OPTICAM triple-channel astronomical image acquisition control software and external triggering synchronization system

Angel Castro[a,b,*], Iván Zavala[a], Enrique Colorado[a], Joel Herrera[a], Noel Castro Segura[b], Raúl Michel[c], Diego Altamirano[b], Liliana Altamirano-Dévora[c,d], Gerardo Sierra[c], Juan Echevarría[e], Javier Hernández-Landa[a]

[a]*Observatorio Astronómico Nacional, Universidad Nacional Autónoma de Mexico, Carr. Ensenada-Tijuana 3918, Ensenada, 22860, México*
[b]*School of Physics and Astronomy, University of Southampton, Southampton, SO17 1BJ, UK*
[c]*Instituto de Astronomía, Universidad Nacional Autónoma de México, Carretera Ensenada-Tijuana 3918, Ensenada, 22860, México*
[d]*Facultad de Ingeniería, Arquitectura y Diseño (FIAD), Universidad Autónoma de Baja California, Carretera Ensenada-Tijuana Km 103, 22860, México*
[e]*Instituto de Astronomía, Universidad Nacional Autónoma de México, Av. Universidad No. 3000, Ciudad de México, 04510, México*

Abstract

We present the development of software to control the multichannel image acquisition process and the design of an external timing module (ETM) for the OPTICAM system [1]. The ETM is used to generate high-speed synchronized pulses for OPTICAM's three different sCMOS cameras and is based on an ESP32-S2 micro-controller. The graphical user interface (GUI) of this instrument is also described, which allows users to enter the configuration parameters of the observations and execute actions such as starting and stopping image acquisition sequences or initializing cameras. 3 mini PCs are in charge of reading each one of the cameras, respectively. The software and hardware tools presented here have been successfully implemented at the Observatorio Astronómico Nacional (OAN-SPM), Mexico. By analyzing the recorded image sequences, it is possible to perform high-precision multi-channel photometric measurements of astrophysical objects that exhibit high temporal variability, revealing information about periodicities, inter-band delays, and quasi-periodic oscillations.

Keywords: Techniques: photometric, stars: imaging, external triggering, image acquisition software, graphical user interface, telescopes

1. Introduction

Time-domain astronomy is a rapidly expanding area in modern astrophysics. Recently developed high-time resolution instruments such as ULTRACAM [2] and HiPERCAM [3] have provided new unique insights into physical phenomena that occur on time scales that were previously unexplored. OPTICAM [1] is a new triple-channel optical system for high-time resolution photometric observations designed for the 2.1 m telescope of the San Pedro Mártir National Astronomical Observatory (OAN-SPM). OPTICAM is equipped with a set of $u'g'r'i'z'$ SDSS filters. The OPTICAM system (Fig. 1) is designed to perform strictly simultaneous observations in its three optical channels of astrophysical phenomena whose variations occur in sub-second time scales. This new instrumental system allows obtaining high-quality photometric measurements from relatively bright sources. Some examples of rapidly changing astrophysical objects include X-ray binaries, pulsating white dwarfs, compact accreting objects, exoplanets, and eclipsing binaries among others.

As recently suggested in [4], technological advances with high-frame-rate sCMOS and EMCCD cameras have extended time-domain astronomy to the sub-second regime, where the use of simultaneous multi-wavelength, high-cadence instruments such as OPTICAM attached to medium and large-sized telescopes may provide the crucial observational requirements for numerous astronomy studies. High-speed photometric ob-

*Corresponding author.
Email address: acastro@astro.unam.mx (Angel Castro)

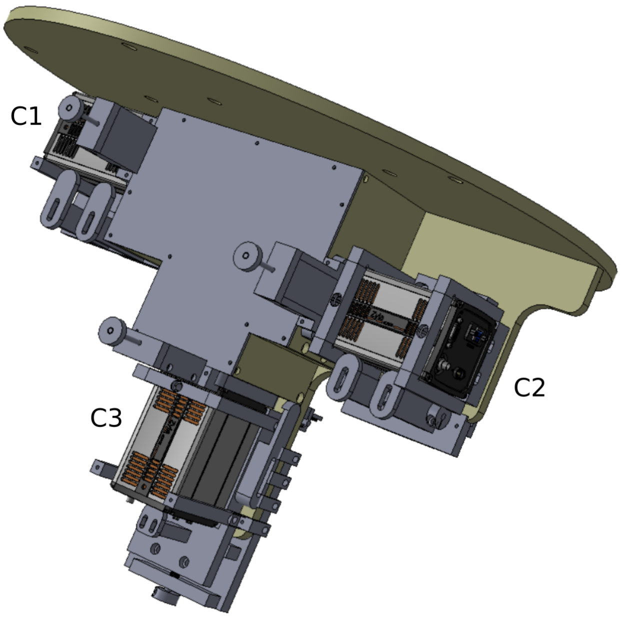

Figure 1: Three-dimensional graphic representation generated with the SolidWorks assisted design program of the OPTICAM instrument. From this perspective, it is possible to observe the 3 Andor Zyla 4.2-Plus USB 3.0 cameras used in the instrument (C1, C2 and C3, respectively).

servations such as those made with OPTICAM will allow precise measurements of orbital periods, spin periods and inter-band delays, as well as the determination of various scales of temporal variability, which could lead to the determination of periodic and quasi-periodic oscillations in some of the aforementioned astrophysical objects.

Data acquisition (DAQ) systems are utilized to measure and store data related to various physical variables [e.g. 5, 6, 7, 8], employing a specific sampling rate determined by the nature of the phenomenon under investigation. Some DAQ systems not only passively perform measurements but also actively engage in control activities as a response to the monitored physical parameters. An example of such an active response is the Telescope Control Unit of the ARIEL telescope [9].

Graphical User Interfaces (GUIs) enable users to interact with instrument systems in a visual and practical manner. Certain interfaces are simple, allowing users to input heterogeneous values that will feed or provide configuration information or functional parameters to systems [e.g. 10], or display images and data sets [e.g. 11, 12, 13]. More complex interfaces facilitate intricate tasks during the post-processing stage of images. For instance, [14] presents a 3-D interactive visual analytics tool for HI data, while `SHAPE` [15] offers a flexible, interactive 3D morpho-kinematical modelling application for astrophysics. The OPTICAM GUI is a simple interface that allows users to enter the configuration parameters of the observations made.

To guarantee the high-time precision of the acquired image series, it is necessary to design hardware and software for the synchronization and triggering of multichannel systems [e.g. 16, 17], that involve the use of micro-controllers, embedded systems, Field Programmable Gate Arrays (FPGAs) and various application programs with Linux drivers accompanied by their respective user interfaces. The timing module presented in this work (Sec. 5), has been developed to provide a precise time reference through the generation of external triggering pulses allowing the synchronization of the 3 Andor Zyla 4.2-Plus USB 3.0 cameras[1] of the OPTICAM system.

This paper presents an overview of the data acquisition system employed by OPTICAM. The system's primary function involves obtaining data to conduct precise photometric measurements, primarily focusing on point-like astrophysical sources. A dedicated graphical user interface (GUI) was explicitly designed to collect the users' observation parameters. Additionally, an external synchronization card was developed and built to ensure the accuracy of the simultaneous image acquisition process. Thorough testing of all systems was conducted on the OAN-SPM 2.1m telescope. In Section 3 we describe the fundamental characteristics of the image acquisition control software of OPTICAM. Section 4 describes the developed GUI. Section 5 offers a concise summary of the hardware and software components of the external timing module (ETM). The functional tests carried out on the coordinated set of OPTICAM subsystems are described in Section 6. Section 7 summarizes the work done and draws overall conclusions. Finally, future work projects are addressed in Section 8.

[1] `https://andor.oxinst.com/products/scmos-camera-series/zyla-4-2-scmos`

2. OPTICAM's Data Acquisition System

The overall operation of the OPTICAM data acquisition system is depicted in Fig. 2, illustrating its general scheme. In this section, a high-level overview of the system is presented. The system is based on three mini-PCs to control the image acquisition process of each Andor Zyla 4.2-Plus USB 3.0 camera and, in turn, a main PC is used to command the mini-PCs, as well as to read the data from the ETM and the GUI. Each mini-PC runs the control software (Section 3) responsible for capturing images produced by the camera, these are situated in a specific channel within the OPTICAM system. The GUI (described in Section 4), facilitates the transmission of instructions and parameter settings to the main PC acting as a remote server.

Three mini-PCs are employed, each one being responsible for acquiring and storing images from an individual sCMOS camera. These mini-PCs receive instructions from the main PC, which also functions as a remote server for command transmission and processing. In the current software version, users can request images with exposure times ranging from 0.01 to 30 seconds. Where the exposure time is defined by the rolling shutter external exposure triggering (non-overlap mode) timing parameters of the Andor Zyla 4.2-Plus USB 3.0 camera (see the Zyla sCMOS Hardware Guide for further details).

When an acquisition sequence is initiated, the first step involves retrieving accurate time information using a GPS antenna. Subsequently, the ETM (detailed in Section 5) synchronizes the cameras with high precision. The captured images are then stored on the solid-state drives of each mini-PC. The bidirectional communication between the mini-PCs and the remote server is provided using a TP-LINK 8 port TL-SG108 Metal Gigabit Switch. The execution of the multi-stage image acquisition process can be suspended by the user from the GUI using the respective cancel button. In the event of an abort, the program will briefly suspend the acceptance of new instructions to free the program memory in use and suspend the various actions being performed by the mini-PCs and peripherals. After three seconds, the GUI can be used to send new observation setups.

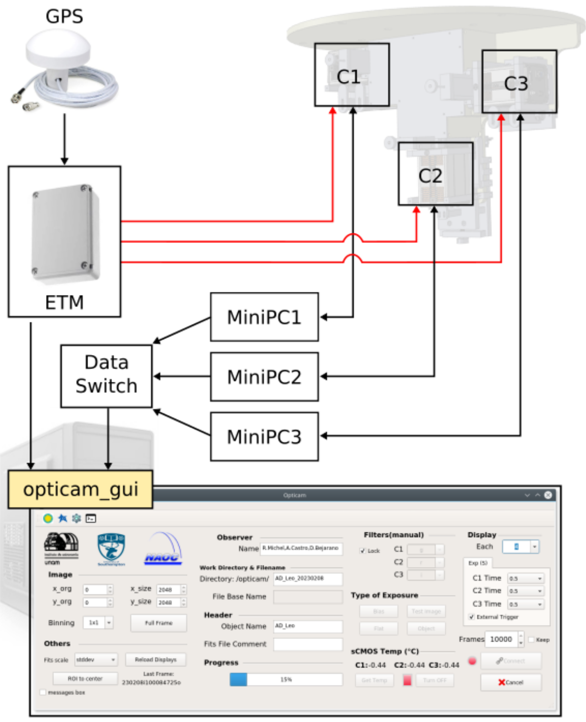

Figure 2: Block diagram of OPTICAM's DAQ system. A main PC, acting as a local host, is responsible for managing the GUI in which observation parameters are fed by the user. The configuration parameters are then sent to the MUFFIN program running simultaneously on 3 mini-PCs. Each mini-PC is responsible for the acquisition of images from its respective camera. The synchronized triggering process for each camera is controlled by an external timing card designed and built in the context of this work. A commercial GPS antenna is used to get absolute date and time references.

3. Image acquisition control software

The multi-channel image acquisition control software used by the OPTICAM system is called MUFFIN. Developed for the Linux operating system, this software is implemented in C++ and takes advantage of its object-oriented architecture. MUFFIN runs independently on each of OPTICAM's mini-PCs. In this way, each Andor Zyla 4.2-Plus USB 3.0 camera is controlled by an instance of the same class, requiring only the camera's serial number to search and initialize it. MUFFIN operates as a server, receiving commands via the TCP/IP protocol on port 9710 through the network. In this way, the control software is capable of receiving instructions via an Ethernet port and executing the required image acquisition tasks based on the observation parameters specified by the user through the GUI. Each command is decoded and processed in an independent thread, enabling simultaneous execution of multiple tasks.

The simultaneous control of the 3 Andor Zyla 4.2-Plus USB 3.0 cameras, MUFFIN, makes use of the basic control tools provided in the core Software Development Kit (SDK3) distributed and licensed by Oxford Instruments (Oxford, UK) at the time of purchase of the Andor Zyla 4.2-Plus USB 3.0 cameras. Particularly, our code utilizes the ATCORE library distributed by Oxford Instruments.

SDK3 is distributed with some examples of editable programs to execute basic tasks and provide training and learning elements to the programmer. After analyzing in detail the operation of these examples and studying in depth the user guide and hardware manual of the cameras Andor Zyla 4.2-Plus USB 3.0 (which explains in great detail the operation and the various configuration parameters of the cameras), we proceeded to implement specific tasks required for OPTICAM, which led to the development of a new programming code for the simultaneous control of the aforementioned cameras.

As an example, the images created in the sample code were stored in the BMP format, which is not a commonly used format in Astronomy, where the FITS format is often used. For this, it was necessary to additionally install the packages CCfits and CFITSIO to handle FITS files using C++ code. OAN-SPM's technical staff provided us with examples of CCD camera controllers currently working at OAN-SPM's facilities. An exhaustive study of these codes was carried out for some months, in parallel with the detailed analysis of the codes included in the ATCORE driver of SDK3. The camera's SDK and Andor Zyla 4.2-Plus USB 3.0 hardware user manuals proved to be particularly valuable tools for accomplishing the current software development.

MUFFIN also implements a set of multiple routines to manage the image acquisition process using 3 different cameras, send instructions and commands, and trigger error or warning messages through the remote server. The general operation of MUFFIN is outlined in the flow chart shown in Figure 4. To comply with the Flexible Image Transport System (FITS) standards, MUFFIN utilizes the CCfits and CFITSIO official libraries provided by HEASARC[2]. These libraries are compiled on the PC hosting the control software, allowing MUFFIN to use high-level routines for reading, writing, and manipulating FITS images, as well as handling the associated metadata (a.k.a. headers) for optimal storage and identification. MUFFIN also includes additional functionalities, such as providing feedback on telescope status and integrating data from a commercial meteorological station. The software can be used with other sCMOS cameras from Oxford Instruments. Each mini-PC is responsible for controlling the image acquisition of the connected sCMOS camera through the image acquisition software. However, to ensure proper coordination, the three mini-PCs communicate bidirectionally with the remote server running the GUI, via the TP-LINK 8 port TL-SG108 Metal Gigabit Switch. The GUI is accessed through a remote server from the telescope observation room, from where the user sends instructions that are received through the main PC's Ethernet port.

Once the mini-PCs validate the frame acquisition and saving instruction, a coordinated external pulse will trigger the acquisition sequence simultaneously in all cameras, ensuring a controlled and coordinated acquisition process. The details of this process are as follows: Prior to initiating the acquisition se-

[2] https://heasarc.gsfc.nasa.gov/fitsio/fitsio.html

Figure 3: Andor Zyla 4.2-Plus USB 3.0 sCMOS camera. The image acquisition control program MUFFIN, the GUI, and the ETM described in this document were developed to work with this camera.

quence, the ETM waits to receive an updated time stamp from the GPS network, which is refreshed at a rate of 1/10 Hz. Immediately after receiving the time reference update, the ETM initiates the pulse generation and triggers the image acquisition process simultaneously across all three cameras within the system.

```
void IMAGE::saveToFits(unsigned char * buf, char* name, AT_64 timestamp_64, bool first)
{
    int status = 0;
    string str;
    string fits_inverted;
    long naxes[2] = {i64_aoiWidth, i64_aoiHeight};
    double time_miliseconds;
    time_miliseconds = (double(1.0 / timestampFrequency) * timestamp_64) * 1000;
    if (first){
        timestamp_offset = time_miliseconds;
        time_miliseconds = 0;
    }
    else
        time_miliseconds = time_miliseconds - timestamp_offset;
    timeStamp->addMiliSeconds(time_miliseconds);
    remove(name);
    fits_create_file(&fptr, name, &status);
    fits_create_img(fptr, USHORT_IMG, 2, naxes, &status);
    fits_write_key(fptr, TDOUBLE, (char *) FITS_KEYS[OP_EXPOSURE], (void *) &m_dExposureTime,
        (char *) "Single frame exposure time (s)", &status);
    fits_write_key(fptr, TDOUBLE, (char *) FITS_KEYS[OP_GAIN], (void *) &m_dEGain, (char *) "Electronic gain (e-/ADU)         ", &status);
    fits_write_key(fptr, TINT, (char *) FITS_KEYS[OP_EXPCOAD], (void *) &m_nNumberExposuresCoAdded, (char *) "Number of exposures co-added", &status);
    ....
    ....
    ....
    fits_write_key(fptr, TSTRING, (char *) FITS_KEYS[OP_HISTORY], (void *) m_cHistory.c_str(), (char *) "Image history modification log", &status);
    fits_write_img(fptr, TUSHORT, 1, i64_aoiWidth * i64_aoiHeight, buf, &status);
    fits_close_file(fptr, &status);
    m_cLast_Image = m_cImDefaultPath + "last_" + m_cCameraID + ".fits";
    str = "ln -sf " + string(name) + " " + m_cLast_Image;
    system (str.c_str());
}
```

Listing 1: Section of the OPTICAM code responsible for saving images in FITS format on hard drives. This code configures the observation parameters registered in the headers of the saved images.

The control software developed, in addition to implementing computing solutions for specific problems associated with OPTICAM, is largely shareable with other operating subsystems in the OAN-SPM. This is useful for efficiently communicating with the multiple instruments, sensors and actuators of the 2.1 m telescope. Some of this information (e.g. telescope position, local temperature and wind, air mass, local time and date) is included in the headers of the recorded images.

4. Graphical user interface (GUI)

The user can enter the configuration parameters of an exposure sequence through a Python 3.1/Qt5-based GUI specifically designed for OPTICAM. The GUI software, along with the image acquisition control software described in Section 3, was

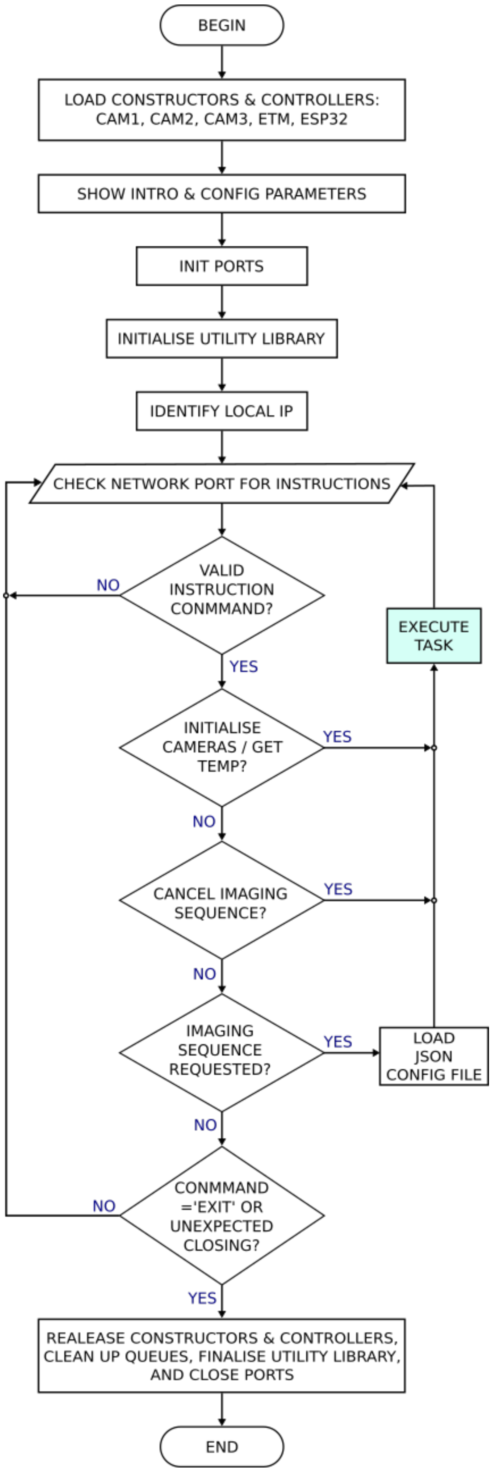

Figure 4: MUFFIN main program software flowchart. In this program, the main libraries are loaded, and the various subprograms responsible for performing the different tasks such as image formatting, setting up of headers, camera synchronization or peripherals control are included. Startup parameters and software configuration are also defined at this step. Commands are received through the Ethernet network port with the instructions sent by the user from the GUI.

developed and debugged using the Visual Studio Code source-code editor. The described GUI (Fig. 5) collects the entered information and transmits these parameters to the OPTICAM main PC through JSON-encoded files[3] sent over the local network. A new JSON file is sent each time a new acquisition sequence is started. The image acquisition control program on the main PC decodes the received JSON files and generates the specific execution commands to the sCMOS cameras. Once the FITS images are generated, they will also record most of the exposure parameters in their respective headers.

Information that can be entered into the GUI includes binning, origin coordinates, pixel array dimensions, exposure times for each camera, number of frames in the particular sequence, and type of image acquired (e.g. test image, object, flat). Additionally, supplementary information is included, such as the names of the observers, filters used during the sequence, the name of the astrophysical object under study, the image data storage directory and general comments. The GUI also features buttons to initiate or cancel the acquisition process for a given image sequence, as well as a progress bar indicating the percentage of completion during the acquisition. A flow chart illustrating the functioning of the GUI is presented in Fig. 6. Multi-channel images acquired with the OPTICAM DAQ can be simultaneously displayed in an SAO-DS9 viewer window (see Fig. 7) based on a user-defined refresh rate specified in the GUI prior to initiating an image sequence request.

```
from c_cliente import *
import c_telescopio_2m
class OPTICAM(CLIENTE):
    def __init__(self,builder):
      print ("OPTICAM Class Ready")
      CLIENTE.__init__(self)
      self.builder=builder
      self.tipo = "Andor Zyla"
      self.label = "Zyla-4.2P-USB3"
      self.label2 = "sCMOS "
      self.xsize_total = 2048
      self.ysize_total = 2048
      ...
```

[3]https://json-schema.org/specification

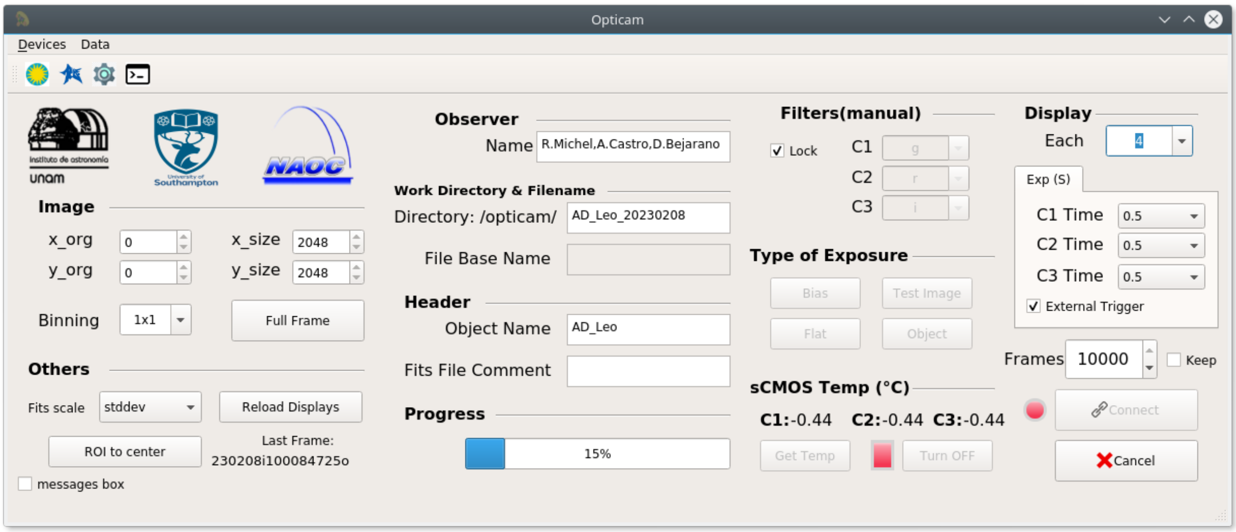

Figure 5: OPTICAM graphical user interface. Through this GUI, the observer can execute the initialization procedure of the cameras and establish a direct connection with them in an automated way by pressing the *Connect* button. Additionally, the user can enter basic field observation parameters and interact with the acquisition control program. For example, the user also defines the exposure time of the cameras, the number of desired frames, the physical dimensions (in pixels) of the field image, the binning factor, the filters in use, the directory for saving data specified on disk, and enters the name of the observers and the astronomy target. In this figure, the main operational sections of the OPTICAM GUI are split into several grids: 1) Regions of interest and grouping, 2) Headers, directory and additional information, 3) Filters, 4) Image acquisition parameters, 5) Others, 6) Progress Bar, 7) Exposure Type, 8) sCMOS Temperature, 9) Connection with cameras and sequence cancellation, 10) SAO-DS9 display update rate and exposure times

, and 11) Message window. The particular functions of these sections are described throughout Section 4.

```
    ...
    ....
    self.INIT_MSG="OPTICAM ANDOR Zyla"
    self.INIT_COMMAND="init"
    try:
       self.inicializa()
    except:
       print ("Initialisation error",self.
  INIT_MSG)
    self.Tel = c_telescopio_2m.TELESCOPIO2M(
  builder.mensajes, builder.update_pinta_info)
  def __del__(self):
    print ("Destructor of OPTICAM")
    self.manda("close ")
    self.manda("exit ")
    self.manda("EXIT ")
  def get_temp(self):
    # print "Read sCMOS temperature"
    t, s = self.manda("READ_TEMP ")
    if not s:
        print ("bad")
        print (t)
        self.builder.mensajes(t, "Log", "red")
        return -1
    t = t.split()
    self.temp = float(t[1])
    self.builder.mensajes("sCMOS Temp=" + str(
  self.temp))
```

Listing 2: Code segment of the muffin program developed in the C++ programming language to control the process of multi-channel image acquisition and storage of data on disk. The small code segment shown here refers to the part in which general instructions are received from the GUI through the remote server.

5. External Timing Module (ETM)

To guarantee precise synchronization of the OPTICAM cameras, an ESP-32-based timing module (see Figures 8 and 9) was developed and built at the OAN-SPM. This device makes use of the transistor-transistor logic (TTL) external trigger I/O ports of the Andor Zyla 4.2-Plus USB 3.0 sCMOS cameras to trigger an exposure or sequence of exposures. The user can define the exposure time in the GUI (see Section 4). The software is

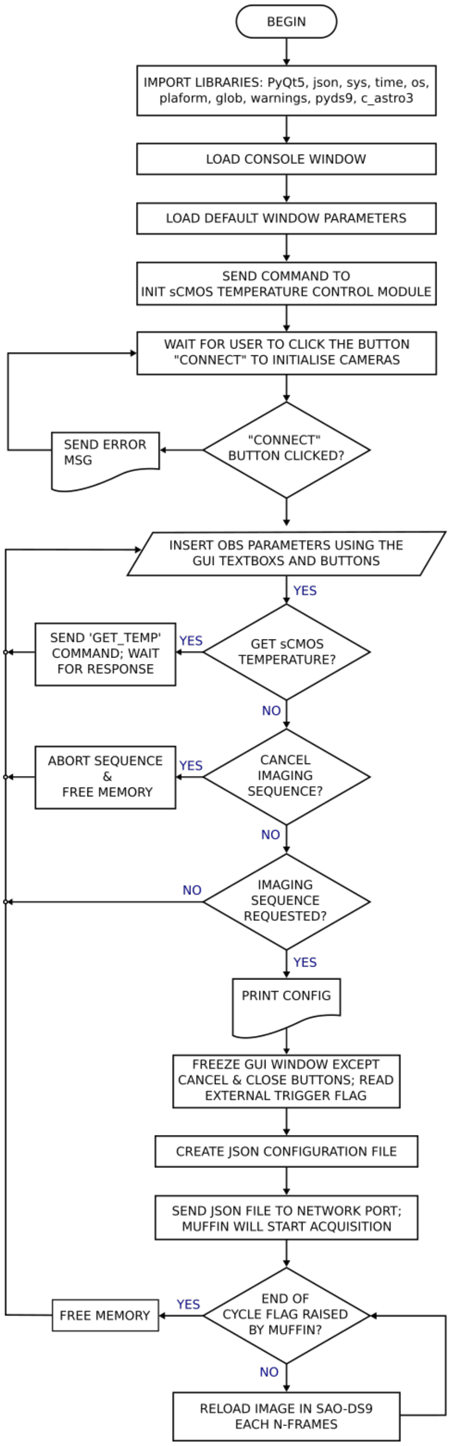

Figure 6: GUI flow chart. In the GUI, the user can enter the basic configuration values of the observations to be made, such as the exposure times for each camera, binning values, observer data, object name and image update frequency in the SAO-DS9 image viewer, as well as specify the directory for data storage and the number of frames to acquire in a specific sequence. Once the exposure request button is clicked, the user configuration data, optical filters in use, and external timing flag are saved in a JSON format file and sent to the remote server. MUFFIN will decode this file and execute the corresponding tasks.

responsible for performing some simple calculations and sending the setup instructions to the timing module. The ETM card (Fig. 8 has 3 output ports. Triggering pulses can be configured completely independently on each output port.

The ESP-32 is a development board equipped with the Tensolica Xtensa LX7 32-bit micro-controller, 120 kB ROM, and 320 kB SRAM. It offers connectivity to numerous peripherals through its 43 programmable GPIOs, supporting full-speed USB OTG, LCD, I2C, I2S, UART, SPI, ADC, and DAC interfaces. The operating temperature range of the ESP32-S2 is -40 to +105 degrees Celsius, and it provides 802.11n WiFi connectivity at speeds of up to 150 Mbps. The board features a 40 MHz crystal oscillator, and its integrated memory components consist of 4 MB SPI flash and 2 MB PSRAM. In the described ETM, an ESP-32 serves as the CPU, generating control pulses received by the 3 Andor Zyla 4.2-Plus USB 3.0 cameras as external activation triggers, thereby synchronizing the image acquisition process of OPTICAM.

The ETM was locally designed by the Instrumentation Department of the OAN-SPM, including the circuit and its corresponding PCB using Autodesk EAGLE 9.6.2[4]. The manufactured PCB (see Fig. 9) underwent testing in the instrumentation laboratory and subsequently in the OPTICAM instrument during several observing campaigns.

The external timing module can generate a software-adjustable frequency TTL pulse train, depending on the user-specified exposure time in the OPTICAM GUI. The voltage output signal from the ETM is transmitted through coaxial cables connected to mini-BNC terminals on the timing module side and a BNC to DB15 adapter on the Andor Zyla 4.2-Plus USB 3.0 camera. The ETM provides three outputs, each capable of generating independent signals across its three channels.

The software embedded within the ESP-32's core has been specifically developed for OPTICAM operations and coded in the C++ programming language. Visual Studio Code[5] editor was utilized for editing and debugging the ETM code during the

[4]https://www.autodesk.com/products/eagle/overview
[5]https://code.visualstudio.com/

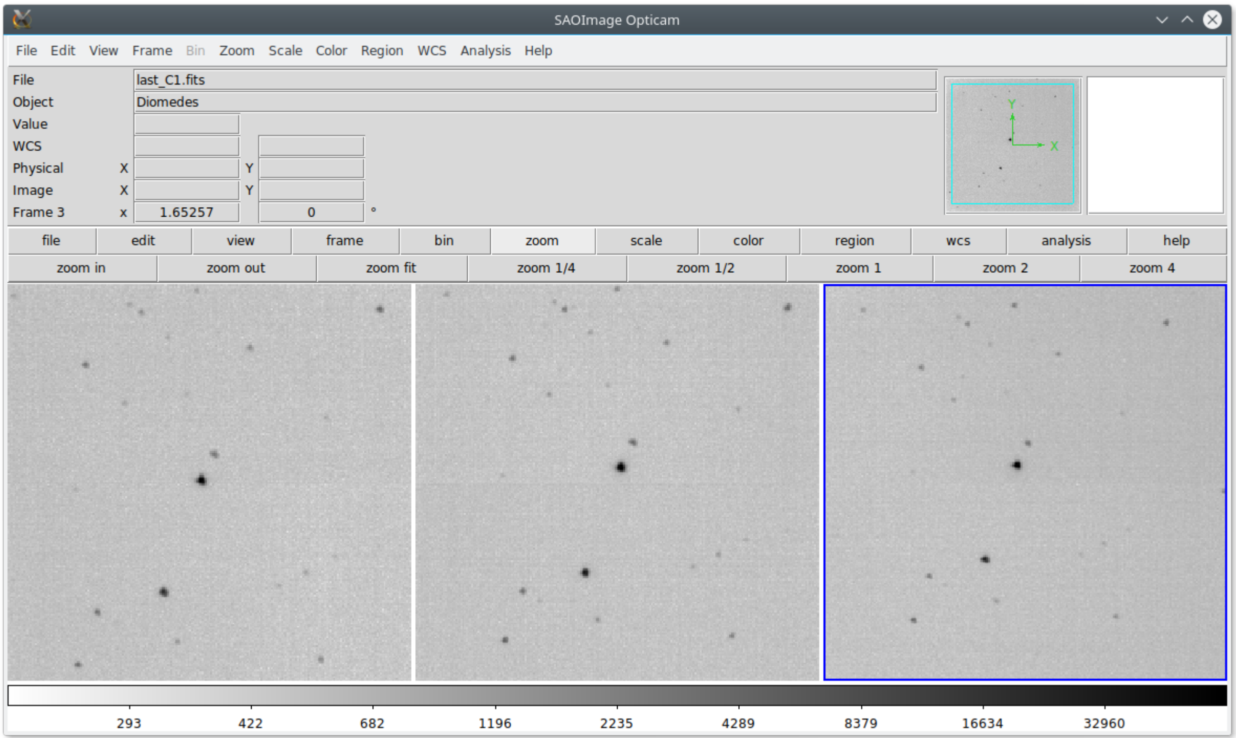

Figure 7: The images acquired by OPTICAM through the image acquisition control software are shown in the SAO-DS9 astronomical image viewer. The SAO-DS9 image viewer refreshing rate can be defined in the GUI. This figure shows, as an example, a star field that was observed at the OAN-SPM with OPTICAM (g', r', and i' filters, from left to right) on February 8, 2023.

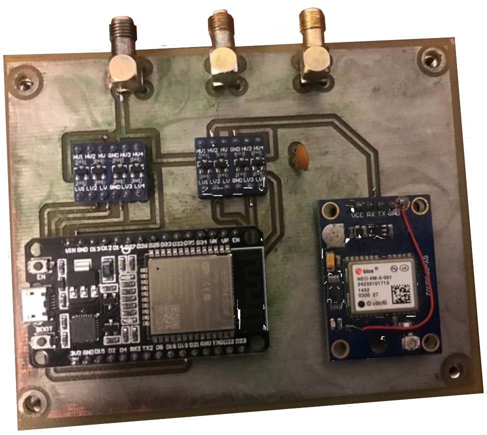

Figure 8: Photo of the printed card on which the various elements of the OPTICAM timing module have been mounted during the laboratory testing stage. This module can generate the same TTL output signal on all three channels or generate independent sync signals on each channel. This is defined by the user when specifying in the GUI the exposure times of the respective observation.

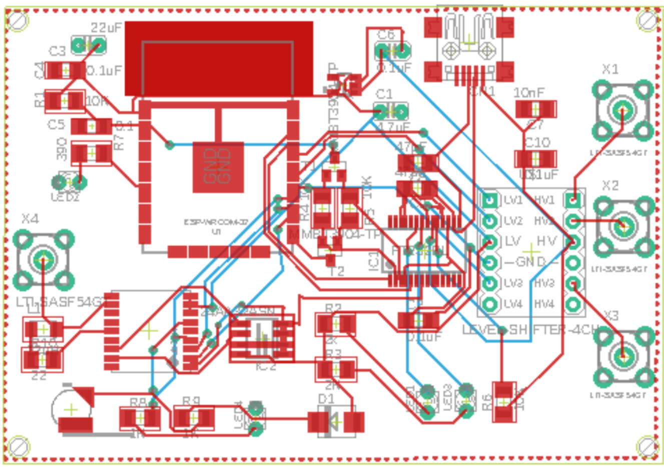

Figure 9: PCB layout of the ETM generated in Autodesk EAGLE 9.6.2. The tracks on the upper face of the circuit are shown in red, while the blue tracks correspond to the lower side of the PCB. In green, the pins of the through-hole mount components are shown. The red and green rectangles correspond to ground planes located on the upper and lower faces of the PCB, respectively.

development and testing stages. Once compiled, the software was downloaded to the ESP-32 board via the USB port.

Communication between the ETM and the OPTICAM's main PC occurs at a baud rate of 9600 with the common bus configuration 8-N-1. A dedicated USB port on the PC is exclusively used for this communication, providing the necessary power supply to the ETM. The ETM incorporates a commercial antenna (refer to Fig. 10) to obtain precise time and date values each time an image sequence is initiated from the GUI.

```
nt PulseController::Set_PulsePeriod(float
    exp_time)
{
    std::ostringstream streamObj;
    string com = "pc,";
    streamObj << com << exp_time;
    Send_Command(streamObj.str());
    return 0;
}
bool PulseController::Send_Command( string
    command )
{
    if ( isOpen() )
    {
        serial_port.Write(command);
        serial_port.DrainWriteBuffer();
        usleep(100);
        try{
          serial_port.Read(response,50,
    timeout_milliseconds) ;
        }
        catch (const ReadTimeout&){
          std::cerr << "Expired command." << std
    ::endl;
        }
        return true;
    }
    return false;
}
```

Listing 3: Fragment of the code of the routine used to define the value of the duration of the synchronization pulse sent to the OPTICAM cameras. Said value is entered as an argument from the main image acquisition control program based on the values entered by the user in the GUI.

The available set of instructions for the ETM is shown in Table 1. These instructions are transmitted from one of the USB ports on the main PC to the corresponding port on the ETM. The micro-controller inside the ETM reads and interprets these instructions, executing them accordingly. The ETM is connected to each Andor Zyla 4.2-Plus USB 3.0 camera via an isolated coaxial cable. This configuration enables the ETM to generate external TTL pulses according to the parameters entered by the user in the GUI through a cable that is connected by means of an SMA connector to the external trigger port of each camera. the ETM is capable of generating pulses with different periods and high pulse duration times for each of its three pulse output terminals.

Command	Description
`status`	Send ETM communication satus
`gpsstat`	Send GPS status
`reset`	Reset ETM
`pc`	Set pulse period
`np`	Set pulse number
`now`	Start pulse sequence
`abt`	Abort pulse sequence.
`gpstime`	Wait for GPS time reference.

Table 1: Set of instructions sent from the OPTICAM control program to the external timing module. These instructions are used to configure the pulse train used to synchronize the OPTICAM cameras.

The pulse sequence can be aborted at any time by sending the `abt` instruction to the ETM micro-controller, for example, to cancel an exposure sequence directly from the GUI. This command informs `MUFFIN` that the user wishes to abort a sequence and potentially reset the pulse settings based on the observation parameters entered in the GUI.

6. Functional tests

Prior to deployment on the telescope, extensive testing of all OPTICAM subsystems was performed under laboratory conditions. Once the technical team validated the functionality of the different capabilities of the instrument, it was mounted on a telescope, where various engineering tests were carried out aimed at testing these same capabilities by observing real astro-

Figure 10: GPS IP65 waterproof antenna w/7M (22.96ft) RG58 coaxial cable SMA and 28 dB gain used by the OPTICAM ETM. The antenna features a standard HYS Marine SMA male GPS antenna connector and ABS mounting base. This antenna is permanently installed on the outside of the OAN-SPM 2.1m telescope dome. It operates at 1575±5 MHz with a bandwidth of 10 MHz and requires a power supply of 3–5 VDC.

physical sources.

6.1. Laboratory study

The OPTICAM optomechanical system was built in late 2019 in the Instrumentation Department of the Astronomy Institute of UNAM. During the first laboratory tests, the OPTICAM system used a timing card supplied by a third-party supplier, as well as an evaluation version of the image acquisition control software developed throughout 2018, and a beta version of the GUI developed at the beginning of 2019. The first functional version of `MUFFIN` had code developed in C++ for reading the aforementioned card. Simple performance tests, which consisted of requesting an image sequence with a certain frame acquisition rate and then cancelling it to request an additional sequence with a different acquisition rate, showed us that this card could not meet the total OPTICAM requirements. After repeated attempts to get feedback from the card vendor, our team decided to develop an external timing card of our own. Due to the global pandemic in 2020, technical activities were suspended for a long time and then resumed in late 2021. A new external timing card based on a micro-controller and GPS was then designed (see Section 5).

OPTICAM remained thereafter in the UNAM optics laboratory, during which ETM integration tests were carried out and the necessary modifications were made for its joint operation with the MUFFIN software. In addition to these tests, cosmetic adjustments were made to the GUI as a result of the tests carried out.

After some tests carried out on a telescope under nominal operating conditions, it was found that during full-frame data acquisition processes at maximum data acquisition speed for said configuration, the system experienced apparent bottlenecks during data transfer. The instrument was sent back to the laboratory, and a 2TB PCIe Gen 4 NVMe Internal Solid State Drive was adapted to its PC and the RAM capacity was increased to 64 GB. To test the stability of the data acquisition process, image sequences were taken continuously in full-frame mode for eight continuous hours, the approximate duration of a nighttime observation. A virtual star was used to test the imaging capabilities of OPTICAM. This virtual star was generated with an 80–micrometer optical fibre and an LED powered by a function generator. In the optical bench, we assembled a relay system composed of a pair of achromatic doublets with unit magnification and a focal ratio of $f/7.5$. A commercial signal generator was used as a signal modulator. In this way, the artificial light source was capable of showing temporal variations, which could be adjusted for specific experiments.

Improvements were implemented based on feedback received from users and developers throughout the testing stages described. Currently, there are two versions of executable control: a) one that uses a single PC to control all the processes, and b) a modified one (the one we present here) capable of using and coordinating three different mini-PCs, which also allows for achieving higher frame acquisition rates, as well as stable data transfer and storage rates. Once the telescope testing period concludes, this latest modified version will remain the standard software for the instrument.

6.2. Field study

In October 2019, the first image was acquired with the OAN-SPM 2.1m telescope during an engineering test. At that time the system did not yet have the dichroic elements installed due to budget reasons. However, we were able to take a series of

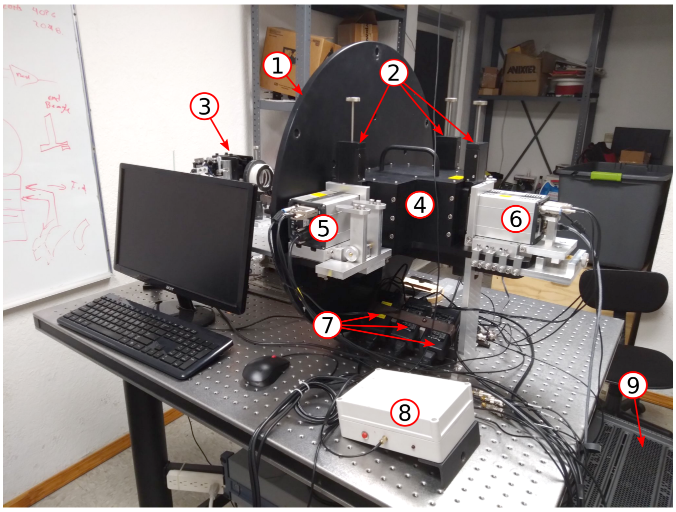

Figure 11: OPTICAM setup was tested in the optics laboratory of the Institute of Astronomy, UNAM. The figure shows: (1) a collimated source of artificial light, (2) filter exchangers, (3) a plate for holding the instrument to the telescope stage, (4) a box containing the optical system, (5) and (6) are the cameras C1 and C3, respectively, (7) power supply regulators, (8) ETM, and (9) the PC running the GUI. This setup allowed simultaneous adjustment of the optical fields as well as software testing.

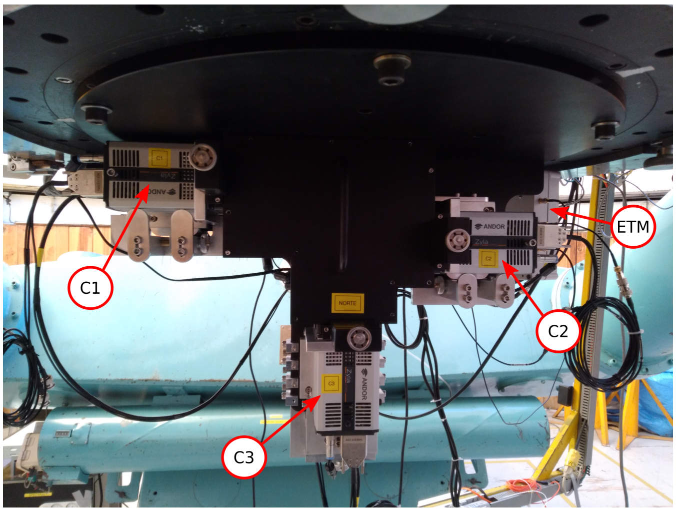

Figure 12: Image of the OPTICAM system mounted on the OAN-SPM 2.1m telescope during engineering tests.

data sequences using various data acquisition rates and an external timing card supplied by a third-party vendor. During these first tests using real telescope images, we were able to identify a problem with the timestamp recorded in the image headers, which did not have the resolution and precision necessary for exploring photometric variations on the sub-second scale. Hardware and software solutions were explored, and the instrument was shipped back to the lab.

OPTICAM was then installed on the OAN-SPM 2.1 m telescope on five occasions throughout 2022 and the first half of 2023. During those observing runs, engineering tests were carried out aimed at configuring the local network connection parameters, fine-tuning the optomechanics of the instrument, hardware and software performance testing, as well as first-light scientific imaging with this instrument.

As results of the field studies carried out, we present light curves of the objects GK Per (Fig. 13) and AD Leo (Fig. 14) acquired with OPTICAM in February and March 2023, respectively. The ETM, image acquisition control software and GUI described in this paper were used to achieve the results shown in these figures. Images were acquired in the three optical channels of OPTICAM and were then processed with the standard aperture photometry routines provided by the IRAF[6] package and some home-made auxiliary programs.

Figure 13 illustrates a light curve of the white dwarf nova GK Persei, gathered during two hours of continuous observation using the g, r, and i filters of the SDSS set. In this plot, the vertical axis is the differential magnitude (the measurement of the difference in brightness between the object of interest and a reference star). These observations were conducted on the OAN-SPM 2.1 m telescope, employing a uniform exposure time of 5 seconds across all three cameras and utilizing the external synchronization mode. The ETM developed in this work generated the external triggering pulses for each sCMOS camera. Figure 14 showcases a two-hour period during which u'-band observations of the M-type flaring star AD Leo were conducted, utilizing an exposure time of 0.2 seconds. Notably, significant small-scale variability activity is observed in the differential photometry performed during this period.

[6]IRAF is distributed by the National Optical Observatories, operated by the Association of Universities for Research in Astronomy, Inc., under a cooperative agreement with the National Science Foundation.

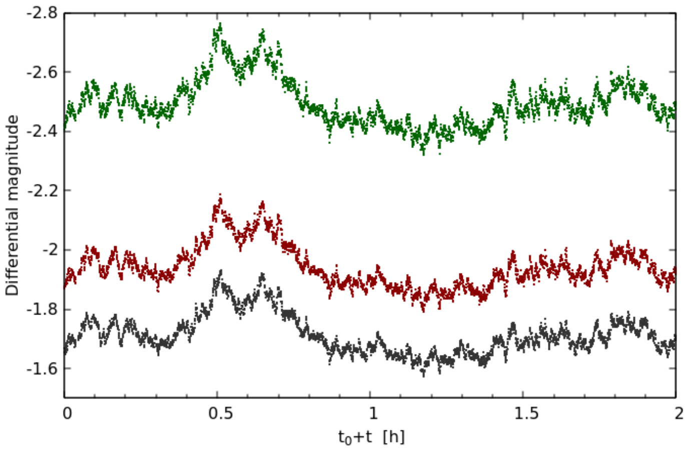

Figure 13: Light curve of GK Persei as observed by OPTICAM on February 06, 2023. The plot shows the simultaneous triple-channel observations (g', r', and i', shown in green, red, and grey, respectively). An average error of 0.004 mag was obtained in the relative magnitude measurements presented in this light curve corresponding to two hours of observation.

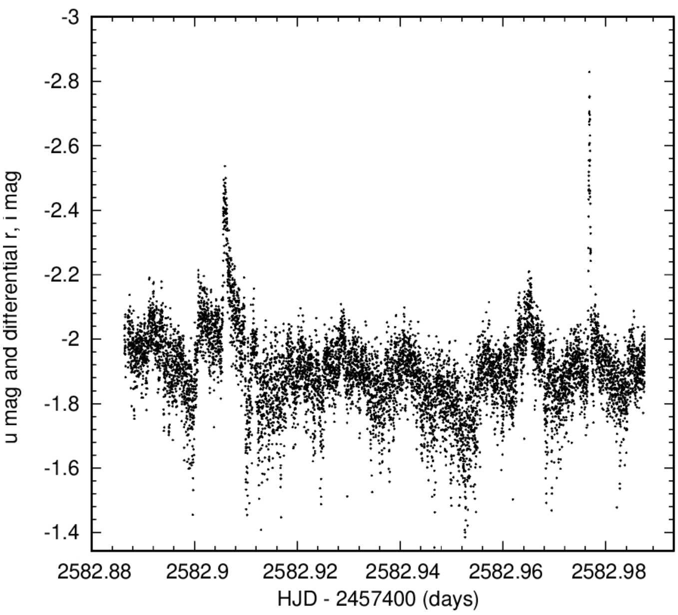

Figure 14: Raw light curve of the flaring star AD Leo as observed by OPTICAM on March 2023 (Sloot et al., in preparation). About two hours of continuous observation of AD Leo at a 5 frames-per-second rate in the u'–band are shown in this plot. Each point corresponds to a single photometric measurement of the astrophysical source compared to a non-changing reference star. Errors in the differential magnitude measurements are in the range of 0.002 mag for the present analysis.

7. Summary and Conclusions

In order to provide control tools and timing capabilities for the new OPTICAM system, image acquisition software, a GUI and an external synchronization module were developed. The software developed for OPTICAM has three fundamental objectives: 1) the development of control software for 3 Andor Zyla 4.2-Plus USB 3.0 cameras fully compatible with the instrumentation already installed in the OAN-SPM, 2) the design and construction of an accurate ETM, and 3) the development of a friendly user interface.

The purpose of the `MUFFIN` software is to allow the realization of precise photometric observations in multi-channel, with negligible dead times between frames, with exposures made in a truly simultaneous way in the three channels of the system. Initially, we used SDK3 to perform performance and functionality tests. SDK3 includes some sample programs written in the C++ programming language. These programs were used to read the first test images with the Andor Zyla 4.2-Plus USB 3.0 cameras that were later installed in OPTICAM. We proceeded to develop a new code according to the needs of our instrument. The multi-thread object-oriented software developed for OPTICAM (i.e. `MUFFIN`) allows us to simultaneously control the three Andor Zyla 4.2-Plus USB 3.0 cameras of the system while being compatible with the common set of instructions and commands used in the OAN-SPM.

The characteristics described above make OPTICAM an instrument of great potential among the scientific community since it will allow astronomical observations to be made of astrophysical objects showing fast variability. An added benefit of being able to perform them in three optical bands strictly at the same time (instead of acquiring images sequentially using a filter wheel).

A GUI was developed to allow the user to define configuration parameters, such as the size of the region of interest, binning, and exposure times, as well as variables related to the specific observed field such as the name of the object, observer name, and data storage directory. The optical filters placed on each arm of the instrument (see Fig. 1) of the system must also be specified in the GUI. These parameters are sent from the GUI via a JSON file over the Internet to the main PC, which acts as a local host, and in turn, is responsible for sending specific instructions to the mini PCs responsible for acquisition control and data transfer of each of the astronomical cameras. Figure 5 shows an image of the GUI window developed for OPTICAM.

Due to the capabilities of the OPTICAM system, it would be possible to perform unprecedented multi-channel, high-speed, photometric studies. This will allow measuring the delay times between bands and making simultaneous observations with other measurement instruments in the X-ray energy band such as NICER [18], AstroSAT [19], as well as other space- and ground-based observatories.

The software described in this paper may be made available to interested parties through direct communication with the authors. It should always be taken into account that the interested party must have at least one Andor sCMOS camera, duly licensed, at their own expense.

8. Future Work

The development of an automated pipeline for high-speed data reduction [e.g. 20, 21] is an immediate step to be taken by the OPTICAM team to efficiently analyze OPTICAM observations. This will represent a significant saving in data analysis time and avoid future problems of massive storage and transfer of data while allowing us to view the results of the observations in a short time, with important implications in terms of decision-making and feasibility of the scientific cases addressed with the instrument. The use of simulated wide-field images [22] could be useful for users in conjunction with a future observation simulator with OPTICAM. Tools like `Period04`[23] and `VARTOOLS` [24], and techniques like the one presented by [25] will be very useful for light curve processing and data analysis, including signal identification, filtering, light curve manipulation, period searching, modelling and simulation of light curves, as well as the potential identification of quasi-periodic oscillations.

Acknowledgments This paper is based upon observations carried out at the Observatorio Astronómico Nacional on the Sierra San Pedro Mártir (OAN-SPM), Baja California, México, which is operated by the Universidad Nacional Autónoma de México (UNAM). A.C. acknowledges support from the Royal Society Newton International Fellowship. D.A. and N.C.S. acknowledge support from the Science and Technology Facilities Council (STFC) grant ST/V001000/1. We thank Felipe Montalvo, Gustavo Melgoza, Hortensia Riesgo, and Francisco Guillén for their valuable support as telescope operators during the first light and commissioning of OPTICAM. Thanks are also due to Ilse Plauchu-Frayn, Tomás Verdugo, Tomás Calvario, José L. Ochoa, Luis Ortiz, Francisco Valenzuela, Edgar Cadena, and the rest of the technical staff for the support received.

References

[1] A. Castro, D. Altamirano, R. Michel, P. Gandhi, J. Hernández Santisteban, J. Echevarria, C. Tejada, C. Knigge, G. Sierra, E. Colorado, J. Hernández-Landa, D. Whiter, M. Middleton, B. García, G. Guisa, N. Castro Segura, Opticam: A triple-camera optical system designed to explore the fastest timescales in astronomy, Revista Mexicana de Astronomía y Astrofísica 55 (2019) 363–376. doi:10.22201/ia.01851101p.2019.55.02.20.

[2] V. S. Dhillon, T. R. Marsh, M. J. Stevenson, D. C. Atkinson, P. Kerry, P. T. Peacocke, A. J. A. Vick, S. M. Beard, D. J. Ives, D. W. Lunney, S. A. McLay, C. J. Tierney, J. Kelly, S. P. Littlefair, R. Nicholson, R. Pashley, E. T. Harlaftis, K. O'Brien, ULTRACAM: an ultrafast, triple-beam CCD camera for high-speed astrophysics, 378 (3) (2007) 825–840. arXiv:0704.2557, doi:10.1111/j.1365-2966.2007.11881.x.

[3] V. Dhillon, S. Dixon, T. Gamble, P. Kerry, S. Littlefair, S. Parsons, T. Marsh, N. Bezawada, M. Black, X. Gao, D. Henry, D. Lunney, C. Miller, M. Dubbeldam, T. Morris, J. Osborn, R. Wilson, J. Casares, T. Muñoz-Darias, E. Pallé, P. Rodriguez-Gil, T. Shahbaz, A. de Ugarte Postigo, First light with HiPERCAM on the GTC, in: C. J. Evans, L. Simard, H. Takami (Eds.), Ground-based and Airborne Instrumentation for Astronomy VII, Vol. 10702 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, 2018, p. 107020L. arXiv:1807.00557, doi:10.1117/12.2312041.

[4] J. A. Hitchcock, C. Helling, A. Scholz, G. Hodosan, M. Dominik, M. Hundertmark, U. G. Jørgensen, P. Longa-Peña, S. Sajadian, J. Skottfelt, C. Snodgrass, V. Bozza, M. J. Burgdorf, J. Campbell-White, R. Figuera Jaimes, Y. I. Fujii, L. K. Haikala, T. Henning, T. C. Hinse, S. Lowry, L. Mancini, S. Rahvar, M. Rabus, J. Southworth, C. von Es-

sen, Mindstep Collaboration, Large-scale changes of the cloud coverage in the Indi Ba and Bb system, 495 (4) (2020) 3881–3899. doi:10.1093/mnras/staa1344.

[5] M. Benghanem, Measurement of meteorological data based on wireless data acquisition system monitoring, Applied Energy 86 (12) (2009) 2651–2660. doi:https://doi.org/10.1016/j.apenergy.2009.03.026.
URL https://www.sciencedirect.com/science/article/pii/S0306261909001512

[6] A. Castro, J. F. Martínez-Osuna, R. Michel, M. Escoto-Rodríguez, S. H. Bullock, A. Cueva, E. López-Reyes, J. Reimer, M. Salazar, S. Villarreal, R. Vargas, A low-cost modular data-acquisition system for monitoring biometeorological variables, Computers and Electronics in Agriculture 141 (2017) 357–371. doi:https://doi.org/10.1016/j.compag.2017.08.010.
URL https://www.sciencedirect.com/science/article/pii/S0168169917300534

[7] C. Vargas-Salgado, J. Aguila-Leon, C. Chiñas-Palacios, E. Hurtado-Perez, Low-cost web-based supervisory control and data acquisition system for a microgrid testbed: A case study in design and implementation for academic and research applications, Heliyon 5 (9) (2019) e02474. doi:https://doi.org/10.1016/j.heliyon.2019.e02474.
URL https://www.sciencedirect.com/science/article/pii/S2405844019361341

[8] J. F. Martínez-Osuna, F. J. Ocampo-Torres, L. Gutiérrez-Loza, E. Valenzuela, A. Castro, R. Alcaraz, C. Rodríguez, L. R. Ulloa, Coastal buoy data acquisition and telemetry system for monitoring oceanographic and meteorological variables in the gulf of mexico, Measurement 183 (2021) 109841. doi:https://doi.org/10.1016/j.measurement.2021.109841.
URL https://www.sciencedirect.com/science/article/pii/S0263224121007880

[9] C. Sierra-Roig, M. Focardi, V. Da Deppo, G. Morgante, J. Colomé Ferrer, The telescope metrology control unit (tcu) on-board the ariel space mission, Measurement 122 (2018) 443–452. doi:https://doi.org/10.1016/j.measurement.2017.12.044.
URL https://www.sciencedirect.com/science/article/pii/S0263224117308151

[10] Retrocam: A versatile optical imager for synoptic studies, The Astronomical Journal 129 (2007) 2504. doi:10.1086/429594.

[11] P. Morse, A. Reading, C. Lueg, Animated analysis of geoscientific datasets: An interactive graphical application, Computers Geosciences 109 (08 2017). doi:10.1016/j.cageo.2017.07.006.

[12] J. Glass, R. Jain, R. Junghanns, J. Sallwey, T. Fichtner, C. Stefan, Web-based tool compilation of analytical equations for groundwater management applications, Environmental Modelling Software 108 (2018) 1–7. doi:https://doi.org/10.1016/j.envsoft.2018.07.008.
URL https://www.sciencedirect.com/science/article/pii/S1364815218301646

[13] A. Oliveira, A. Fortunato, J. Rogeiro, J. Teixeira, A. Azevedo, L. Lavaud, X. Bertin, J. Gomes, M. David, J. Pina, M. Rodrigues, P. Lopes, Opencoasts: An open-access service for the automatic generation of coastal forecast systems, Environmental Modelling Software 124 (2020) 104585. doi:https://doi.org/10.1016/j.envsoft.2019.104585.
URL https://www.sciencedirect.com/science/article/pii/S1364815219304967

[14] D. Punzo, J. van der Hulst, J. Roerdink, J. Fillion-Robin, L. Yu, Slicerastro: A 3-d interactive visual analytics tool for hi data, Astronomy and Computing 19 (2017) 45–59. doi:https://doi.org/10.1016/j.ascom.2017.03.004.
URL https://www.sciencedirect.com/science/article/pii/S2213133717300173

[15] W. Steffen, N. Koning, S. Wenger, C. Morisset, M. Magnor, Shape: A 3d modeling tool for astrophysics, IEEE transactions on visualization and computer graphics 17 (04 2010). doi:10.1109/TVCG.2010.62.

[16] Z. qing Liu, S. wu Pan, Y. gang Zhang, A design and implementation of and triggering system for lxi instruments, Measurement 46 (8) (2013) 2753–2764. doi:https://doi.org/10.1016/j.measurement.2013.04.034.
URL https://www.sciencedirect.com/science/article/pii/S0263224113001450

[17] G. C. George, A. A. Prince, J. Buch, S. K. Pathak, Design of configurable multi-mode trigger unit, Measurement 139 (2019) 482–489. doi:https://doi.org/10.1016/j.measurement.2019.03.028.
URL https://www.sciencedirect.com/science/article/pii/S0263224119302374

[18] K. C. Gendreau, Z. Arzoumanian, P. W. Adkins, C. L. Albert, J. F. Anders, A. T. Aylward, C. L. Baker, E. R. Balsamo, W. A. Bamford, S. S. Benegalrao, D. L. Berry, S. Bhalwani, J. K. Black, C. Blaurock, G. M. Bronke, G. L. Brown, J. G. Budinoff, J. D. Cantwell, T. Cazeau, P. T. Chen, T. G. Clement, A. T. Colangelo, J. S. Coleman, J. D. Coopersmith, W. E. Dehaven, J. P. Doty, M. D. Egan, T. Enoto, T. W. Fan, D. M. Ferro, R. Foster, N. M. Galassi, L. D. Gallo, C. M. Green, D. Grosh, K. Q. Ha, M. A. Hasouneh, K. B. Heefner, P. Hestnes, L. J. Hoge, T. M. Jacobs, J. L. Jørgensen, M. A. Kaiser, J. W. Kellogg, S. J. e. a. Kenyon, The Neutron star Interior Composition Explorer (NICER): design and development, in: J.-W. A. den Herder, T. Takahashi, M. Bautz (Eds.), Space Telescopes and Instrumentation 2016: Ultraviolet to Gamma Ray, Vol. 9905 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, 2016, p. 99051H. doi:10.1117/12.2231304.

[19] K. P. Singh, S. N. Tandon, P. C. Agrawal, H. M. Antia, R. K. Manchanda, J. S. Yadav, S. Seetha, M. C. Ramadevi, A. R. Rao, D. Bhattacharya, B. Paul, P. Sreekumar, S. Bhattacharyya, G. C. Stewart, J. Hutchings, S. A. Annapurni, S. K. Ghosh, J. Murthy, A. Pati, N. K. Rao, C. S. Stalin, V. Girish, K. Sankarasubramanian, S. Vadawale, V. B. Bhalerao, G. C. Dewangan, D. K. Dedhia, M. K. Hingar, T. B. Katoch, A. T.

Kothare, I. Mirza, K. Mukerjee, H. Shah, P. Shah, R. Mohan, A. K. Sangal, S. Nagabhusana, S. Sriram, J. P. Malkar, S. Sreekumar, A. F. Abbey, G. M. Hansford, A. P. Beardmore, M. R. Sharma, S. Murthy, R. Kulkarni, G. Meena, V. C. Babu, J. Postma, ASTROSAT mission, in: T. Takahashi, J.-W. A. den Herder, M. Bautz (Eds.), Space Telescopes and Instrumentation 2014: Ultraviolet to Gamma Ray, Vol. 9144 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, 2014, p. 91441S. doi:10.1117/12.2062667.

[20] J. C. Neves Campagnolo, ASTROPOP: the ASTROnomical POlarimetry and Photometry Pipeline, 131 (996) (2019) 024501. arXiv:1811.01408, doi:10.1088/1538-3873/aaecc2.

[21] E. Sánchez, J. Castro, J. Silva, J. Hernández, M. Reyes, B. Hernández, F. Alvarez, T. García, APPHi: Automated Photometry Pipeline for High Cadence Large Volume Data, Astrophysics Source Code Library, record ascl:1804.017 (Apr. 2018). arXiv:1804.017.

[22] J. Bergé, L. Gamper, A. Réfrégier, A. Amara, An ultra fast image generator (ufig) for wide-field astronomy, Astronomy and Computing 1 (2013) 23–32. doi:https://doi.org/10.1016/j.ascom.2013.01.001. URL https://www.sciencedirect.com/science/article/pii/S2213133713000024

[23] P. Lenz, M. Breger, Period04 User Guide, Communications in Asteroseismology 146 (2005) 53–136. doi:10.1553/cia146s53.

[24] J. Hartman, G. Bakos, Vartools: A program for analyzing astronomical time-series data, Astronomy and Computing 17 (05 2016). doi:10.1016/j.ascom.2016.05.006.

[25] A. Adam, G. Marín, A. Christen, L. Cidale, Synthetic light curve design for pulsating binary stars to compare the efficiency in the detection of periodicities, Galaxies 11 (2023) 69. doi:10.3390/galaxies11030069.