12 x 0.5 (S-Mount)
Resolution, Full Field	60 lp/mm @ 20% Contrast
Resolution, 0.7 Field	65 lp/mm @ 20% Contrast
Resolution, On-Axis	75 lp/mm @ 20% Contrast
IR Cut Filter	Yes
Field of View @ Max Sensor Format	17
Working Distance (mm)	400 - ∞
Distortion (%)	-2.50 @ Full Field
Maximum Diameter (mm)	14.0

Table 5 - Edmund Optics #66-894 lens specifications

Lens dimensions:

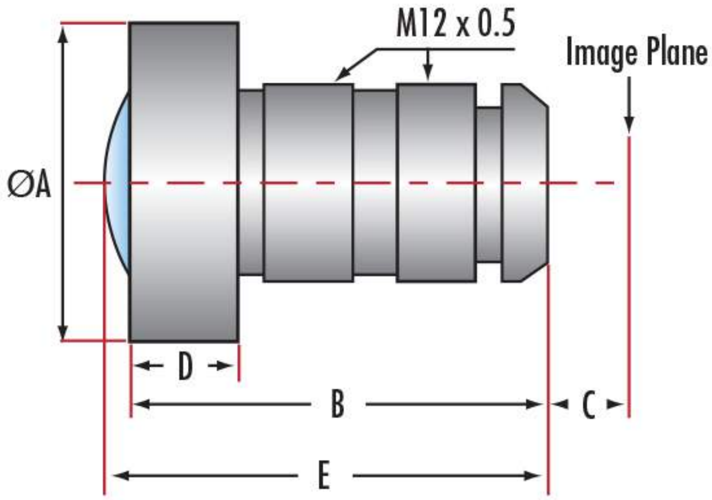


Figure 8 - Lens dimensions reference

Dimension	Value (mm)
A	14.0
B	14.4
C	8.0
D	4.5
E	N/A

Table 6 - #66-894 lens dimensions

To mount the lens to our module, we chose an M12 PCB Lens Holder (Edmund Optics #66-382) [13].

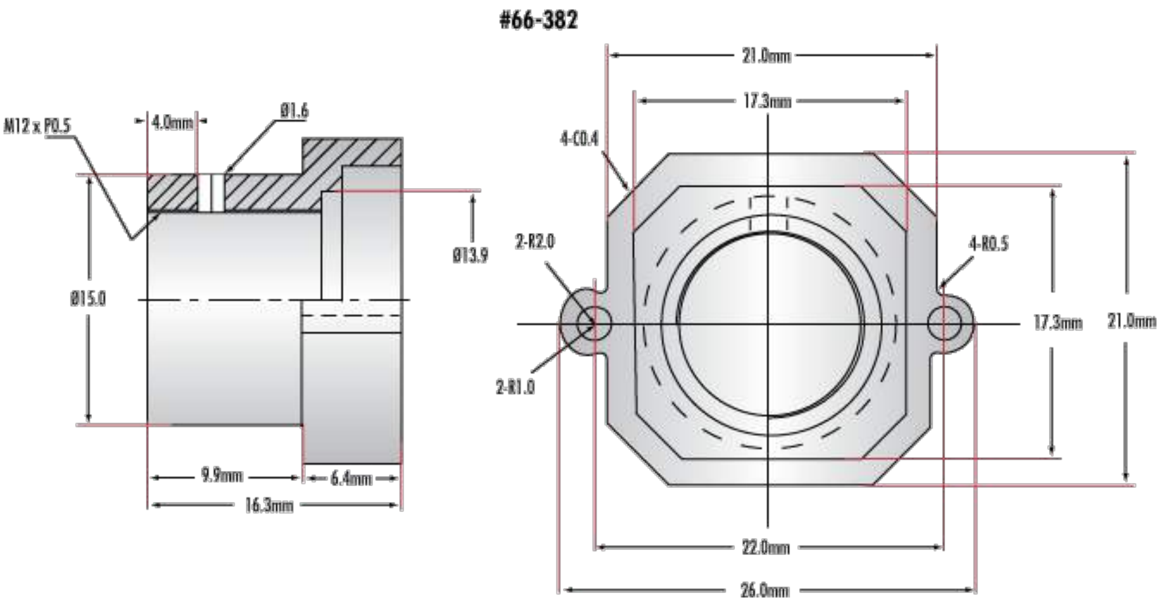


Figure 9 - Edmund Optics #66-382 lens holder

4.2.3 Gyroscope

The chosen three-rate gyroscope is **I3G4250DTR**, produced by STMicroelectronics [14]. It has a sensibility of 8.75 mdps for ± 245 dps (degrees per second). Our choice has been oriented to this type of 3-axis angular rate sensor for the remarkable stability at zero level (important for very slow rates response), the sensitivity over temperature, and low-power characteristic. Equipped with standard SPI and an I^2C compatible interfaces, has allowed to use a protocol shared with other devices.

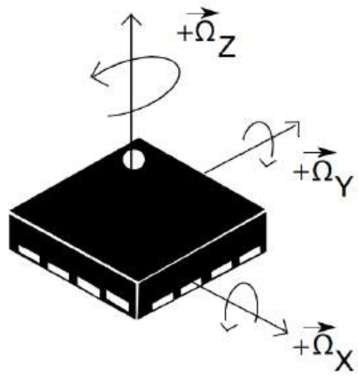


Figure 10 - I3G4250DTR package and axis orientation

4.2.4 Microcontroller

The need to perform floating-point computations (e.g. for rotation matrices applications) drove the selection of a low-power 32-bit microcontroller. The selected MCU is the **PIC32MZ2048EFM100**, produced by Microchip Technologies [15].

The microcontroller main specifications concerning the project are analyzed below:

System Clock: different clock modes are supported. The most convenient and effective one is based on an external crystal oscillator, with internal PLL for frequency multiply. The standard value of system clock of this microcontroller is 200MHz, increasable to maximum 252MHz. This is one of the maximum clock frequency that a 32bit microcontroller can be reach, so it’s suitable for optimizing the response times of the sensor, having to manage a considerable amount of data.

The possibility to generate an output clock reference with dedicated PLL is also important to provide the required input clock for the CMOS/APS sensor (26.667MHz).

SRAM: this microcontroller has a 512KB integrated SRAM memory. Considering the characteristics of the image sensor, a full frame image is composed by 360,960 pixels. With an 8-bit image depth acquisition, 361KB of SRAM are enough to contain a full frame image. The remaining 151KB space can be used to contain some secondary windows acquisitions, gyroscope acquisitions and other system variables. For example, a parametric window image of 50x50 pixel, at 8-bit depth, needs 2.5KB.

Interfaces: multiple serial ports are available, with reprogrammable (shared) pins: 6 UART (25 Mbps), 6 SPI (50Mbps), 5 I^2C (1Mbaud) and so on. The availability of these ports will allow us to manage the various interfaces necessary for the sensors and different external interfaces.

Temperature tolerance and qualification: this is not a space qualified component, but the AEC-Q100 qualification – grade 1 (Automotive Electronic Council - Failure Mechanism Based Stress Test Qualification for Integrated Circuits) assure a temperature tolerance from -40°C to 125°C, with an acceptable operating life (1000 hours at +125°C), electrostatic discharge tolerance (up to 500V) and good mechanical shock and vibration tolerance (tested with 5 pulses 0.5ms 1500g acceleration and 20Hz to 2KHz 50g peak vibrations) [16].

5 Hardware design

5.1 Power supply

Respecting the project specifications, a common supply of 3.3V is needed, starting from a system supply from a minimum of 5V to at least 28V. Considering a potentially high voltage dropout, a linear regulator is not recommended, as it would generate an excessive heat dissipation. A switching power supply stage was therefore designed.

To calculate the maximum current required, consider the maximum specific consumption of each component:

Component	Maximum current (mA)
PIC32MZ2048EFM100	170
MT9V034	48.5
I3G4250D	6.1

Table 7 - Components power consumption

The total maximum current required is 225mA, so we designed a 500mA switching power supply, with Microchip MCP16331T step-down regulator [17].

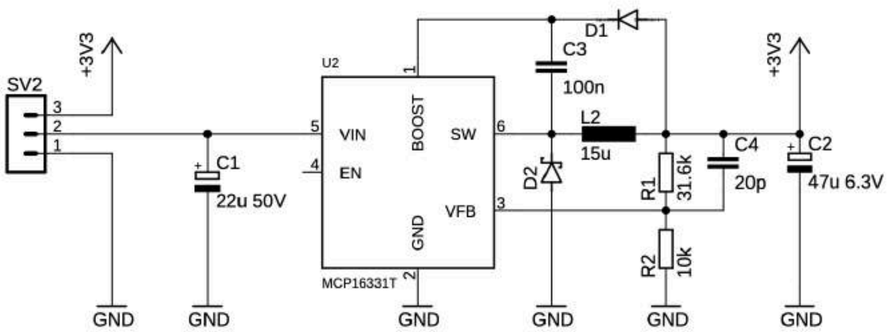


Figure 11 - Power supply stage

The schematic is in line with that proposed by the datasheet of the manufacturer, as a typical application.

Since it is an adjustable output voltage regulator, we have to calculate the values of feedback resistor divider:

$$R_1 = R_2\left(\frac{V_{OUT}}{V_{FB}} - 1\right)$$

Fixing R_2 to 10kΩ and V_{OUT} to 3.3V (V_{FB}=0.8V), we obtain an R_1=31.25kΩ.
The nearest standard value for R_1 is 31.6kΩ, which will determine a V_{OUT}=3.328V.
The value of L2 inductor is recommended by the datasheet, depending on the output voltage: 15μH for 3.3V. Considering the maximum current consumption of the system, the current tolerance of L1 should be minimum 250mA.
Values of input and output capacitor also match the minimum capacitance values recommended. The voltage tolerance of C1 should be minimum 50V, to match the maximum input voltage of the regulator. C2 can have a voltage tolerance of 6.3V, because the output voltage cannot exceed 3.3V.
With a 3 pin connector (SV3) it is possible to supply an input voltage (4.8 to 50V) to the module (using pin 1 and 2), or bypass the step-down regulator, if the host system can directly provide a stable 3.3V power supply (using pin 1 and 3).

5.2 Microcontroller

The microcontroller power supply is directly the main 3.3V of the module, with two decoupling capacitors. There is no need to separate analog voltage supply and analog ground, because in this project we will not use any analog input signals.
The main clock source is a 24MHz crystal oscillator, which allows to generate a 200MHz system clock with an internal PLL, as described in the next chapter.
Another internal PLL stage will be used to generate a 26.667MHz clock (REFCLK3), required by the APS sensor.
All other signals have been chosen compatibly with the sensors requirements and microcontrollers interfaces capabilities. Table 6 shows the port/pin assignments for each signal.
The following paragraphs explain the functions of the different connections and external interfaces.

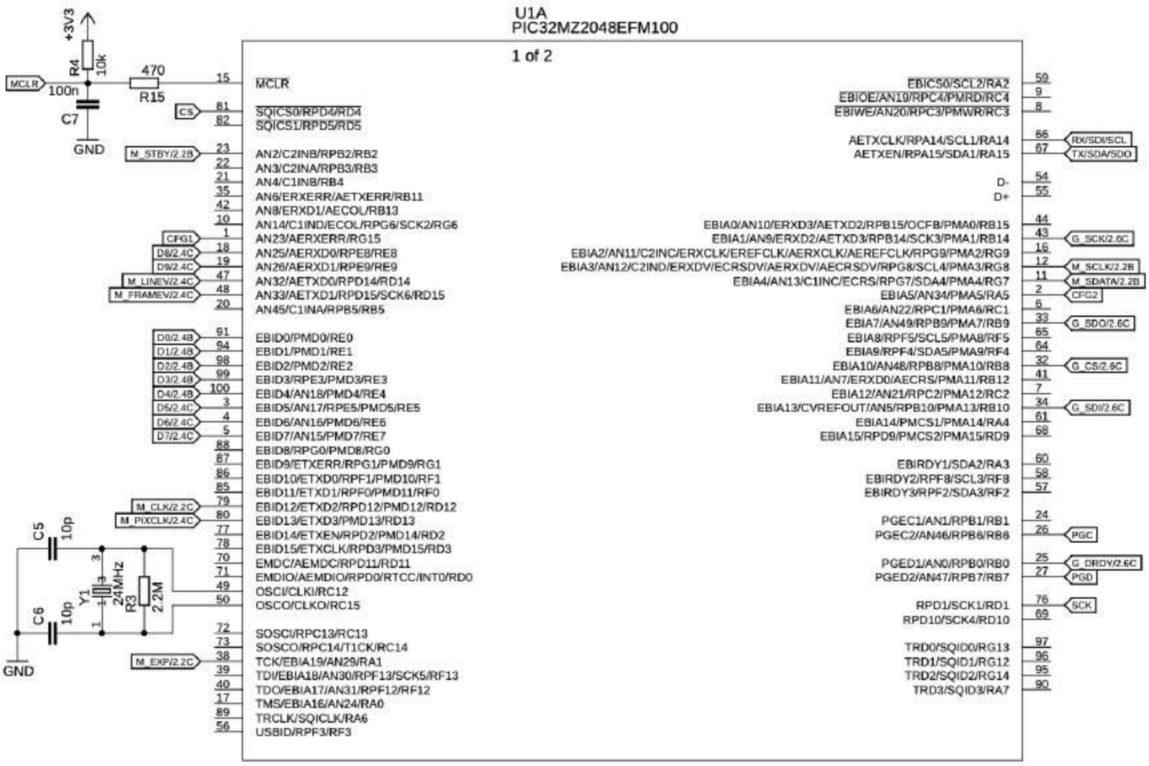


Figure 12 - Microcontroller schematic A

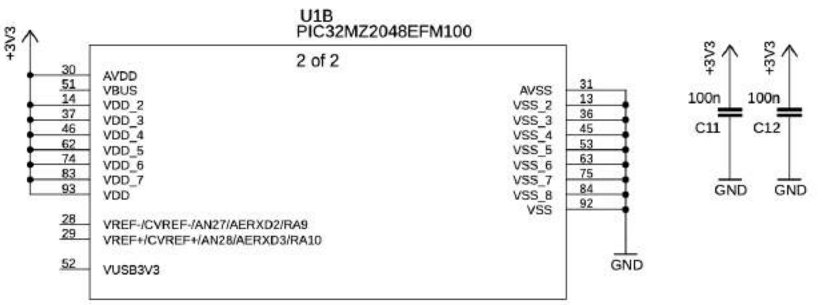


Figure 13 - Microcontroller schematic B

NAME	#	I/O Type	SIGNAL LABEL	DESCRIPTION
RA1	38	OUTPUT	M_EXP	CMOS Sensor Exposure Signal
RA5	2	INPUT	CFG2	Config jumper (interface selection)
RA14	66	UART1/SPI1/I2C1	RX/SDI/SCL	External Interface
RA15	67	UART1/SPI1/I2C1	TX/SDA/SDO	External Interface
RB0	25	INPUT	G_DRDY	Gyro Sensor Interface [Data Ready]
RB2	23	OUTPUT	M_STBY	CMOS Sensor Standby Signal
RB6	26	ICSP	PGC	ICSP Port
RB7	27	ICSP	PGD	ICSP Port
RB8	32	SPI3	G_CS	Gyro Sensor Interface [Chip Select]

RB9	33	SPI3	G_SDO	Gyro Sensor Interface [Data Output]
RB10	34	SPI3	G_SDI	Gyro Sensor Interface [Data Input]
RB14	43	SPI3	G_SCK	Gyro Sensor Interface [Clock]
RD1	76	SPI1	SCK	External Interface
RD4	81	SPI1	CS	External Interface
RD12	79	REFCLK3	M_CLK	CMOS Sensor Master Clock
RD13	80	INPUT	M_PIXCLK	CMOS Sensor Pixel Clock Output
RD14	47	INPUT	M_LINEV	CMOS Sensor Line Valid Signal
RD15	48	INPUT	M_FRAMEV	CMOS Sensor Frame Valid Signal
RE0	91	INPUT	D0	CMOS Sensor parallel interface
RE1	94	INPUT	D1	CMOS Sensor parallel interface
RE2	98	INPUT	D2	CMOS Sensor parallel interface
RE3	99	INPUT	D3	CMOS Sensor parallel interface
RE4	100	INPUT	D4	CMOS Sensor parallel interface
RE5	3	INPUT	D5	CMOS Sensor parallel interface
RE6	4	INPUT	D6	CMOS Sensor parallel interface
RE7	5	INPUT	D7	CMOS Sensor parallel interface
RE8	18	INPUT	D8	CMOS Sensor parallel interface
RE9	19	INPUT	D9	CMOS Sensor parallel interface
RG7	11	I2C4	M_SDATA	CMOS Sensor Serial Interface Data
RG8	12	I2C4	M_SCLK	CMOS Sensor Serial Interface Clock
RG15	1	INPUT	CFG1	Config jumper (interface selection)

Table 8 - Microcontroller pin assignment

5.3 APS-CMOS sensor interface

This APS-CMOS sensor has two main interfaces:

- 10-bit parallel interface (D0-D9) for data output stream
- 2-wire (I^2C) serial interface, for sensor configuration

A 26.667MHz clock input should be provided to the SYSCLK pin (connected to clock reference output of the microcontroller).

Other digital signal will be used:

- STANDBY (input): Shut down sensor operation for power saving
- EXPOSURE (input): Rising edge start exposure
- PIXCLK (output): Pixel clock out
- FRAME_VALID (output): Asserted when D_{OUT} data is valid (frame control)
- LINE_VALID (output): Asserted when D_{OUT} data is valid (line control)

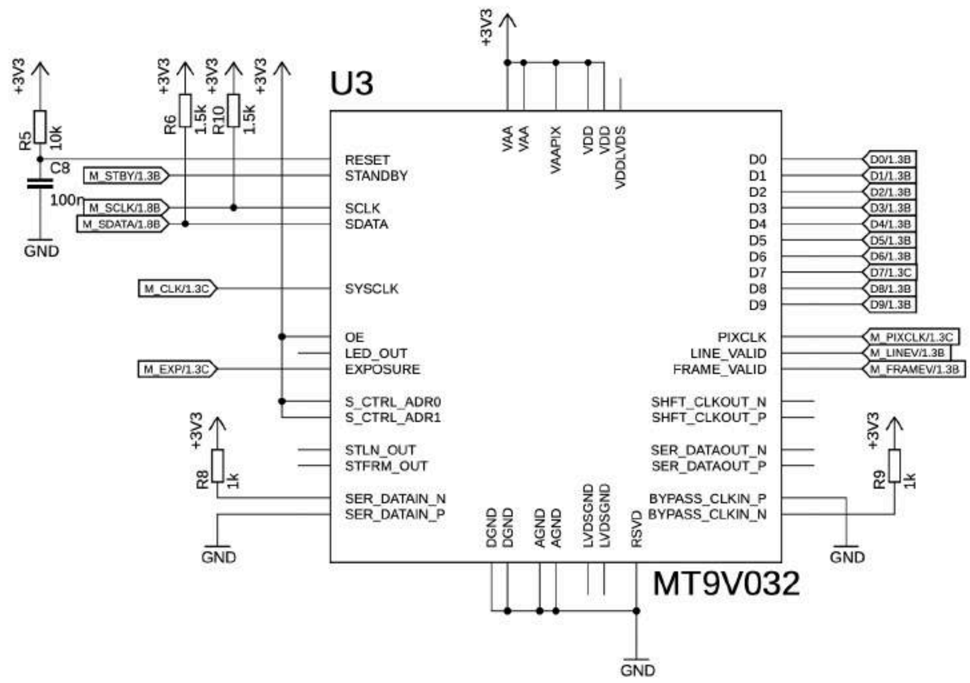


Figure 14 - CMOS/APS sensor schematic

5.4 Gyroscopic sensor

The gyroscopic sensor is powered by the main 3.3V supply, with a decoupling 100nF capacitor.

A filter with C9 and C10 capacitor and R7 resistor is implemented as recommended on manufacturer datasheet.

The 4 signals of SPI interfaced (Data Output, Data Input, Clock, Chip Select) is properly mapped to the microcontroller. DRDY/INT2 signal is also connected to a microcontroller input and can be used as data ready signal or interrupt.

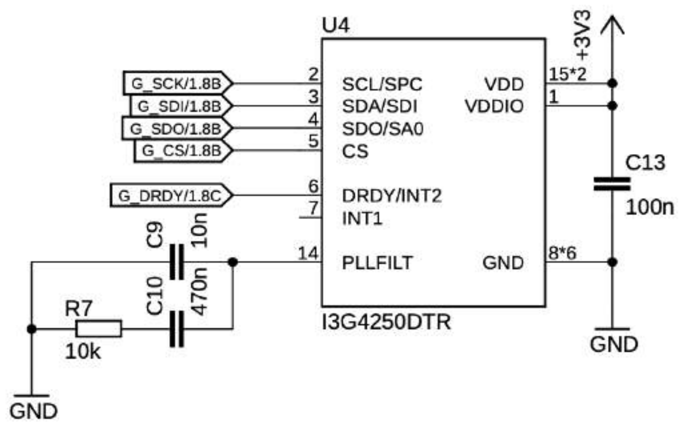


Figure 15 - Gyroscopic sensor schematic

5.5 External communication interface

A **DEBUG I/F** (ICSP port) is available to program the device in a first set-up, but the onboard reprogramming by the on-board computer can also be provided, with a proper interface.

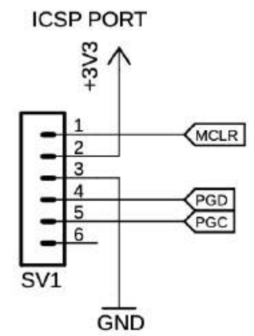


Figure 16 – DEBUG I/F

An external interface (**DATA I/F**) is needed to exchange data with the On-Board Computer and also for testing and debug operation. Given the numerous interfaces available to the microcontroller, the user can choose to use an UART, SPI or I^2C serial port, based on the available interface on the OBC. The needed signals share the same connector and it will be possible to choose by mounting two zero ohm resistors (jumpers) in different combinations. At firmware level, different interface will be configured in function of a jumpers-based configuration (CFG1 and CFG2 digital input).

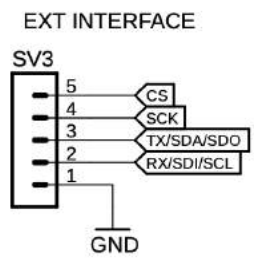


Figure 17 – DATA I/F

In Appendix D is reported the general schematics for DATA I/F interface type selection with CFG1 and CFG2 settings.

Table 7 shows the pin assignment on SV3 connector for different interfaces.

SV3 pin	UART	I²C	SPI
1	GND	GND	GND
2	RX	SCL	SDI
3	TX	SDA	SDO
4			SCK
5			CS

Table 9 - External interface pin assignment

5.6 PCB design

The PCB (Printed Circuit Board) was designed with Autodesk Eagle (v. 9). The board was designed on 2 layers, to contain production costs and because the complexity of the circuit did not require a higher number of layers. All the passive components used, with the exception of high-capacity capacitors, were chosen in the 0805 standard, to facilitate prototyping.

It was decided to mount the microcontroller, the power supply stage and the other discrete components on the bottom layer and the two sensors on the top layer, in order to minimize the dimensions of the module.

A ground plane has been designed on the top layer, where it is the largest free space. A guard ring was designed around the crystal, on bottom layer, in order to optimize the noise immunity.

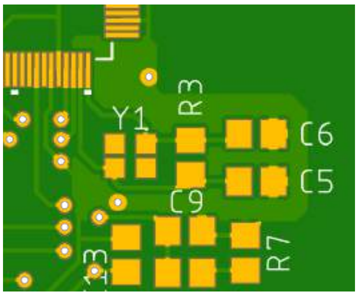


Figure 18 - Crystal guard ring

The result is a board measuring 50x50mm, which can be easily integrated into small satellites. The PCB layouts are shown on Appendix F.

Adapter board

In addition to the main module, an adapter board was designed with a form factor compatible with the CubeSatKit[2] standard.

All the signals used, including the power supplies and the different DATA I/F options (UART, I^2C or SPI) have been mapped to the compatible PC104 standard bus signals (4x26 pins with 0.100 "pitch). Most OBCs use the same bus with compatible pinout.

Four external JST connectors makes available the same ports both for debugging purposes and for connecting systems different from a CubeSat in a convenient way.

A series of optional zero-ohm resistors allow to enable selected signals, depending on those available on host system. Appendix D shows the complete schematic of this board, with pin mapping and possible jumper settings.

[2] http://www.cubesatkit.com/

6 Firmware design

6.1 Microcontroller configuration

Before processing the necessary functions, we need to configure the microcontroller registers related to the features that will be used. To do this, the MHC tool (MPLab Harmony Configurator) was used, which automatically generated the values of the registers, depending on the desired mappings and configurations. The Harmony libraries have also been included for the use of serial interfaces.

6.1.1 Peripherals configuration

With reference to Table 8, it was necessary to configure the registers that regulate the direction of the ports (input/output), the enabling of the serial interfaces (UART, I^2C, SPI) and their mapping to the desired pins.
In the following code, TRIS* registers set the port direction for each bit, where 0 is output and 1 is input, LAT* set the initial value of the port, ANSEL* set the analog or digital input mode:

```
/* PORTA Initialization */
LATA = 0x0; /* Initial Latch Value */
TRISACLR = 0x2; /* Direction Control */
ANSELACLR = 0x22; /* Digital Mode Enable */

/* PORTB Initialization */
LATB = 0x0; /* Initial Latch Value */
TRISBCLR = 0x4; /* Direction Control */
ANSELBCLR = 0x4705; /* Digital Mode Enable */

/* PORTC Initialization */
ANSELCCLR = 0x9000; /* Digital Mode Enable */

/* PORTD Initialization */
ANSELDCLR = 0xc000; /* Digital Mode Enable */

/* PORTE Initialization */
ANSELECLR = 0x3f0; /* Digital Mode Enable */
```

The following registers is needed to set all the used remappable pins (RP*):

```
/* PPS Input Remapping */
```

```
U1RXR = 13;
SDI3R = 6;

/* PPS Output Remapping */
RPA15R = 1;
RPB8R = 7;
RPB9R = 7;
RPD12R = 15;
```

6.1.2 Clock configuration

The diagram in Figure 19 explains the complex oscillator architecture of our PIC32 microcontroller. There are different selectable clock sources (internal RC oscillator, external clock generator or external crystal oscillator) and several PLL modules to synthetize different system clocks.

Our configuration includes a 24MHz crystal input and provide a 200MHz system clock.

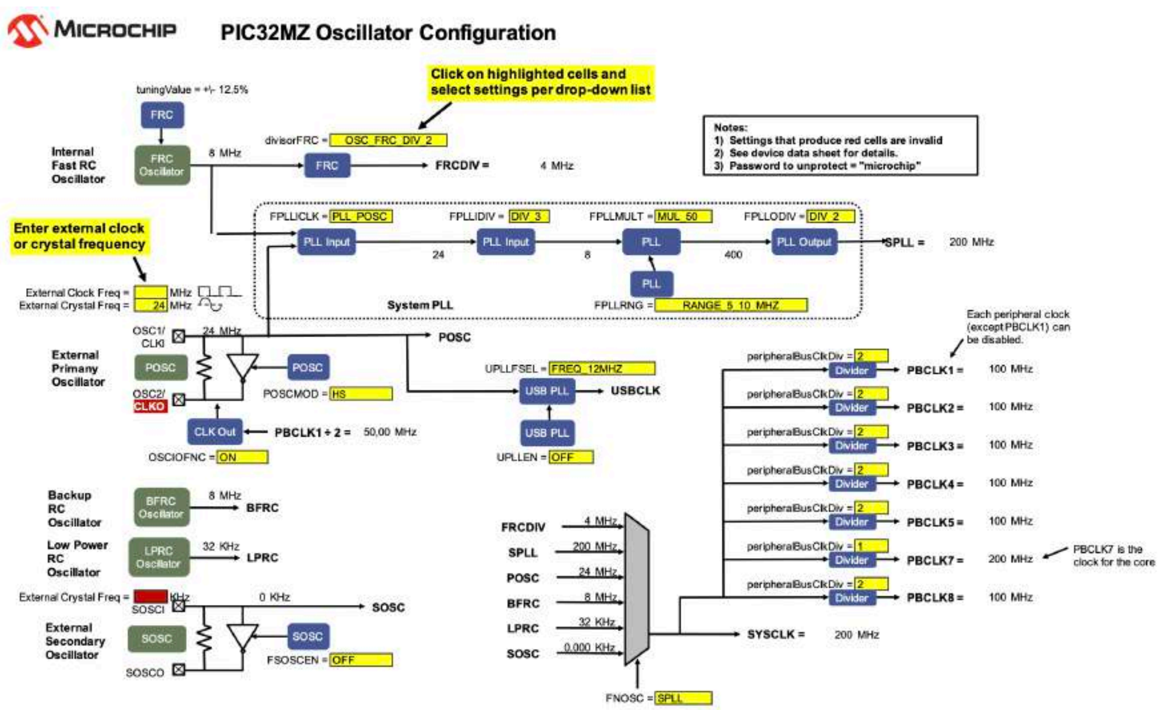


Figure 19 - PIC32MZ Oscillator Configuration

The following configurations derive from the previous diagram:

```
// Device Config Bits in  DEVCFG1:
#pragma config FNOSC =     SPLL
#pragma config FSOSCEN =   ON
```

```
#pragma config POSCMOD =   HS
#pragma config OSCIOFNC =  ON

// Device Config Bits in  DEVCFG2:
#pragma config FPLLICLK =  PLL_POSC
#pragma config FPLLIDIV =  DIV_3
#pragma config FPLLMULT =  MUL_50
#pragma config FPLLODIV =  DIV_2
#pragma config FPLLRNG =   RANGE_5_10_MHZ
#pragma config UPLLEN =    OFF
#pragma config UPLLFSEL =  FREQ_12MHZ
```

The clock reference generator (REFCLKO3) is used to provide a 1MHz clock to the CMOS sensor (Figure 1).
The clock of the CMOS sensor has been lowered to 1MHz (compared to the recommended 26.667MHz) in order to reduce the data rate and make it easier for the microcontroller to read.

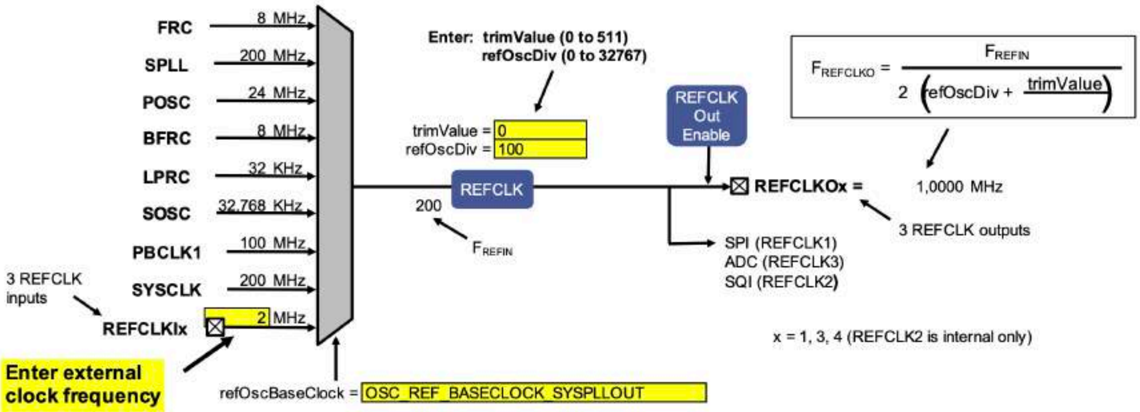


Figure 20 - Clock reference generator

With the following register settings, we can configure and enable REFCLKO3:

```
/* Set up Reference Clock 3 */
/* REFO3CON register */
/* ROSEL =  SYSCLK */
/* DIVSWEN = 1 */
/* RODIV = 100 */
REFO3CON = 0x640200;

/* REFO3TRIM register */
/* ROTRIM = 1 */
REFO3TRIM = 0x800000;

/* Enable oscillator (ON bit) and Enable Output (OE bit) */
REFO3CONSET = 0x00001000 | 0x00008000;
```

6.2 Image sensor control

The configurations and operating modes of the sensor must be set in a preliminary phase, sending data via the I²C interface.

To write a value in a register, it is necessary to transmit in sequence the address of the sensor, the address of the register and the two bytes of the value, as shown in the following diagram. The sensor address in the current configuration, is 0xB8 for write mode and 0xB9 for read mode.

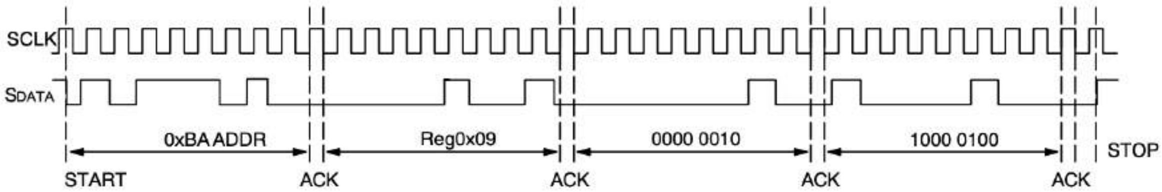


Figure 21 - Timing diagram to setting a 16 bit value (0x0284) to register 0x09

Operational mode configuration

The sensor can operate in different modes: the default provides the continuous acquisition of frames at the maximum framerate, scanned by the sensor itself. A second mode, more interesting for our application, is the "snapshot mode", which provides the acquisition of a frame and subsequent transmission only when a high pulse is received on the input signal "exposure". The following time diagram shows the waveforms of the different signals in this mode.

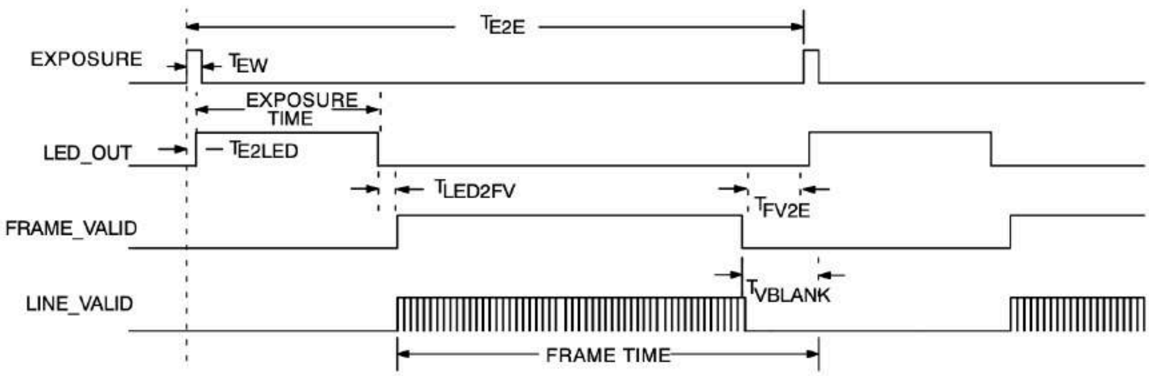


Figure 22 - Snapshot mode frame synchronization waveforms

To set the snapshot mode we need to set the value of 0x07[3:4] register to 3.

Exposure and Gain control

Figure 23 explains the exposure and gain regulation, with all involved registers. The gain represents an analog amplification factor therefore it affects the sensitivity of the sensor. The exposure value instead regulates the acquisition exposure time.

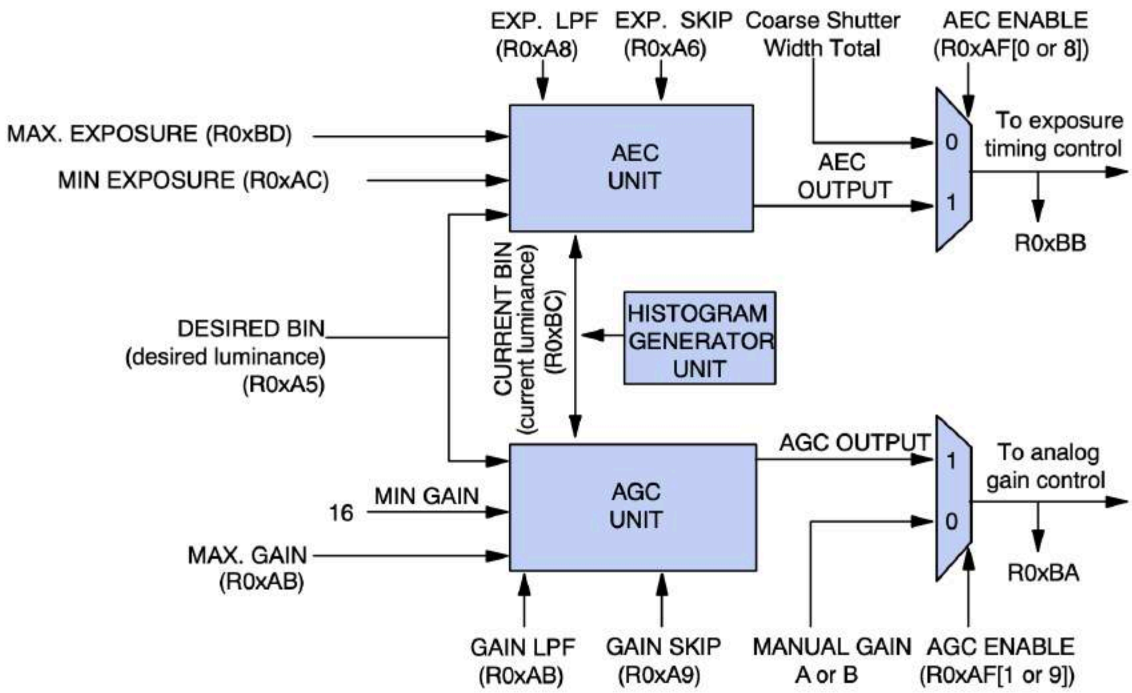


Figure 23 - AEC/AGC registers

In this first phase we will enable the Automatic Gain Control and Automatic Exposure Control units, avoiding to manually control the exposure.

Registers configuration:

```
void APS_Initialize(void){
   uint8_t i2c_data[3];

   i2c_data[0] = 0xAF;   //AEC/AGC
   i2c_data[1] = 0x00;   //AEC/AGC enable
   I2C4_Write(APS_Addr, i2c_data, sizeof(i2c_data));
   while(I2C4_IsBusy());

   i2c_data[0] = 0x07;   //Chip Control
   i2c_data[1] = 0x02;   //
   i2c_data[2] = 0x98;   //
   I2C4_Write(APS_Addr, i2c_data, sizeof(i2c_data));
   while(I2C4_IsBusy());
}
```

6.3 Full frame acquisition

As explained in Figure 22, in the snapshot mode it is necessary to generate a short pulse on the EXPOSURE signal to start the acquisition. The FRAME_VALID and LINE_VALID signals delimit the sequence of bytes, scanned by the PIXCLK signal, as shown in Figure 24.

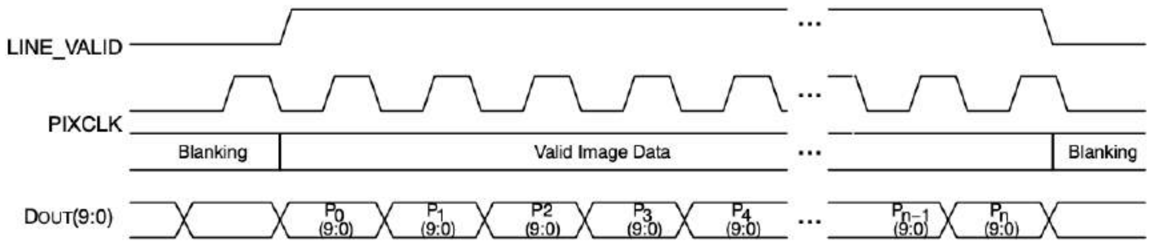


Figure 24 - Timing example of pixel data

During the acquisition process it is advisable to disable all the microcontroller interrupts, to ensure the continuity of the acquisition and preserve the data integrity.

The following function implements the standard reading of an entire frame:

```
void APS_AcquireFF(void){

    /* Save current Interrupt Enable state and Disable Interrupts. */
    unsigned int saved_state;
    saved_state = __builtin_get_isr_state();
    __builtin_disable_interrupts();

    unsigned long c=0;
    while(!PORTDbits.RD15);     // wait the frame start
    while(!PORTDbits.RD14);     // wait the first line start
    while(PORTDbits.RD15){      // cycle until the frame is valid
        while(PORTDbits.RD14){  // cycle until the line is valid
            while(!PORTDbits.RD13 && PORTDbits.RD14); // wait pixclk
            frame[c++] = PORTE >> 2 & 0xFF;    // read pixel byte
            while(PORTDbits.RD13 && PORTDbits.RD14);  // wait pixclk
        }
        while(!PORTDbits.RD14 && PORTDbits.RD15);     // wait line end
    }

    /* Set back to what was before. */
    __builtin_set_isr_state(saved_state);
}
```

6.4 Sub-window acquisition

As explained in section 4.2.1, to optimize the tracking algorithm, it is advisable to acquire a sufficient number of sub-windows of limited size, after having acquired the first full frame.

For sub-window acquisition, it is necessary to set the registers related to the coordinates of the acquisition start point (top right corner of the window) and the window size. However, as input parameters of the function it is convenient to use the coordinates of the center of the window, corresponding to the expected coordinates of the clamped star at the next acquisition (the low angular rate when the star tracker is in use ensures that the star will still be found in the window during next acquisition).

The dimensions of the sub-window must take into account the number of pixels representing the star added to twice the number of pixels corresponding to an angular displacement at the maximum rate measurable with the optical sensor (beyond which the gyroscopic sensor will be used).

The following function has been implemented:

```
void APS_AcquirePF(uint16_t x_pos, uint16_t y_pos, uint16_t wsize){
   uint8_t i2c_data[3];

   i2c_data[0] = 0x01;     // Column start register
   i2c_data[1] = (x_pos - wsize / 2) >> 8 & 0xFF;  // Column start MSB
   i2c_data[2] = (x_pos - wsize / 2) & 0xFF;       // Column start LSB
   I2C4_Write(APS_Addr, i2c_data, sizeof(i2c_data));
   while(I2C4_IsBusy());

   i2c_data[0] = 0x02;     // Row start register
   i2c_data[1] = (y_pos - wsize / 2) >> 8 & 0xFF;  // Row start MSB
   i2c_data[2] = (y_pos - wsize / 2) & 0xFF;       // Row start LSB
   I2C4_Write(APS_Addr, i2c_data, sizeof(i2c_data));
   while(I2C4_IsBusy());

   i2c_data[0] = 0x03;     // Window height register
   i2c_data[1] = wsize >> 8 & 0xFF;;   // Window height MSB
   i2c_data[2] = wsize & 0xFF;         // Window height LSB
   I2C4_Write(APS_Addr, i2c_data, sizeof(i2c_data));
   while(I2C4_IsBusy());

   i2c_data[0] = 0x04;     // Window width register
   i2c_data[1] = wsize >> 8 & 0xFF;;   // Window width MSB
```

```
    i2c_data[2] = wsize & 0xFF;         // Window width LSB
    I2C4_Write(APS_Addr, i2c_data, sizeof(i2c_data));
    while(I2C4_IsBusy());

    /* Save current Interrupt Enable state and Disable Interrupts. */
    unsigned int saved_state;
    saved_state = __builtin_get_isr_state();
    __builtin_disable_interrupts();

    unsigned long c=0;
    while(!PORTDbits.RD15);     // wait the frame start
    while(!PORTDbits.RD14);     // wait the first line start
    while(PORTDbits.RD15){      // cycle until the frame is valid
        while(PORTDbits.RD14){  // cycle until the line is valid
            while(!PORTDbits.RD13 && PORTDbits.RD14); // wait pixclk
            frame[c++] = PORTE >> 2 & 0xFF;    // read pixel byte
            while(PORTDbits.RD13 && PORTDbits.RD14);  // wait pixclk
        }
        while(!PORTDbits.RD14 && PORTDbits.RD15);     // wait line end
    }

    /* Set back to what was before. */
    __builtin_set_isr_state(saved_state);
}
```

6.5 Gyroscope sensor acquisition

To acquire the gyroscopic sensor, it was chosen to use the SPI protocol (instead I^2C), based on the availability and combination of interfaces used on the microcontroller. The necessary registers to be configured for sensor initialization and the registers to be read to acquire the angular rate values were identified from the datasheet [14].

Sensor initialization:

```
void GYRO_Initialize(void){

    uint8_t spi_set[2];

    spi_set[0] = 0x20;     // CTRL_REG1
    spi_set[1] = 0x0F;     // Enable sensor
    SPI3_Write(&spi_set, 2);

}
```

Global variables:

```
uint16_t gyro_x,gyro_y,gyro_z;
```

Function:

```
void GYRO_Read(void){

    uint8_t spi_req;
    uint8_t gyro_data[6];

    spi_req = 0xA8;
    SPI3_WriteRead(&spi_req, 1, &gyro_data[0], 1);
    spi_req = 0xA9;
    SPI3_WriteRead(&spi_req, 1, &gyro_data[1], 1);
    spi_req = 0xAA;
    SPI3_WriteRead(&spi_req, 1, &gyro_data[2], 1);
    spi_req = 0xAB;
    SPI3_WriteRead(&spi_req, 1, &gyro_data[3], 1);
    spi_req = 0xAC;
    SPI3_WriteRead(&spi_req, 1, &gyro_data[4], 1);
    spi_req = 0xAD;
    SPI3_WriteRead(&spi_req, 1, &gyro_data[5], 1);

    gyro_x = (gyroXH << 8) + gyroXL;
    gyro_y = (gyroYH << 8) + gyroYL;
    gyro_z = (gyroZH << 8) + gyroZL;

}
```

A temperature sensor is also available in the same component, so this function is useful to read this value too:

```
uint8_t temp;        // temperature global variable

void TEMP_read(void){

    uint8_t spi_req;

    spi_req = 0xA6;
    SPI3_WriteRead(&spi_req, 1, &temp, 1);

}
```

7 Prototyping and testing

7.1 Prototype realization

A first prototype of this project was developed by printing the PCBs from an industrial prototyping service (starting from the Gerber files).
The assembly of the SMD components was carried out manually in the laboratory, with the aid of classic and hot air welder.

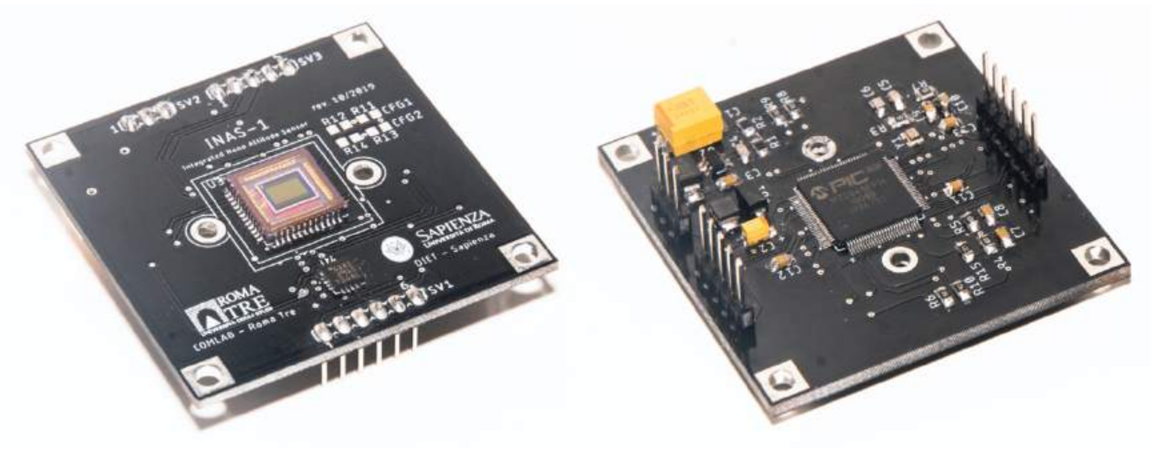


Figure 25 - Main module prototype – Top view (left) and bottom view (right)

Figure 26 - Complete prototype, with lens, mounted on adapter board

7.2 Preliminary tests

Power supply

The output voltage of the power supply stage was tested, without any loads, before the assembly of all other device, to prevent the possible damage of more valuable components (microcontroller, CMOS sensor and gyroscope sensor).

During the nominal operations, the voltage ripple (measured with an oscilloscope) is limited to 48.8mV, as shown in the following oscilloscope measurement:

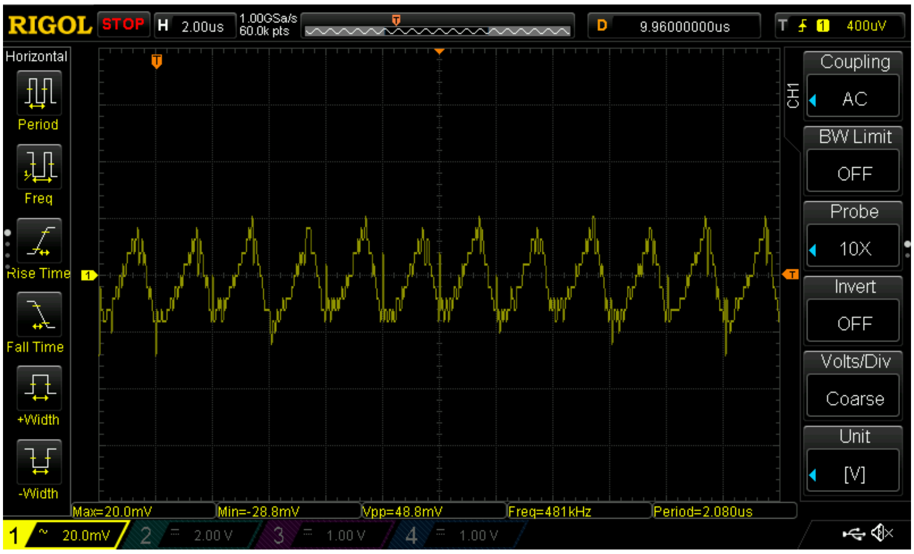


Figure 27 - 3.3V power supply voltage ripple (nominal operation)

Reference clock

The following measurement shows the clock reference of 1MHz generated by the microcontroller for the CMOS sensor:

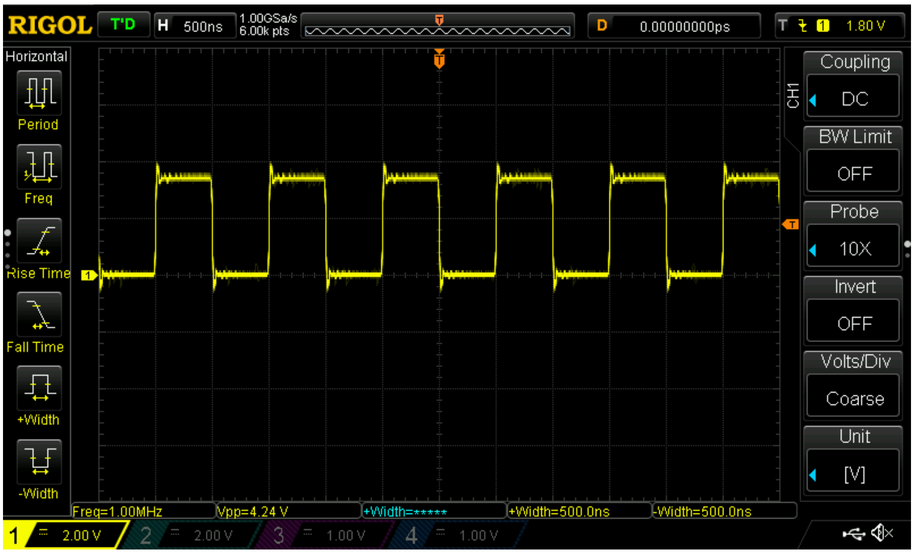


Figure 28 - 1MHz clock reference

Communication and control signals

A measurement of FRAME_VALID and LINE_VALID signals shows that the duration of a single frame (at test frequency of 1MHz) is 406 ms, with 38 ms pause between 2 frame transmission (which includes the exposure time):

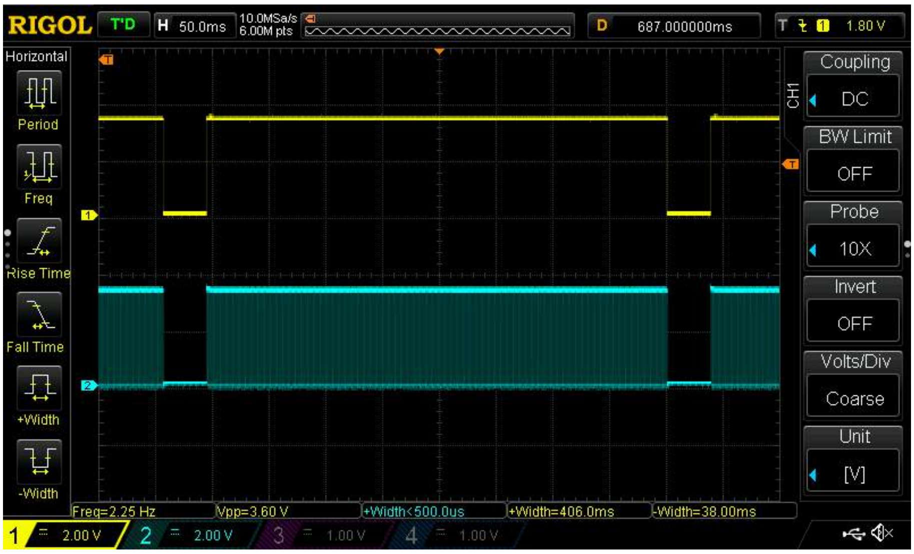


Figure 29 - FRAME_VALID signal (yellow) and LINE_VALID signal (blue)

The following waveforms show an example of image data output (bit D0) with the clock signal:

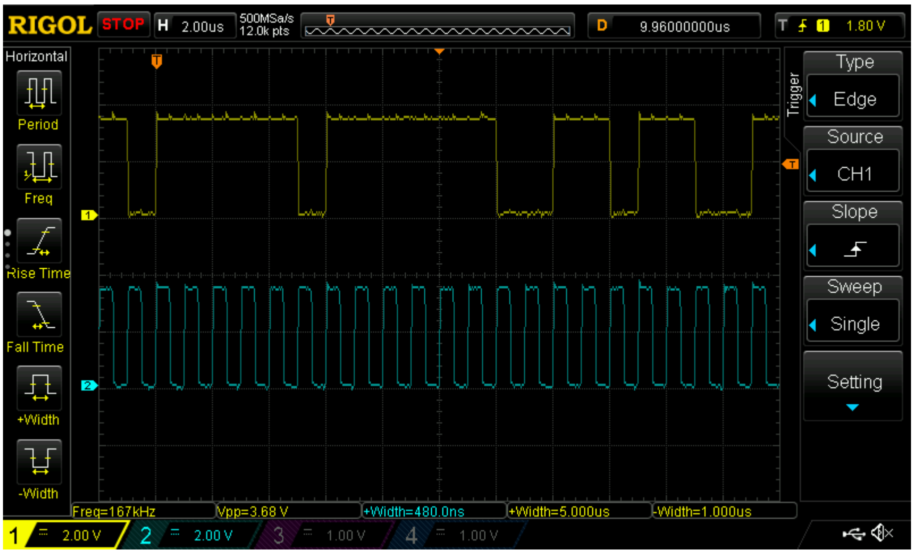


Figure 30 - Image data example (yellow) and PIXCLK signal (blue)

A first test of transmission of 2-byte register configuration via I^2C shows the waveform of CLOCK and DATA signals:

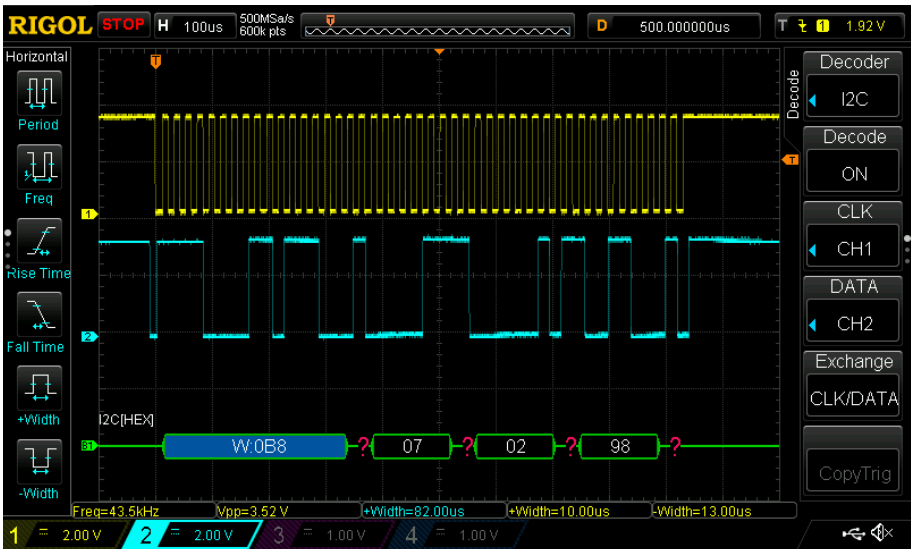


Figure 31 - I2C clock (yellow) and data (blue) waveforms

7.3 Acquisition test

A main function has been developed for testing purposes that read a simple command (one char) from external UART (DATA I/F) and invoke different functions. The acquired data is transferred on the UART (at 115200 baud/s) and received by a computer through a USB/serial converter.

The following code was used on the microcontroller:

```
void main(void){

   char read_msg;

   APS_Initialize();
   GYRO_Initialize();

   while(true){
      UART1_Read(&read_msg,1);
      switch(read_msg){
         case ‘F’:
            APS_AcquireFF();
            UART1_Write(&frame,360960);
            while(UART1_WriteIsBusy());
            break;
         case ‘P’:
            APS_AcquirePF(300,200,50);
            UART1_Write(&frame,2500);
            while(UART1_WriteIsBusy());
            break;
         case ‘G’:
            GYRO_Read();
```

```
            UART1_Write(&gyro_data,6);
            while(UART1_WriteIsBusy());
            break;
         case ‘T’:
            TEMP_Read();
            UART1_Write(&temp,2);
            while(UART1_WriteIsBusy());
            break;
         default:
            break;
   }
}
```

In case of CMOS full frame acquisition, the sequence of bytes received in this way was acquired with the Matlab software, in which the function defined below memorizes the entire sequence as a byte array, rearranged bytes in a matrix form of size 752x480 and displayed as an image.

```
function acquire
    s = serialport("COM5",115200);
    write(s,’F’,”char”);
    data = read(s,360960,"uint8");
    img = reshape(data,[752 480]);
    img = img';
    img = imrotate(img,180);
    figure,imshow(img,[]);
end
```

In Figure 32 is showed a test set for image acquisition and focus, using a test graphic target. Figure 33 shows the result image.

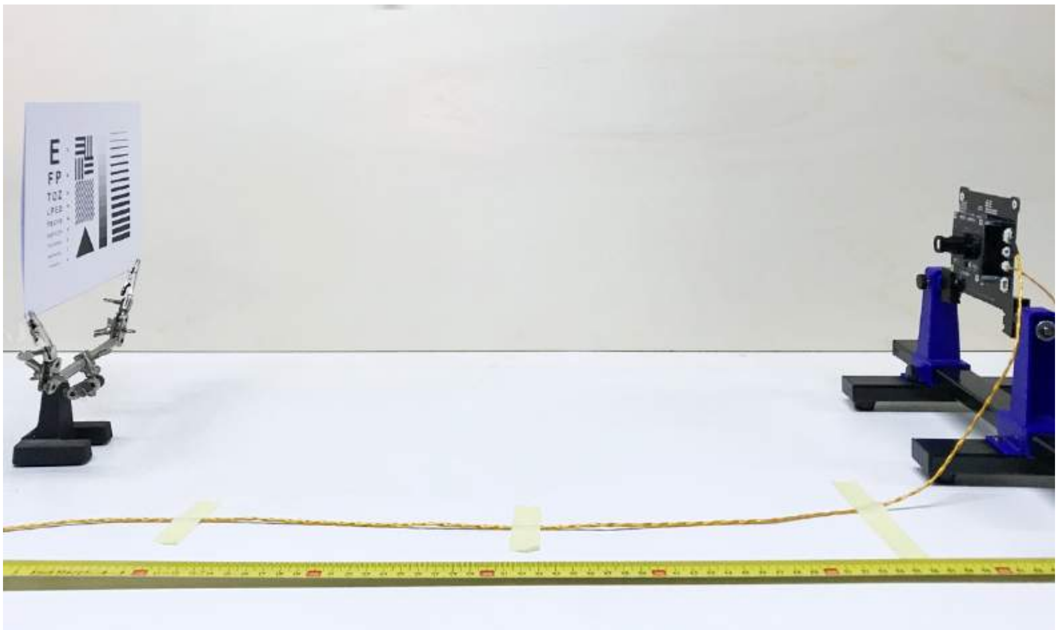


Figure 32 - Test bench for image acquisition

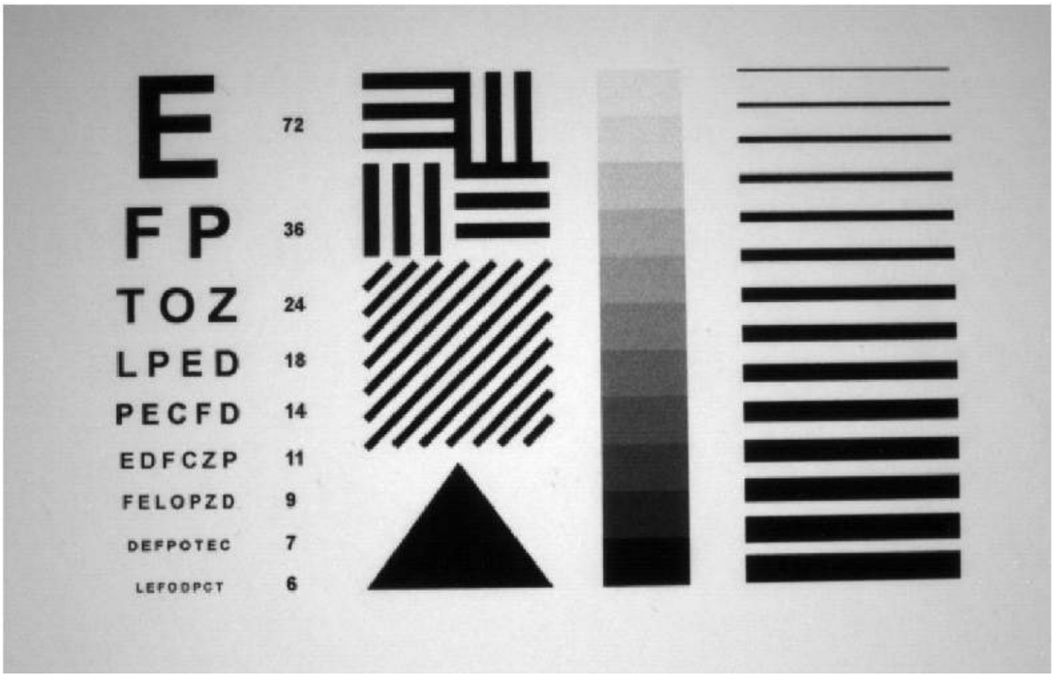


Figure 33 - First test image acquisition

8 Conclusions

8.1 Achieved goals

The realization of this project has satisfied the initial objectives and requirements, although some design complications have been identified.

The most important complication concerns the image acquisition speed. We verified that, with the current algorithm with cyclic reading of the parallel interface, it is not possible to drive the CMOS sensor at a frequency higher than 1MHz, compared to the recommended 26.667MHz. This mainly involves slowing down the acquisition of an image, which takes 445ms in the case of a full frame. Considering the angular resolution calculated (paragraph 4.2.2) of 0.021 degrees, the maximum angular rate measurable by the optical sensor will be less than 0.047 deg / s (47mdps), not knowing yet the processing times of the algorithm necessary for star tracking.

Considering the minimum angular rate measurable by the gyro rate sensor of 8.74mdps, we can claim to be able to cover all the possible angular rates, without discontinuity. In fact, by measuring the time between two frames in which a star moves only one pixel, corresponding to 0.021 degrees, we are able to calculate a very slow gyro rate, limited only by the size of the time variable.

The limited size (50x50x24 mm) and mass (19g, 54g with CubeSat adapter board) characteristics are compatible with CubeSat integration and in general micro and nano satellites.

The production cost (less than 100 euros for a single prototype) may lead to a final product (adding additional electronics and software) that is still competitive compared to existing solutions on the market, as shown in next table.

Product	**Dimensions**	**Mass**	**Price**
NST-3 Nano Star Tracker[3]	50x50x102mm	165g	30,000€
MAI-SS Space Sextant[4]	50x50x47mm	170g	32,500$

[3] https://www.cubesatshop.com/product/nst-3-nano-star-tracker/
[4] https://www.cubesatshop.com/product/mai-ss-space-sextant/

KU Leuven Star Tracker[5]	45x50x95	250g	30,000€

Table 10 - Comparison of commercial star tracker for small satellite

8.2 Future developments

The next phase of this project involves the implementation of a star tracking algorithm for the calculation of angular rates starting from the acquired images.
A management algorithm must be able to establish at what time it is necessary to start acquiring images and triggering star tracking functions, for angular rates below a threshold.
It may be possible to optimize the reading algorithm of the parallel interface of the CMOS sensor, using a DMA (Direct Memory Access) function instead of the currently implemented cyclic reading, in order to increase the acquisition speed of the sensor, improving its performance. During the thesis we have received some support by **Microchip International** to explore this possibility but time constraints have hampered the implementation of these tests.
It would also be possible to insert into the internal EEPROM (Program Memory) a database of known stars, to add the autonomous absolute attitude determination capability to the system, making it possible to determine the absolute attitude of the satellite. The maximum available program memory is 2048KB.
In the current firmware development, the DATA I/F (external interface) has been enabled only with UART protocol. However, the hardware configuration described in paragraph 5.5 allows the alternative selection of the I2C and SPI interfaces. Reading the status of the CFG1 and CFG2 inputs at power up, it is possible to recall the 3 different software configurations of the interfaces, making compatibility with different systems possible.

[5] https://www.cubesatshop.com/product/kul-star-tracker/

Bibliography

[1] S. Scibelli, V. Ferrara and F. Bernardini, "Low-cost stellar sensor for attitude control of small satellites," in *41st PhotonIcs and Electromagnetics Research Symposium*, Rome, Italy, 2019.

[2] G. Konecny, "Small Satellites - A Tool for Earth Observation?," Oct 2016. [Online]. Available: http://www.cartesia.org/geodoc/isprs2004/comm4/papers/428.pdf.

[3] The CubeSat Program, Cal Poly SLO, "CubeSat Design Specification," Feb 2014. [Online]. Available: http://www.cubesat.org/s/cds_rev13_final2.pdf. [Accessed Jul 2019].

[4] NASA CubeSat Launch Initiative, "CubeSat 101: Basic Concepts and Processes for First-Time CubeSat Developers," Oct 2017. [Online]. Available: https://www.nasa.gov/sites/default/files/atoms/files/nasa_csli_cubesat_101_508.

[5] V. Francois-Lavet, *Attitude and Determination Control Systems for the OUFTI nanosatellites,* 2010.

[6] F. L. Markley and J. L. Crassidis, Fundamentals of Spacecraft Attitude Determination and Control, Springer New York, 2014.

[7] ECSS, *Star sensor terminology and performance specification,* ECSS-E- ST-60-20C (Rev.2), 2019.

[8] *IEEE Std 1431–2004 Coriolis Vibratory Gyroscopes.*

[9] Microchip Technology, "AN656 In-Circuit Serial Programming (ICSP™) of Calibration Parameters Using a PICmicro™ Microcontroller," 1997. [Online]. Available: http://ww1.microchip.com/downloads/en/appnotes/00656b.pdf.

[10] ON Semiconductor, "NOIP1SN1300A - PYTHON 1.3/0.5/0.3 MegaPixels Global Shutter CMOS Image Sensors," Apr 2018 (Rev. 5). [Online]. Available: https://www.onsemi.com/pub/Collateral/NOIP1SN1300A-D.PDF. [Accessed Jul 2019].

[11] ON Semiconductor, "MT9V034 1/3-Inch Wide-VGA CMOS Digital Image Sensor," Jan 2017. [Online]. Available: https://www.onsemi.com/pub/Collateral/MT9V034-D.PDF. [Accessed Jul 2019].

[12] Edmund Optics, "Understanding Focal Length and Field of View," [Online]. Available: https://www.edmundoptics.com/knowledge-center/application-notes/imaging/understanding-focal-length-and-field-of-view/. [Accessed Jul 2019].

[13] Edmund Optics, "Imaging Optics - Catalog," [Online]. Available: https://quad2.mydigitalpublication.com/publication/?i=576918. [Accessed Jul 2019].

[14] ST Microelectronics, "I3G4250DTR MEMS motion sensor: 3-axis digital output gyroscope," Apr 2015 (Rev.2). [Online]. Available: https://www.st.com/resource/en/datasheet/i3g4250d.pdf. [Accessed Jul 2019].

[15] Microchip Technology, "PIC32MZ Embedded Connectivity with Floating Point Unit (EF) Family," Mar 2019 (Rev. F). [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/PIC32MZ-EF-%20Family-DS60001320F.pdf. [Accessed Jul 2019].

[16] Automotive Electronics Council, *Failure Mechanism Based Stress Test Qualification for Integrated Circuits,* AEC-Q100, 2017.

[17] Microchip Technology, "MCP16331 - High-Voltage Input Integrated Switch Step-Down Regulator," Dec 2016 (Rev. C). [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/20005308C.pdf. [Accessed Sep 2019].

Appendix A – MT9V034 specifications

Key Performance Parameters

Parameter	Value
Optical Format	1/3-inch
Active Imager Size	4.51 mm (H) x 2.88 mm (V) 5.35 mm diagonal
Active Pixels	752 H x 480 V
Pixel Size	6.0 x 6.0 µm
Color Filter Array	Monochrome or color RGB Bayer
Shutter Type	Global Shutter
Maximum Data Rate Master Clock	27 Mp/s 27 MHz
Full Resolution	752 x 480
Frame Rate	60 fps (at full resolution)
ADC Resolution	10-bit column-parallel
Responsivity	4.8 V/lux-sec (550 nm)
Dynamic Range	>55 dB linear; >100 dB in HDR mode
Supply Voltage	3.3 V ± 0.3 V (all supplies)
Power Consumption	<160 mW at maximum data rate (LVDS disabled); 120 µW standby power at 3.3 V
Operating Temperature	–30°C to + 70°C ambient
Packaging	48-pin CLCC

Table 11 - MT9V034 Key performance parameters

Block Diagram

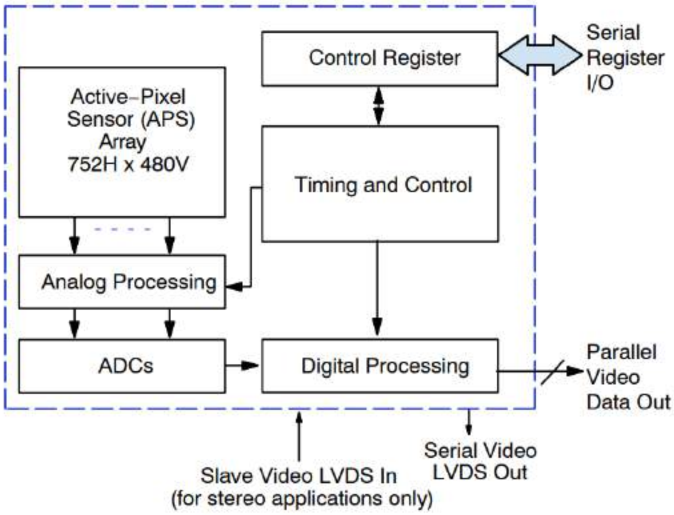


Figure 34 - MT9V034 block diagram

Dimensions:

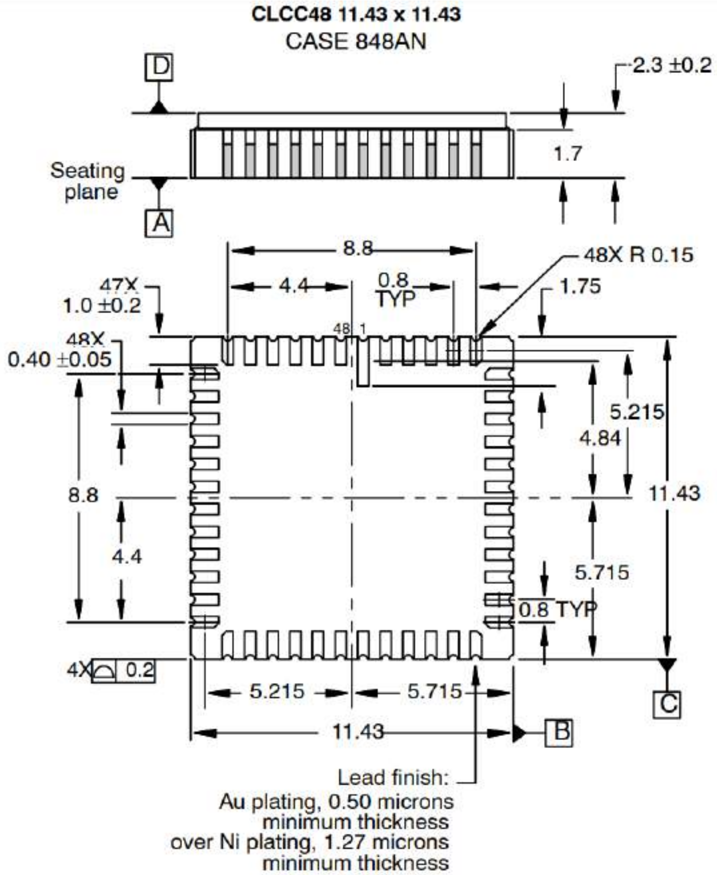


Figure 35 - MT9V034 Dimensions

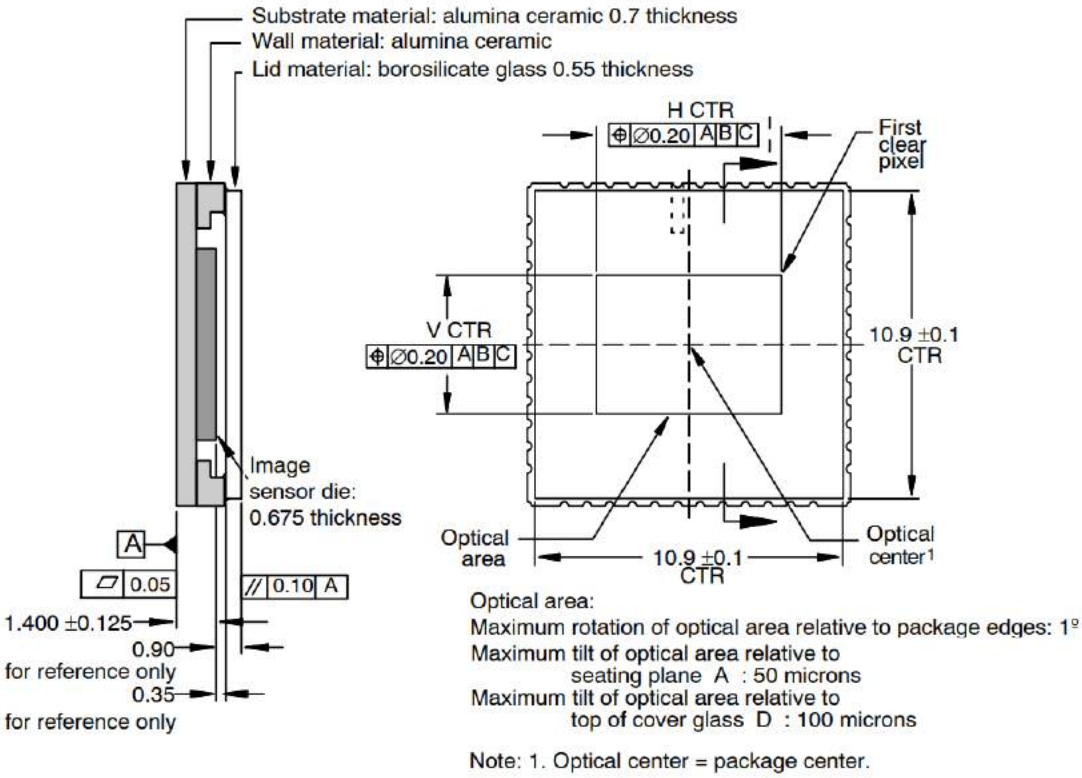


Figure 36 - MT9V034 Section and optical area position

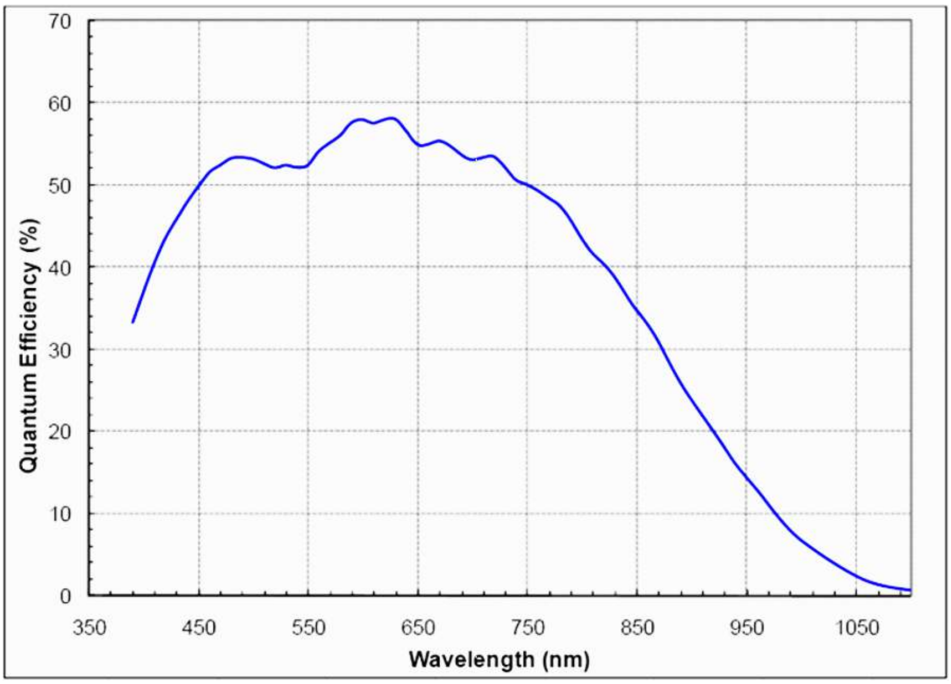


Figure 37 - MT9V034 Typical Quantum Efficiency – Monochrome

Appendix B – I3G4250DTR specifications

Features:

- Wide supply voltage: 2.4 V to 3.6 V
- Selectable full scale (245/500/2000 dps)
- I^{2}C /SPI digital output interface
- 16-bit rate value data output
- 8-bit temperature data output
- Two digital output lines (interrupt and data ready)
- Integrated low- and high-pass filters with user-selectable bandwidth
- Ultra-stable over temperature and time
- Low-voltage-compatible IOs (1.8 V)
- Embedded power-down and sleep mode
- Embedded temperature sensor
- Embedded FIFO
- High shock survivability
- Extended operating temperature range (-40 °C to +85 °C)
- ECOPACK®, RoHS and “Green” compliant

Block Diagram

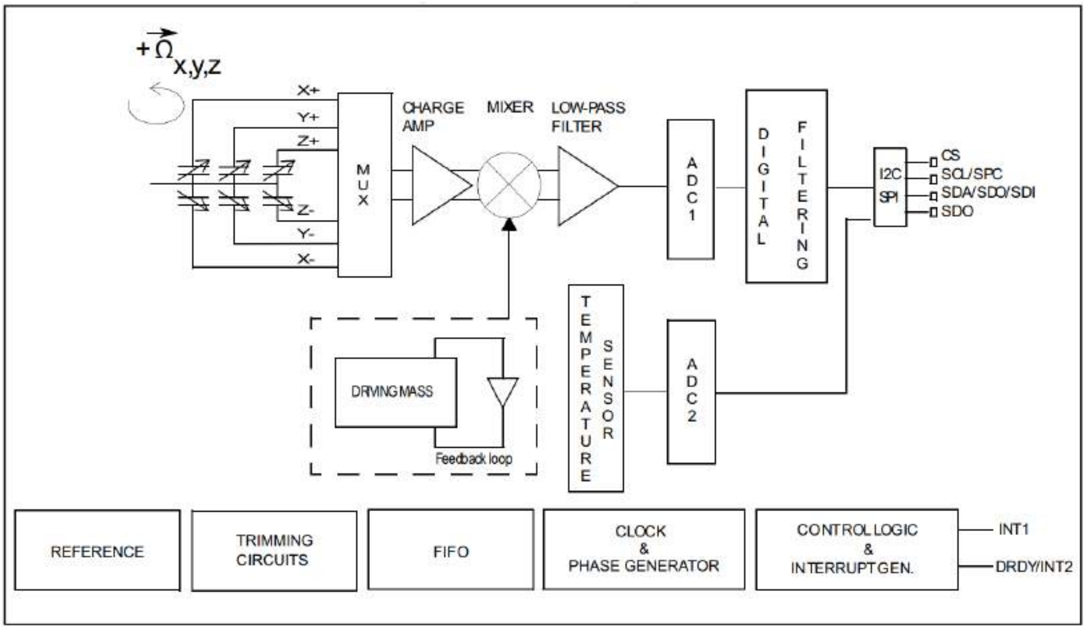


Figure 38 - I3G4250DTR block diagram

Mechanical characteristics:

Symbol	Parameter	Test condition	Min.(1)	Typ.(2)	Max.(1)	Unit
FS	Measurement range(3)	User-selectable		±245		dps
				±500		
				±2000		
So	Sensitivity(4)	FS = 245 dps	7.4	8.75	10.1	mdps/digit
		FS = 500 dps	14.8	17.50	19.8	
		FS = 2000 dps	59.2	70	79.3	
SoDr	Sensitivity change vs. temperature	From -40°C to +85°C		±2		%
DVoff	Digital zero-rate level(4)	FS = 245 dps	-25	±10	+25	dps
		FS = 500 dps	-37.5	±15	+37.5	
		FS = 2000 dps	-187.5	±75	+187.5	
OffDr	Zero-rate level change vs. temperature	FS = 245 dps		±0.03		dps/°C
		FS = 2000 dps		±0.04		
NL	Non linearity(3)	Best fit straight line	-5	0.2	+5	% FS
DST	Self-test output change	FS = 245 dps		130		dps
		FS = 500 dps		200		
		FS = 2000 dps		530		
Rn	Rate noise density	BW = 50 Hz		0.03		dps/ sqrt(Hz)
ODR	Digital output data rate			105/208/ 420/840		Hz
Top	Operating temperature range		-40		+85	°C

1. Minimum and maximum values are not guaranteed; based on characterization data.
2. Typical specifications are not guaranteed; typical values at +25 °C.
3. Guaranteed by design.
4. Min/Max values for DVoff are across temperature (-40°C to 85°C) and after MSL3 preconditioning. Based on characterization data. Not guaranteed and not tested in production.

Table 12 - I3G4250DTR mechanical characteristics

Appendix C – PIC32MZ2048EFM100 specifications

PIC32MZ EF Family Block Diagram:

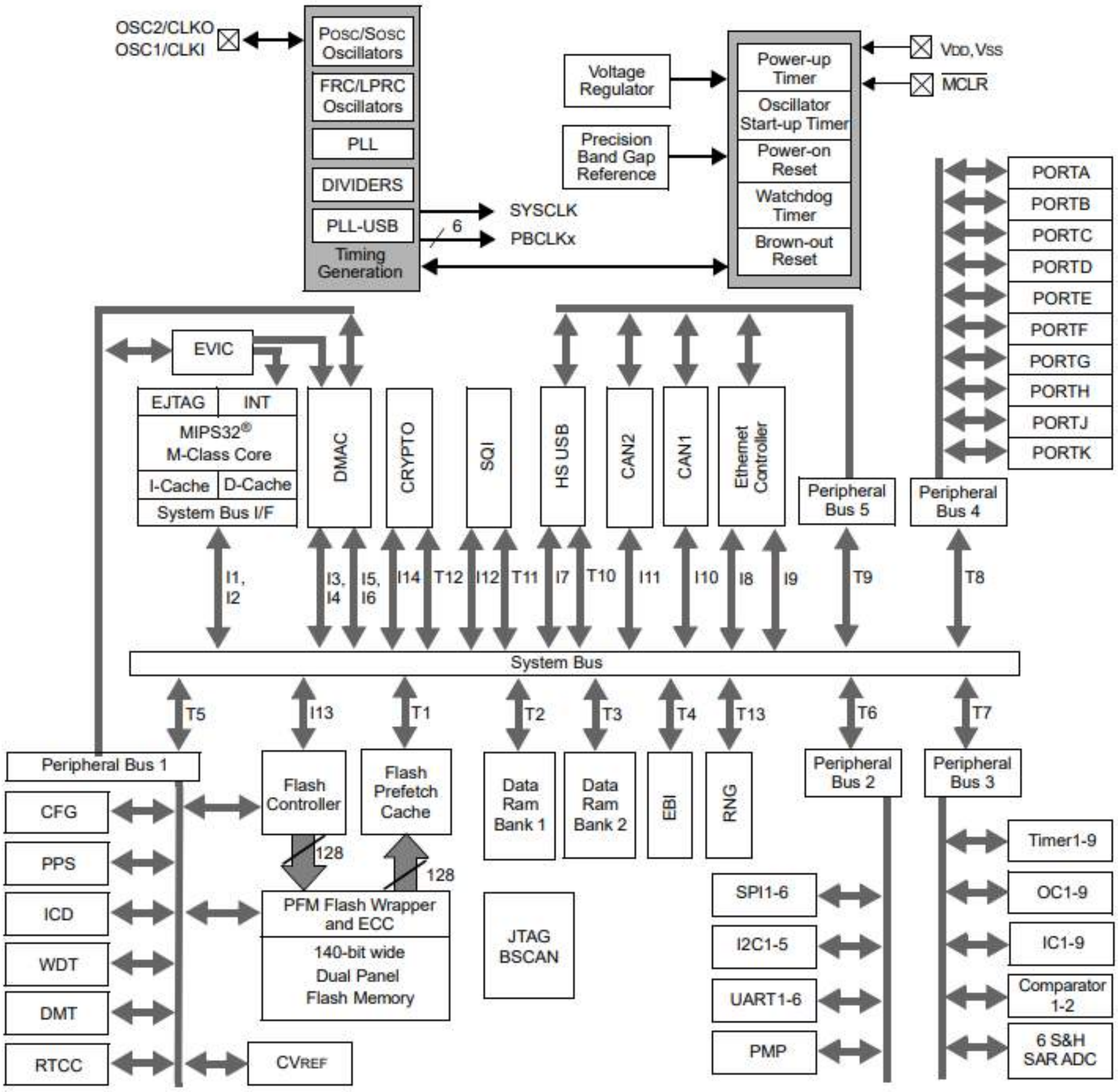


Figure 39 - PIC32MZ EF Family Block Diagram

Key Features

- 200 MHz/330 DMIPS, MIPS M-class core
- DSP-enhanced core:
  - Four 64-bit accumulators
  - Single-cycle MAC, saturating and fractional math
  - IEEE 754-compliant
- Dual Panel Flash for live update support

- FPU for fast single- and double-precision math
- 12-bit, 18 MSPS, 40-channel ADC module
- Memory Management Unit for optimum embedded OS execution
- microMIPS mode for up to 35% code compression
- CAN, UART, I^2C, PMP, EBI, SQI & Analog Comparators
- SPI/I^2S interfaces for audio processing and playback
- Hi-Speed USB 2.0 Device/Host/OTG
- 10/100 Mbps Ethernet MAC with MII and RMII interface
- Crypto Engine with a RNG for data encryption/decryption and authentication (AES, 3DES, SHA, MD5, and HMAC)
- Temperature Range: - 40°C to 85°C; - 40°C to 125°C
- AEC-Q100 Qualified
    - Grade 1
    - - 40°C to 125°C

Microcontroller Features

- Operating voltage range of 2.2V to 3.6V
- 2MB Flash memory (plus an additional 160 KB of Boot Flash)
- 512KB SRAM memory
- microMIPS mode for up to 35% smaller code size
- DSP-enhanced core:
    - Four 64-bit accumulators
    - Single-cylce MAC, saturating and fractional math
    - IEEE 754-compliant
- FPU for fast single- and double-precision math
- Code-efficient (C and Assembly) architecture
- Low-power management modes (Idle and Sleep)

Peripheral Features

- 50 MHz External Bus Interface (EBI)
- 50 MHz Serial Quad Interface (SQI)
- Peripheral Pin Select (PPS) functionality to enable function remap
- 8 channels of hardware programmable DMA and 18 channels of dedicated DMA with automatic data size detection

- Six UART modules (25 Mbps): Supports LIN 1.2 and IrDA protocols
    - Two CAN modules 2.0B Active with DeviceNet addressing support
    - Six 4-wire SPI modules (50 Mbps)
    - SQI configurable as an additional SPI module (50 MHz)
    - Five I^2C modules (up to 1 Mbaud) with SMBus support
    - Parallel Master Port (PMP)
    - Hardware Real-Time Clock and Calendar (RTCC)
    - Nine 16-bit Timers/Counters (four 16-bit pairs combine to create four 32-bit timers)
    - Nine Capture inputs and Nine Compare/PWM outputs
    - Audio/Graphics/Touch HMI Features
- Graphics interface: EBI or PMP
- Audio data communication: I2S, LJ, RJ, USB
- Audio data control interface: SPI and I^2C ™
- Audio data master clock: Fractional clock frequencies with USB synchronization

Advanced Analog Features

- 12-bit ADC Module:
- 18 Msps rate with six Sample and Hold (S&H) circuits (five dedicated and one shared)
- Up to 40 analog inputs
- Can operate during sleep and idle modes
- Multiple trigger sources
- Six digital comparators and six digital filters
- Two analog comparators with 32 programmable voltage references
- Temperature sensor with ±2°C accuracy

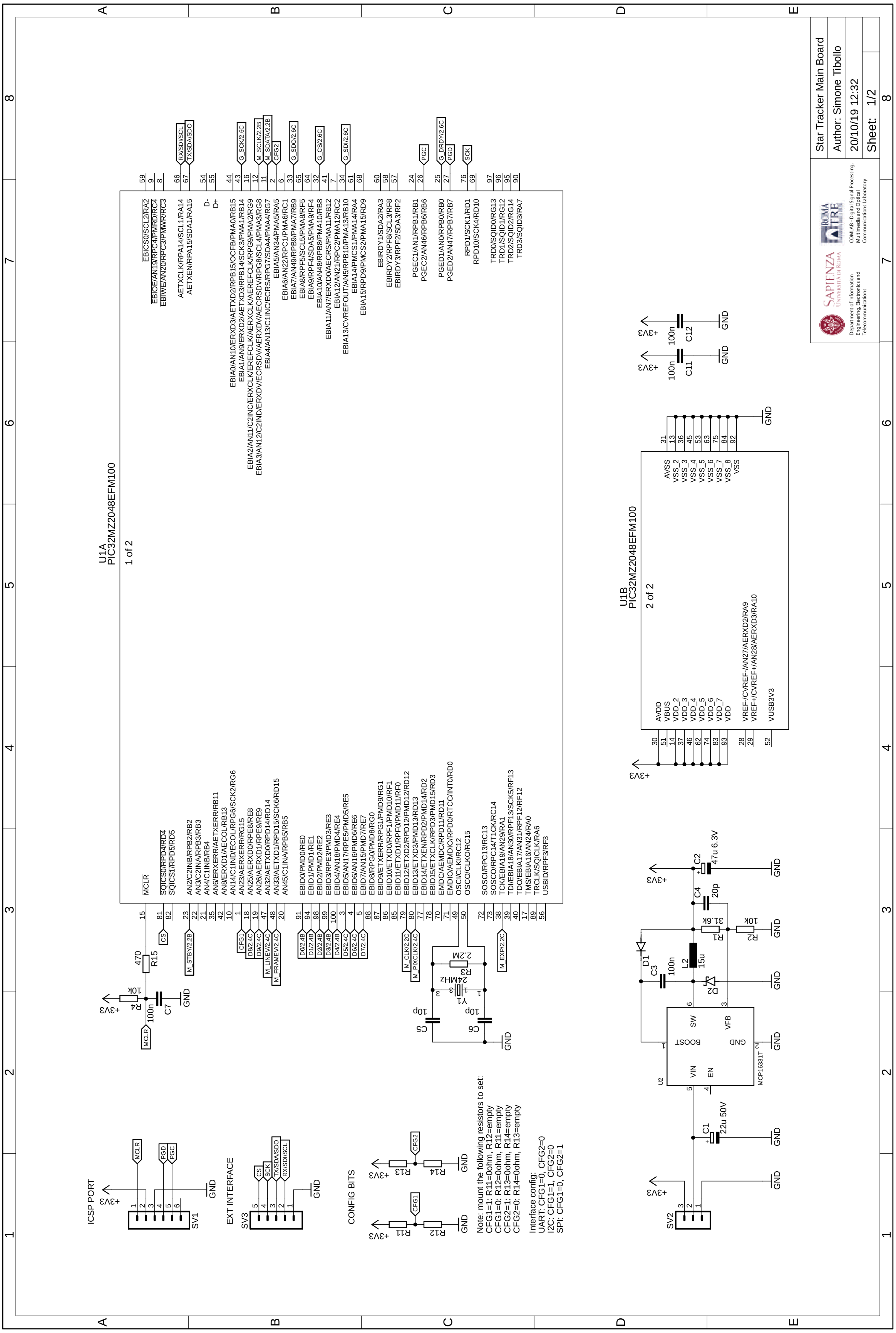
Debugger Development Support

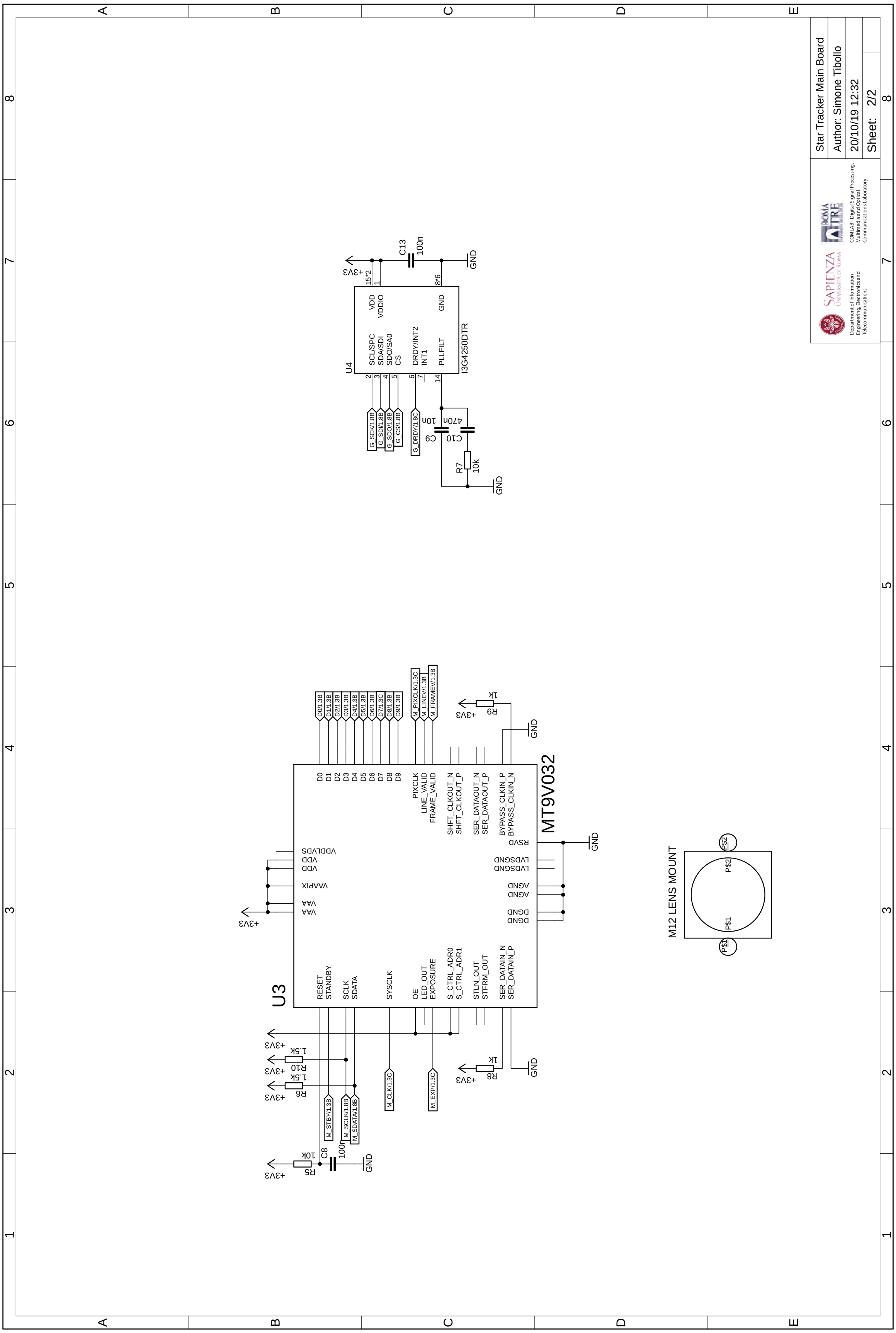
- In-circuit and in-application programming
- 4-wire MIPS® Enhanced JTAG interface
- Unlimited program and 12 complex data breakpoints
- IEEE 1149.2-compatible (JTAG) boundary scan
- Non-intrusive hardware-based instruction trace

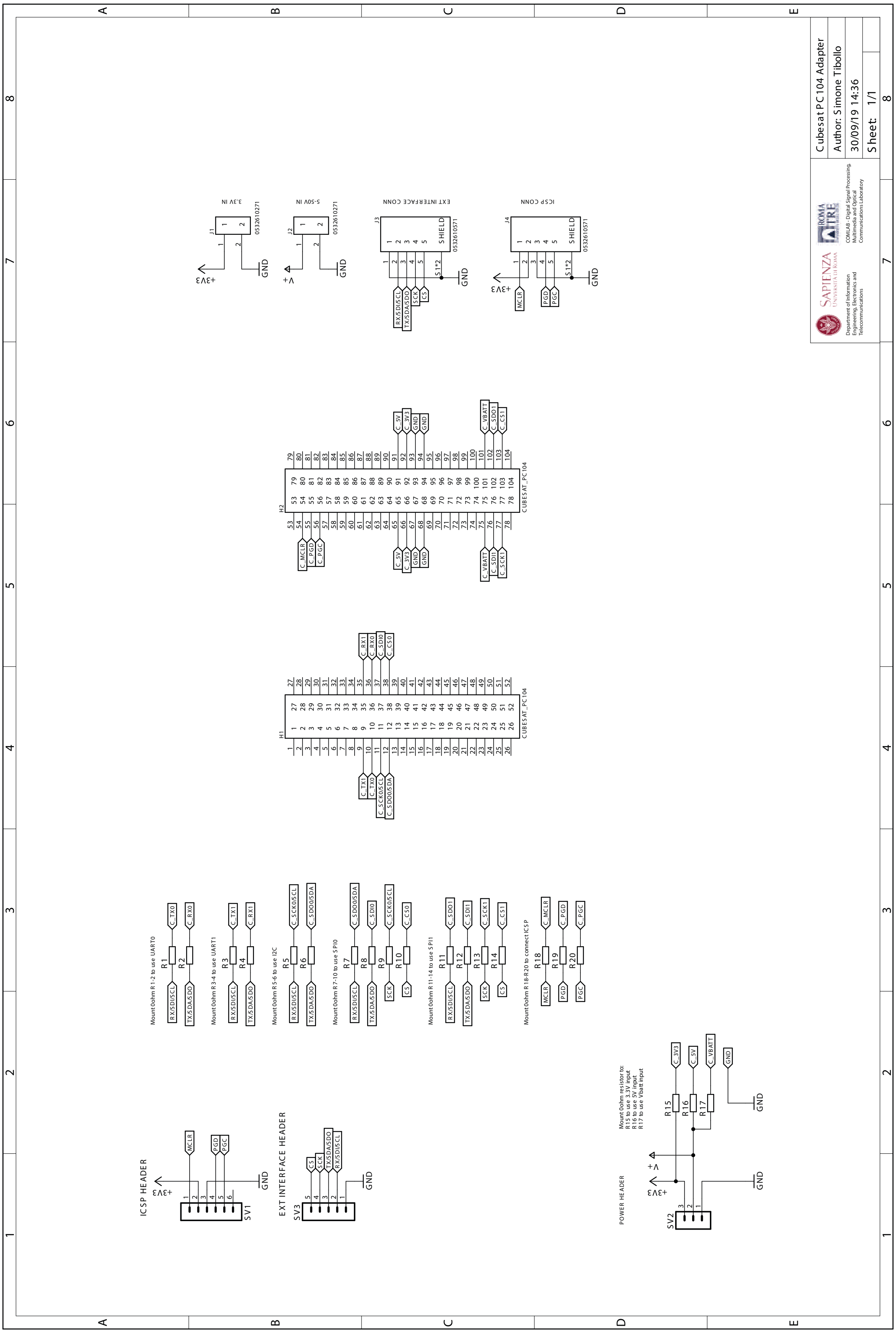
Integrated Software Libraries and Tools

- MPLAB Harmony - PIC32 software development framework
- C/C++ compiler with native DSP/fractional and FPU support
- TCP/IP, USB, Graphics and mTouch middleware
- MFi, Android and Bluetooth audio frameworks
- RTOS Kernels, Express Logic, ThreadX, FreeRTOS, OPENRTOS, Micriμm, μC/OS and SEGGER embOS

Appendix D – Schematics




Star Tracker Main Board
Author: Simone Tibollo
20/10/19 12:32
Sheet: 2/2
U3
MT9V032
U4
I3G4250DTR
M12 LENS MOUNT


Cubesat PC104 Adapter
Author: Simone Tibollo
30/09/19 14:36
Sheet: 1/1
SAPIENZA UNIVERSITÀ DI ROMA
Department of Information Engineering, Electronics and Telecommunications
ROMA TRE
COMLAB - Digital Signal Processing, Multimedia and Optical Communications Laboratory
ICSP HEADER
EXT INTERFACE HEADER
POWER HEADER
Mount 0ohm R1-2 to use UART0
Mount 0ohm R3-4 to use UART1
Mount 0ohm R5-6 to use I2C
Mount 0ohm R7-10 to use SPI0
Mount 0ohm R11-14 to use SPI1
Mount 0ohm R18-R20 to connect ICSP
Mount 0ohm resistor to:
R15 to use 3.3V input
R16 to use 5V input
R17 to use Vbatt input
CUBESAT_PC104
3.3V IN
5-50V IN
EXT INTERFACE CONN
ICSP CONN
0532610271
0532610571
SHIELD
+3V3
V+
GND

Appendix E – Bills of Material

Main board parts:

Part	Value	Package
C1	22u 50V	D/7343-31W
C2	47u 6.3V	A/3216-18W
C3	100n	C0805
C4	20p	C0805
C5	10p	C0805
C6	10p	C0805
C7	100n	C0805
C8	100n	C0805
C9	10n	C0805
C10	470n	C0805
C11	100n	C0805
C12	100n	C0805
C13	100n	C0805
D1	1A 400V	DO214AC
D2	100V 1A Schottky	DO214AA
L2	15u	L3230M
R1	31.6k	R0805
R2	10k	R0805
R3	2.2M	R0805
R4	10k	R0805
R5	10k	R0805
R6	1.5k	R0805
R7	10k	R0805
R8	1k	R0805
R9	1k	R0805
R10	1.5k	R0805
R11	0 (optional)	R0805
R12	1 (optional)	R0805
R13	2 (optional)	R0805
R14	3 (optional)	R0805
R15	470	R0805
SV1	6 pin male header (0.100" pitch)	MA06-1
SV2	3 pin male header (0.100" pitch)	MA03-1
SV3	5 pin male header (0.100" pitch)	MA05-1
U1	PIC32MZ2048EFM100	TQFP100_12X12MC-M
U2	MCP16331T	SOT95P280X145-6N

U3	MT9V032	CLCC-48
U4	I3G4250DTR	PQFN65P400X400X110N
Y1	24MHz	OSCCC160X200X50-4N
	M12 (S-Mount) 16mm f/2 1/3" lens	Edmund Optics 66-894
	M12 Lens Holder for Camera Boards	Edmund Optics 66-382

Table 13 - Main board part list

Adapter board parts:

Part	Value	Package
H1	PC104 (4x26 0.100" pitch) header	
J1	Picoblade 1.25mm SMD R/A 2pin connector	MOLEX_0532610271
J2	Picoblade 1.25mm SMD R/A 2pin connector	MOLEX_0532610271
J3	Picoblade 1.25mm SMD R/A 5pin connector	MOLEX_0532610571
J4	Picoblade 1.25mm SMD R/A 5pin connector	MOLEX_0532610571
R1	0 (optional)	R0805
R2	0 (optional)	R0805
R3	0 (optional)	R0805
R4	0 (optional)	R0805
R5	0 (optional)	R0805
R6	0 (optional)	R0805
R7	0 (optional)	R0805
R8	0 (optional)	R0805
R9	0 (optional)	R0805
R10	0 (optional)	R0805
R11	0 (optional)	R0805
R12	0 (optional)	R0805
R13	0 (optional)	R0805
R14	0 (optional)	R0805
R15	0 (optional)	R0805
R16	0 (optional)	R0805
R17	0 (optional)	R0805
R18	0 (optional)	R0805
R19	0 (optional)	R0805
R20	0 (optional)	R0805
SV1	6 pin female header (0.100" pitch)	MA06-1
SV2	3 pin female header (0.100" pitch)	MA03-1
SV3	5 pin female header (0.100" pitch)	MA05-1

Table 14 - Adapter board part list

Appendix F - PCB Layouts

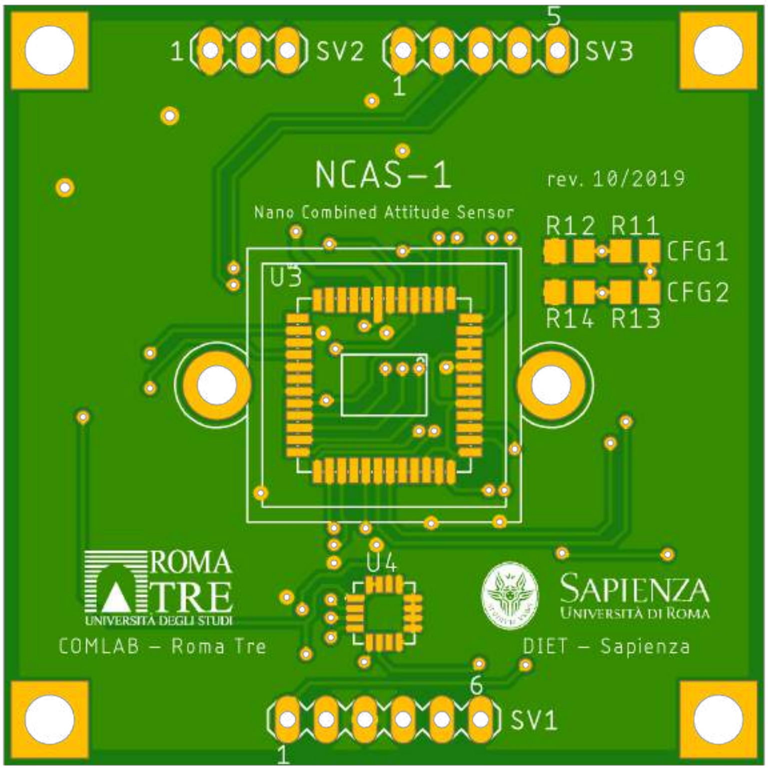


Figure 40 - Main Board - Top layer (render)

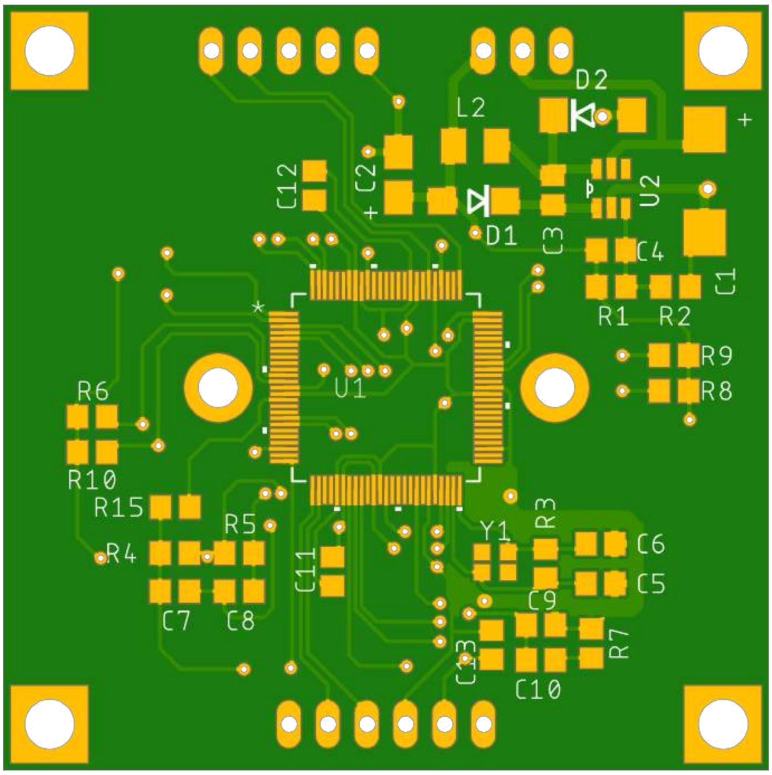


Figure 41 - Main Board - Bottom layer (render)

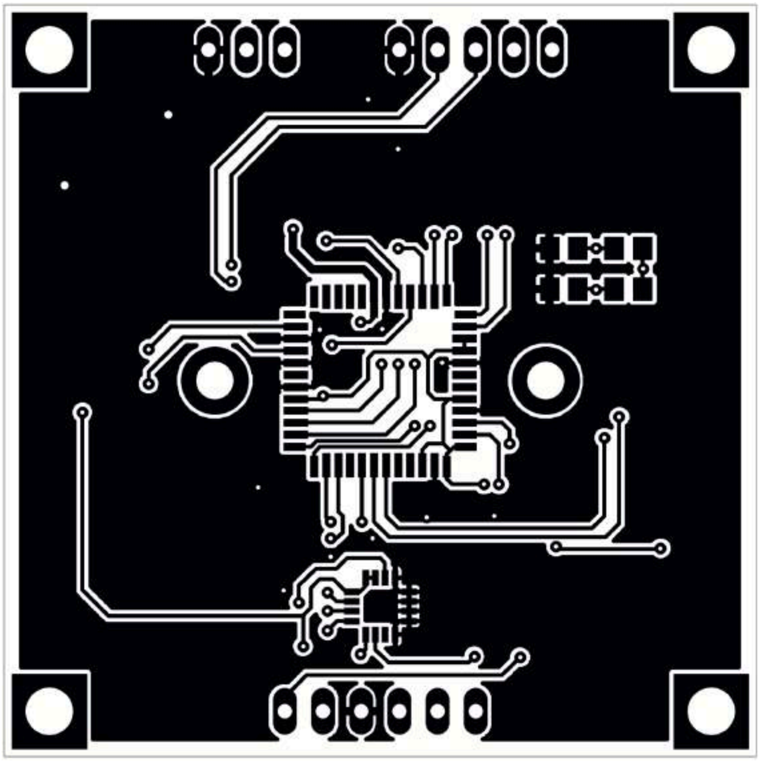


Figure 42 - Main board - Top layer (copper)

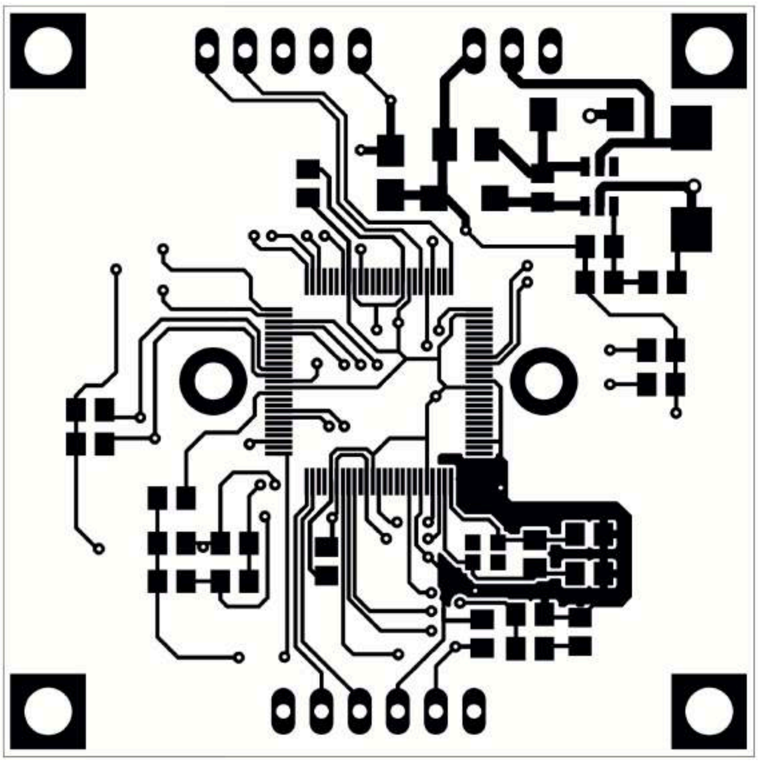


Figure 43 - Main board - Bottom layer (copper)

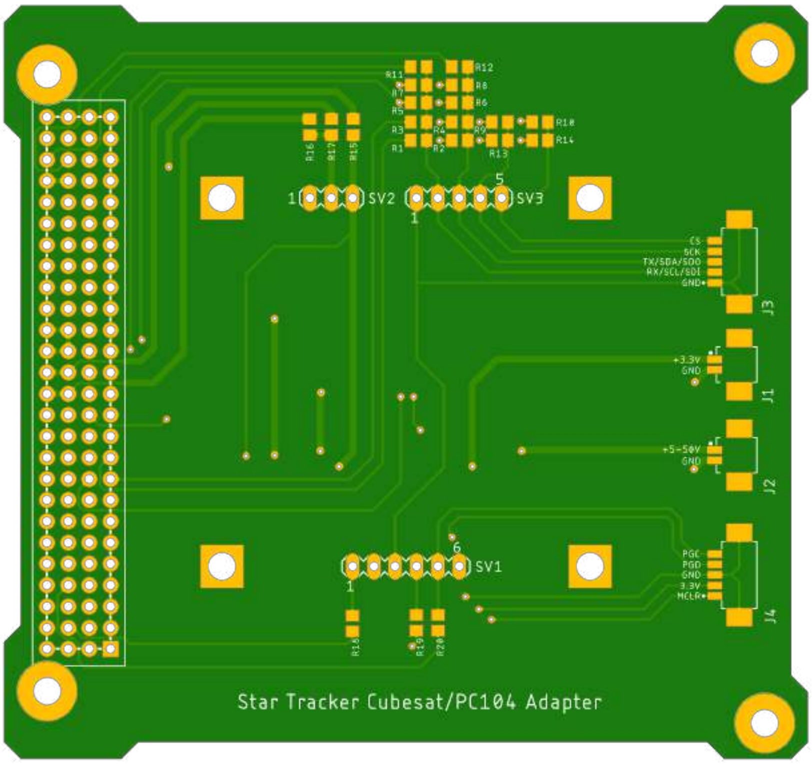


Figure 44 - Adapter board - Top layer (render)

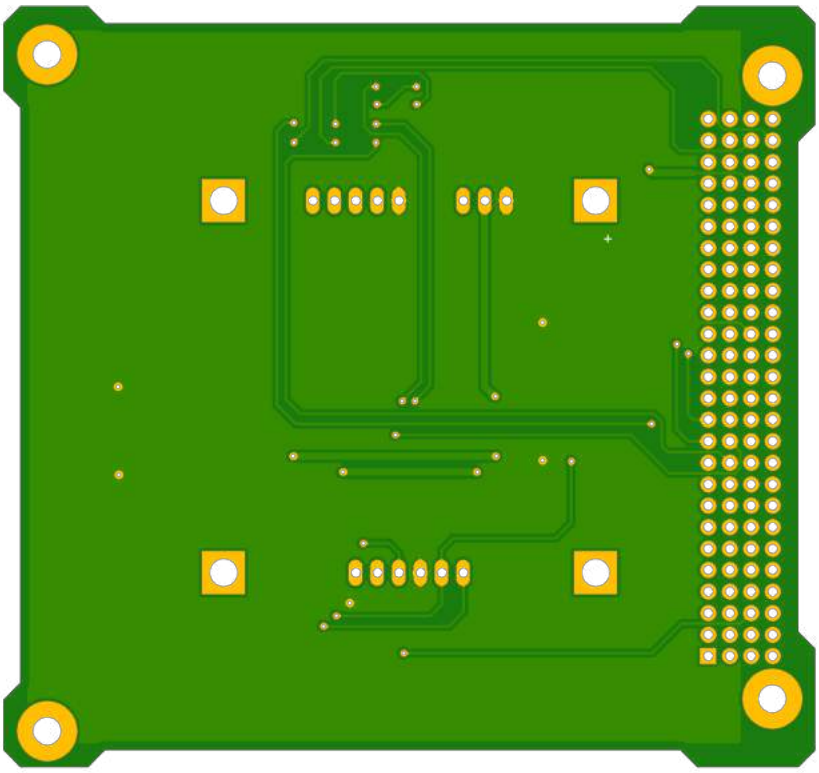


Figure 45 - Adapter board - Bottom layer (render)

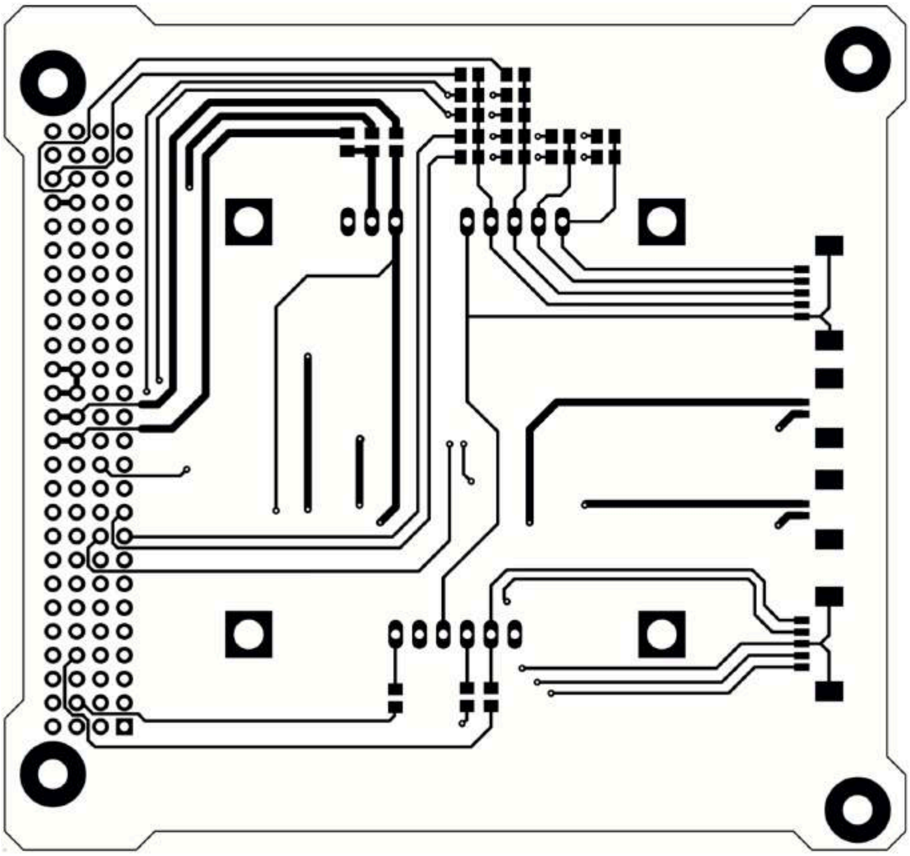


Figure 46 - Adapter board - Top layer (copper)

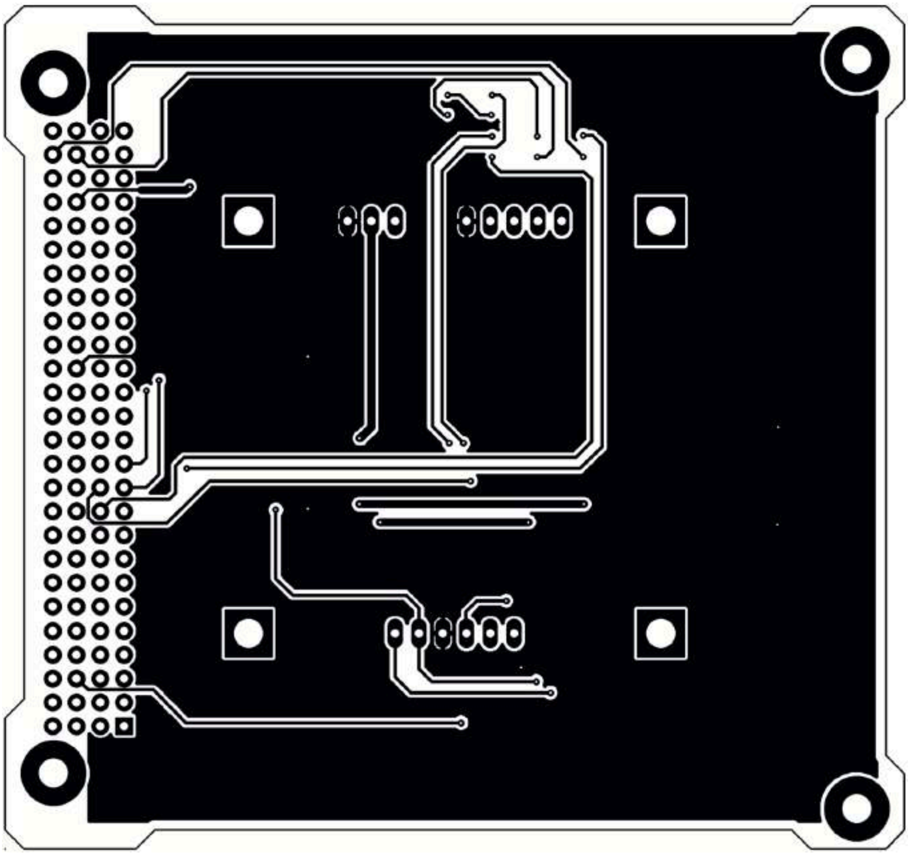


Figure 47 - Adapter board - Bottom layer (copper)

Ringraziamenti

A conclusione di questo lungo percorso di studi, affiancato alle esperienze lavorative che hanno accompagnato gli ultimi anni, vorrei spendere due parole per ringraziare tutti coloro che hanno reso possibile il raggiungimento di questo traguardo.
Vorrei anzitutto ringraziare i relatori, prof.ssa Federica Battisti e prof. Vincenzo Ferrara, che hanno supportato la realizzazione del progetto di tesi a cui ho lavorato.
Ringrazio in particolar modo Fabrizio Bernardini, che oltre ad essere stato il riferimento principale di questo progetto, è stato negli ultimi anni un'importante fonte di ispirazione e di conoscenze tecniche che difficilmente si trovano sui libri. Insieme a lui, ringrazio l'intera associazione BIS-Italia (sezione italiana della British Interplanetary Society), che mi ha dato l'opportunità di prendere parte a numerose attività interessanti e di conoscere persone straordinarie.
Ringrazio il personale della biblioteca scientifico-tecnologica e i ragazzi che hanno condiviso con me l'esperienza della borsa di collaborazione, ma un grazie particolare va a Marisa, dispensatrice di libri, di forza e di sorrisi.
Ringrazio i miei genitori e la mia famiglia, che mi ha supportato durante questi anni di studio e di lavoro. In particolare mio fratello Fabio, che mi ha trasmesso la passione per l'elettronica da bambino.
Ringrazio tutti i miei amici, che mi hanno aiutato ad affrontare tutte le sfide con lo spirito giusto.
Infine ringrazio Francesca, che negli ultimi anni mi è stata accanto, mi ha dato una forza incredibile e mi ha spinto a non mollare.