Journal of
Astronomical Telescopes,
Instruments, and Systems

AstronomicalTelescopes.SPIEDigitalLibrary.org

Comparison of communication architectures for a fiber-positioning spectrograph

Sofia Martínez-García
Fernando Lopez-Colino
Javier Garrido
Guillermo Glez-de-Rivera
Justo Sanchez
Francisco Prada

Sofia Martínez-García, Fernando Lopez-Colino, Javier Garrido, Guillermo Glez-de-Rivera, Justo Sanchez, Francisco Prada, "Comparison of communication architectures for a fiber-positioning spectrograph," *J. Astron. Telesc. Instrum. Syst.* **5**(1), 014007 (2019), doi: 10.1117/1.JATIS.5.1.014007.

Comparison of communication architectures for a fiber-positioning spectrograph

Sofia Martínez-García,[a] Fernando Lopez-Colino,[a,*] Javier Garrido,[a] Guillermo Glez-de-Rivera,[a] Justo Sanchez,[b] and Francisco Prada[b]
[a]Universidad Autónoma de Madrid, Escuela Politécnica Superior, Madrid, Spain
[b]Instituto Astrofísico de Andalucía, CSIC, Granada, Spain

Abstract. The communication architecture required to provide a bidirectional communication between a central command node and a full set of fiber positioners feeding a spectrograph is studied. Six different architectures have been analyzed in terms of communication time and power consumption. These architectures are the result of the combination of three different communication protocols: transmission control protocol/internet protocol (TCP/IP) over ethernet, interintegrated circuit (I^2C), and controller area network. The design of communication architecture must prioritize between communication time and power consumption. The fastest architecture is the hybrid TCP/IP over ethernet-I^2C. This architecture requires the least time to provide a full set of coordinates to every fiber positioner less than 50 ms. The most power efficient solution is the I^2C—I^2C with demultiplexers. This architecture solves a bidirectional communication between a central node and a full set of fiber positioners requiring only an addition of 27 mW. [DOI: 10.1117/1.JATIS.5.1.014007]

Keywords: astronomical instrumentation; spectrograph; telescopes; communication; fiber positioning system.

Paper 18092 received Oct. 31, 2018; accepted for publication Mar. 7, 2019; published online Mar. 23, 2019.

1 Introduction

The astronomy and physics communities' strong interest in gaining a better understanding of the nature of the accelerated expansion of the universe has triggered the goal, in the next decade, of constructing the greatest spectrography mapping of the universe ever conceived. High-efficiency survey spectrographs are becoming necessary to face the challenge of unveiling the nature of dark energy (see Ref. 1 for a review). These instruments are dedicated to observing massive portions of the sky. They must be as efficient as possible in order to minimize the management of inherent delay, thus allowing for the achievement of the largest possible number of simultaneously observed objects.

The most versatile type of multiobject instrument is the fiber-fed approach, consisting of thousands of robotically controlled fiber positioners feeding a number of cloned spectrographs, e.g., DESI[2], 4MOST,[3] and prime focus spectrograph (PFS).[4] These robots are specially designed to position fibers precisely and quickly, allowing for rapid reconfiguration of the entire field for each observation. The primordial and common problem with these kinds of positioners is the positioning of the fiber ends. These must match the position of the objects in a given sky target field.

Therefore, each robot position must be changed depending on its specific target. The most efficient devices for this task are fiber positioner arrays, which are capable of positioning all the fiber heads simultaneously, making the reconfiguration time extremely short (typically less than 1 min) and increasing the efficiency of the instrument. Each of these positioners is an embedded device—a robot—the optical fibers of which can be positioned with an accuracy within the range of a few microns. The high number (usually thousands) and density of these robots, usually located within a meter-diameter circle, poses a challenge for handling the communication from a central control device (CD) to each of these positioners.

Several projects have adopted the fiber positioner-based approach to achieve their objectives. One example is the Cobra fiber positioner developed for New Scale Technologies for the PFS instrument on the Subaru telescope at Mauna Kea, Hawaii. It uses 2400 robots that position their respective fibers to be analyzed simultaneously with a spectrograph (e.g., Ref. 5). There are also other projects based on the Echidna "tilting spine" technology that offer simultaneous positioning of hundreds to thousands of densely packed optical fibers at a telescope's focus. This technology was originally developed by the Australian Astronomical Observatory and implemented for the 400-fiber multiobject spectroscopy for the Subaru telescope. It has since benefited from a number of mechanical refinements that improve positioning performance, which is now being implemented in 4MOST.[6–9]

DESI[10] is another example of this technology that will use 5000 robotically controlled fiber-positioners feeding 10 identical spectrographs. LAMOST[11] is an on-going project using 4000 positioning fibers connected to 16 spectrographs.[12]

To our knowledge, there has been no extensive analysis related to the implementation of the communication architecture. The controller area network (CAN) bus is a serial bus communication protocol, developed by Bosch in the early 1980s. It is the most used communication protocol in astrophysical instrumentation. It defines a standard for efficient and reliable communication among sensors, actuators, controllers, and other nodes in real-time application.[13] The use of CAN bus is well known in the automobile and aircraft world because it allows for the exchange of a large amount of information among electronic control units, smart sensors, and actuators.[14] However, to our knowledge, there are no additional arguments

*Address all correspondence to Fernando Lopez-Colino, E-mail: fj.lopez@uam.es

2329-4124/2019/$25.00 © 2019 SPIE

that justify its use in astrophysical instrumentation nor are there any comparison to other communication systems, such as the systems for fiber positioning spectrographs. In the HERMES instrument,[15] the use of CANOpen protocol running over the CAN bus is explicitly mentioned as being used for the instrument's electronics. In Ref. 9, the use of CAN bus is mentioned as an example of standard communication because "its use removes the need for almost all ancillary electronics around the field and considerably reduces mass," but there is no comparison made to other communication systems, and the use of this protocol in the instrument is not clearly specified. Schubnell et al.[10] and Lang-Bardl et al.[16] present the use of CAN bus but provide no explanations (project 4MOST), either comparative or argumentative. Schubnell et al.[10] only state that each positioner receives a synchronization signal to facilitate easy co-ordination of the motion of multiple positioners and that each one has a unique ID that can be physically placed anywhere on the bus. Li et al.[17] propose a method of wire communication to control the fiber positioning nodes, which are driven by two stepping motors. The authors used a CAN bus and proved that the system has good real-time performance and stability. Li et al.[17] mention that the use of a serial communication system as CAN bus is justified because "it is widely used among industry circle because of its outstanding reliability, real-time capacity, and flexibility." Also, Schubnell et al.[18] propose the use of CAN for DESI but in combination with standard Internet-like communication in a different level. This hybrid approach uses a dedicated CAN bus for each of the 10 segments of the focal plate. Each bus is managed by an embedded device, which communicates with a central computer using transmission control protocol-internet protocol (TCP-IP) over ethernet. Table 1 presents a brief summary of the different instruments that have been reviewed and their communication architecture-related characteristics.

Based on our previous experiences in the analysis of different communication protocols,[20] this paper presents a comparison and evaluation of several embedded-system-oriented serial communication protocols, including CAN bus, wired Internet, inter-integrated circuit (I^2C) protocol, and a combination of these. For the later comparison, we have evaluated different parameters comprising hardware complexity, time of communication, and power consumption. The proposed communication architectures are positioner-technology independent. Thus, they can be applied to every fiber positioner approach, such as Cobra,[21] Echidna,[22] or Megara,[23,24] that receives orders from a central command node using a digital communication protocol.

The paper is organized as follows. Section 2 describes the communication architectures with requirements, protocols, and proposed architectures. Section 3 presents the results with a comparison of architectures. Different scenarios were studied, and the parameters of communication time and power consumption are analyzed and discussed. Finally, Sec. 4 presents the conclusions.

Table 1 Summary of the different state-of-the-art instruments.

Instrument	# Fiber positioners	Communication architecture	Reference
DESI	5000	Hybrid ethernet-CAN	Ref. 18
4MOST	2400	CAN-bus	Ref. 9
PFS	2550	No data available	Ref. 5
LAMOST	4000	Zigbee protocol	Ref. 12
MEGARA	100	CAN-bus	Ref. 19

2 Communication Architectures

The communication architecture must solve the transmission of the positioning information from a central node (usually a personal computer) to each fiber positioner. First, the requirements and conditions that any proposal must fulfill to solve the problem will be presented. Second, the most suitable communication protocols are presented and reviewed. Finally, different architecture proposals are depicted.

2.1 *Architecture Requirements*

The architecture must solve the communication between a central command node, known as the master controller (MC), and each of the fiber positioners, known as the final devices (FDs). This requirement defines a tree-like architecture. The MC is the upper node, and thousands of FDs are at the bottom. The intermediate communication elements, the CDs, required by each particular technology will define intermediate layers. The information is mainly transmitted from the MC node to each of the FDs. However, some information could need to be retrieved from an FD, such as the current status of the positioner (e.g., motor status, electronic status, command acknowledgement, unexpected error, etc.). This requirement necessitates a bidirectional communication architecture.

The next step is to define the information that has to be delivered. To define the size of the data packages, it is necessary to review what information has to be transmitted.

This information can be segregated into instructions and the parameters for these instructions. The limited amount of operations allowed by a fiber positioner (e.g., position, position with path, reboot, status report, disconnect, etc.) implies a low number of instructions. Therefore, using a byte to identify these instructions will suffice. The same applies to the status report that each positioner may provide (status OK, motor error, format error, etc.). A byte would be enough.

Regarding the parameters for the instructions, the most complex instructions are the ones related to position. Although there are different positioner technologies, every fiber positioner can transform a coordinate within its patrol area to its own system of reference. The amount of data required to define a coordinate is the relationship between the patrol area width (PAW) and the positioners resolution (PosR) [Eq. (1)]:

$$\text{bits} = \left\lceil \log_2\left(\frac{\text{PAW}}{\text{PosR}}\right)\right\rceil. \tag{1}$$

A generic approach has been selected, defining a coordinate within the patrol area using a X, Y coordinate system. If 2 bytes are used for the axis X or Y, it would be possible to define $2^{16} = 65{,}536$ different positions per every single axis. For example, this would be enough to cover a 10-mm wide patrol area with a 0.15-μm precision. This far exceeds current fiber positioner requirements established by the projects mentioned in the previous section.

One coordinate (4 bytes) would suffice to specify the target to the positioner. However, the positioner's technology may require the inclusion of a collision-avoidance strategy. The Echidna positioner[9] handles the collisions by using flexible

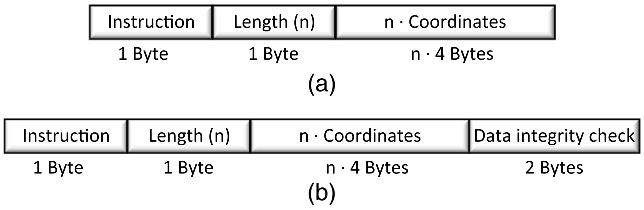

Fig. 1 Data package definition for the move instruction transmitted to an FD: (a) the basic structure and (b) extended message including a data integrity check field.

spines. Therefore, no particular measure has to be considered to avoid the collisions. Others that have rigid parts[5,10] have to avoid the collision between adjacent positioners. Some solutions have reported that handling the movement speed would suffice for avoiding the collision.[25] This approach requires including a parameter along with the coordinate to specify this speed. Also, following a particular path to reach a final destination has been reported as a way to avoid collisions.[20] This path is defined by a variable number of coordinates. Hence, the instruction that tells a positioner to follow a path requires two parameters: the number of transmitted coordinates and the coordinates themselves. An initial estimation has reported that a path of 16 coordinates or fewer is enough to avoid collisions.[26] One byte for this length of field would suffice. Therefore, the largest amount of data to be sent to a single positioner consists of the move instruction and its two parameters: the field specifying the number of coordinates and 16 coordinates. The structure depicted in Fig. 1(a) presents 66 bytes. A requirement of the communication architecture is the ability to handle variable data lengths. The architecture also must be able to segment this data into different transmission blocks and to recompose these blocks at a final destination.

The communication architecture must ensure data integrity during the transmission. This integrity can be achieved using data verification fields. Several communication protocols include these fields within the communication overhead. For those protocols that do not include this kind of verification, an extra field must be included [Fig. 1(b)].

Finally, a project requirement may impose a time and power consumption limitation to the whole fiber-positioning system. These two parameters will be considered for the different communication architectures and protocols. This project requirement imposes a time limitation of a few seconds to send hundreds of bits to thousands of fiber positioners. The communication speed should pass this transmission rate. This time limitation will discard the protocols that do not reach a speed of 1 megabit per second (Mbps). The power consumption is considered in the final architecture comparison to select from the solutions that comply with the time limitation. The main reason of considering power consumption is its relationship with heat dissipation. Heat may interfere with the observation process due to air perturbations.[18]

2.2 *Communication Protocols*

Once the requirements have been analyzed, the communication architecture should comply with the network layer definition established by the OSI standard[27] to ensure that it can transmit any length of data over multiple networks using different addressing systems. Regarding the simplification provided by a multidrop-based protocol and its achieved speeds, the best candidates for the communication architecture are based on the CAN protocol and the I^2C protocol. However, neither of these fulfill the OSI network layer definition, which requires the inclusion of additional extensions in these protocols. Finally, TCP/IP is not recommended for the use with fiber positioners as the exclusive protocol due to its complexity and wire connectivity. However, this protocol complies with the time requirements and can be used as a solution when mixing different protocols. We will present the main characteristics of these three protocols (CAN, I^2C, and TCP/IP).

1. CAN: The ISO 11898-2 standard[28] is a serial asynchronous protocol that uses a differential bus for the transmission of messages in a distributed environment. The differential bus reduces sensibility to interferences when used in noisy environments.[29] This is a multimaster broadcast half-duplex serial bus technology in which all nodes may receive and send data. The ISO 11898-2 standard defines a maximum speed of 1 Mbps over a differential pair allowing a maximum of 2048 devices in the same network. This version of CAN defines an overhead of 44 bits [CAN header (19 bits) plus CAN tail (25 bits)] for a maximum of 64 bits of data [Fig. 2(a)]. However, CAN only defines the data link layer of the OSI model and requires an extension to fulfill the network layer. To achieve the required OSI network layer, two CAN protocol extensions have been selected. These are CANOpen[30] and ISO 15765-2.[31] These protocols use a variable number of bytes within the CAN package's data field to achieve OSI network layer functionality.

- CANOpen, the first byte of the CAN data field is used for the CANOpen header [Fig. 2(a)], allowing for the segmentation of the data into multiple packages. The protocol requires an acknowledgement response for every transmitted package [Fig. 2(b)]. Hence, when using CANOpen, the overhead is increased to 104 bits [44 from standard CAN protocol, 8 for the CANOpen overhead, and 52 for the CANOpen acknowledgement (ACK) package] for a maximum of 56 bits of transmitted data (instead of 64). However, CANOpen allows data segmentation over multiple packages but not over multiple networks. Thus, it can only be used in particular architectures.

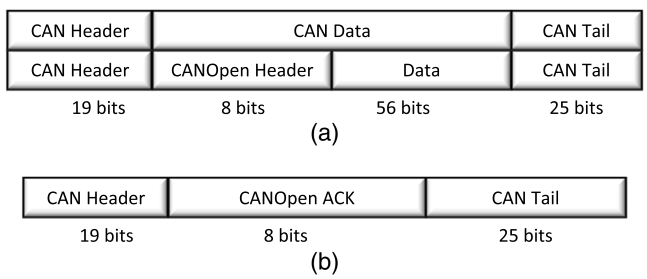

Fig. 2 Transmission packages for the CANOpen standard: (a) data package allowing up to 56 bits of data and (b) acknowledgment package sent for every data package.

CAN Header	CAN Data		CAN Tail
CAN Header	ISO 15765-2 Header	Data	CAN Tail
19 bits	16-24 bits	48-40 bits	25 bits

(a)

CAN Header	ISO 15765-2 ACK	CAN Tail
19 bits	8 bits	25 bits

(b)

Fig. 3 ISO 15765-2 standard packages: (a) package structure defined in the ISO 15765-2 standard and (b) acknowledgment package in the ISO 15765-2 standard.

- ISO 15765-2 allows for the segmenting of data over multiple packages by sending them over multiple networks. The overhead is increased compared to CANOpen to fulfill the network layer. ISO 15765-2 requires the use of 16 or 24 bits (depending on the transmission) from the CAN data field to achieve its functionality [Fig. 3(a)]. Also, this protocol uses an acknowledgement package of 68 bits [CAN header plus CAN tail plus 24 bits required by the protocol, Fig. 3(b)]. Hence, as shown in Fig. 3(a), sending 48/40 bits of relevant data requires the transmission of additional 128/136 bits (CAN header and tail, ISO 15765-2 header, and ISO 15765-2 ACK package).

ISO 15765-2 provides more functionality compared to CANOpen by increasing the overhead. The proposed architectures (see Sec. 2.3) will use the protocol that fulfills the required functionality with the lower overhead.

2. I^2C is a serial synchronous communication protocol that uses two ground-referred bus lines to transmit information.[32] It is a half-duplex protocol with a master–slave relationship between the different nodes in the bus. The master always starts the communication by addressing one slave and sending or requesting data, allowing variable data packages.

The I^2C protocol establishes a 10-bit header, 1 bit tail, and an extra ACK bit for every transmitted byte (Fig. 4). The protocol reserves 7 bits for addressing purposes. Therefore, it allows a theoretical maximum of 127 possible addresses within the same network.

To fulfill the network layer, it is possible to extend the address using extra bytes of the data section. Similar to the CAN protocol,[31] by using bytes from the data field, it is possible to extend the functionality of this protocol (field FD_{Addr} in Fig. 8). The maximum speed of the I^2C protocol is set by the "high speed" mode at 3.4 Mbps. However, the real speed, distance, and number of nodes per network are limited by the bus' electrical capacity. I^2C is practically limited to less than 3 m for moderate speeds.[33] It is also possible to increase the tolerance of the bus by means of bus amplifiers.[34]

3. TCP/IP over ethernet: The higher complexity of this protocol requires extra capability from the fiber positioner's microprocessor. Moreover, the wire distribution of ethernet requires connecting a cable from each FD to a switch (star bus) rather than the single multidrop bus shown in both previous protocols. Therefore, it is discarded by its exclusive usage for reaching to the FDs. However, this protocol can be useful for a hybrid approach in the communication between the MC and the CDs due to its high speed (up to 10 Gbps) and its presence in a personal computer, which would perform as the MC. Each of the three protocols includes some overhead to fulfill the OSI network layer. The overhead in this protocol is 64 bytes (18 bytes from ethernet, 24 bytes from IP, and 20 bytes from TCP) allowing the transmission of 1456 bytes of useful data [Fig. 5(a)]. However, for each transmitted package, TCP requires the transmission of an acknowledgement package for a total amount of 64 bytes [Fig. 5(b)].

I^2C Header				Data					I^2C Tail
Start	Address	R/W	ACK	$Data_1$	ACK	…	$Data_n$	ACK	End
1 bit	7 bits	1 bit	1 bit	8 bits	1 bit		8 bits	1 bit	1 bit

Fig. 4 I^2C transmission structure.

2.3 *Proposed Architectures*

This section will describe different architectures combining the former three communication protocols. Each tree-like architecture presents the MC as the top element and all the FDs in the bottom of the structure, with CD in the intermediate layers. The CD component solves the communication between different subnetworks of the communication architecture. The complexity of this component is different for each proposal and will be described for each of them.

2.3.1 *Single architectures*

Full CAN architecture approaches the solution to the communication system by using just the CAN protocol (see Fig. 6). The main limitation of every embedded communication protocol is the real number of devices in the same bus. The higher the number of devices, the higher the capacitance of the bus, which reduces the communication speed.[35] To avoid this speed reduction, the network must be divided into several interconnected subnetworks. Considering that the transmitted information may exceed the 8 bytes allowed by the CAN protocol and faces the necessity to transmit over different networks, a full CAN architecture requires the use of the ISO 15765-2 extension. This protocol requires that an acknowledgement be transmitted from the receiver to the sender for every received package. This ACK package is depicted in Fig. 3(b).

This full CAN architecture comprises a level 1 network composed of the MC and m CDs (Fig. 6) and m different networks composed of the second interface of the CD and a variable number (n) of FDs to cover the number of fiber positioners of the instrument. The number of devices in each network will define the maximum speed of the architecture. In this architecture, the CDs are routers, which have two CAN interfaces to two different networks. In the level 1 network, the package is addressed to an FD from the MC using the CAN-ID. The extra address byte provided within the ISO 15765-2 header identifies the FD in the lower network.

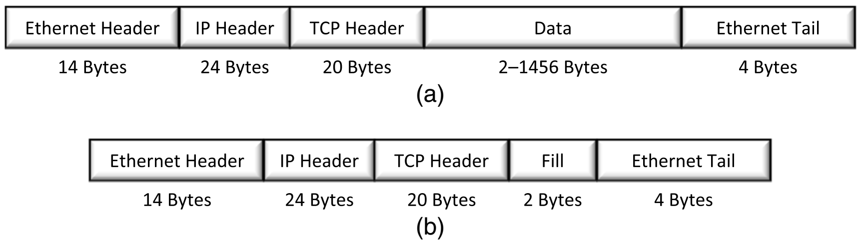

Fig. 5 Ethernet packages structure: (a) ethernet, IP, and TCP overheads and (b) acknowledgment package in the TCP/IP over ethernet communication.

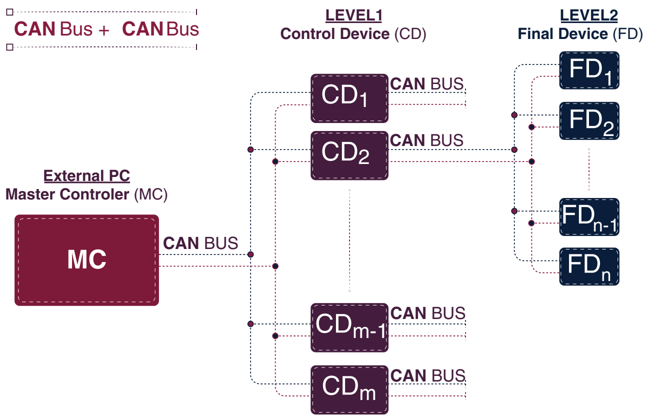

Fig. 6 Diagram of the full CAN proposal.

If the MC requires information from a particular FD, it will send the request through the corresponding CD. The CD will forward the request from the upper network to the bottom one. When the FD has prepared the information, it will send it to the CD. The CD will, again, forward the response to the upper network.

The full CAN approach is the most popular solution adopted in state-of-the-art systems,[10] including the communication architecture in the HERMES project[15] and probably in the 4MOST project.[9] While the authors note the use of this protocol, they neither provide technical details nor explain the communication architecture.

The full I^2C architecture approaches the solution to the problem using only the I^2C protocol. Since I^2C allows a maximum of 127 nodes per network, a multilayer network is required (see Fig. 7). The main structure is similar to the full CAN architecture. This approach requires the CDs to be routers between the level 1 and the level 2 I^2C networks. Both levels of the network have one master: the MC in the level 1 network and the CD in the level 2 network.

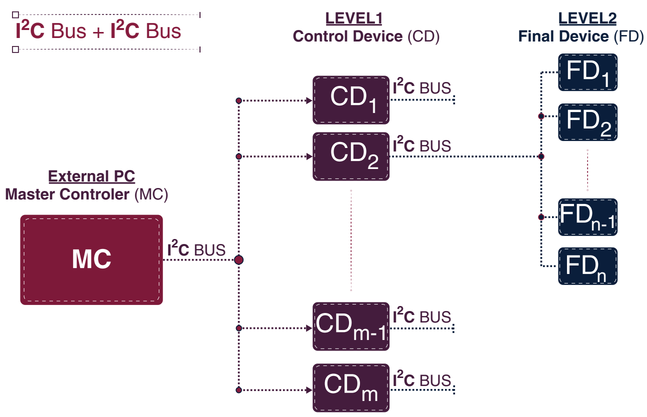

Fig. 7 Diagram of the full I^2C proposal.

Since the I^2C protocol does not limit the size of the package, the only required additional information is an extended address. The first byte of data is used to extend the address in this protocol (see Fig. 8). In this network, the MC includes an extra byte in the data field to tell the CD which of the FDs in the level 2 I^2C network has to receive the package. This package is sent to the corresponding CD and then the CD removes the extra address byte and transmits the rest of the package to the FD, using the package structure shown in Fig. 4.

The main advantage of the I^2C bus is its allowance of demultiplexers. These demultiplexers permit the division of an I^2C bus into several independent buses. The input of the demultiplexer is connected to the CD, whereas multiple sets of FDs are connected to different outputs of the demultiplexer (shown in Fig. 9). This division reduces the global capacitance of the bus, thus allowing higher speeds. However, these demultiplexers require a configuration that consumes time in the process. The package sent to the demultiplexers ($\text{demux}_{\text{Addr}}$) commands the demultiplexer to enable a particular output, as depicted in Fig. 10. This configuration has to be completed before the transmission from the CD to the FD (to enable one output) and again after the transmission (to disable the output).

Bidirectionality can be solved by means of a two-phase request. First, the MC sends a write I^2C package to the CD. The CD forwards the write package to the FD and starts a read package to retrieve the answer from the FD. Then, the MC starts the read package and retrieves the information from the CD. Therefore, the MC must idle between the write and read packages to allow the CD to complete the back and forth transmissions.

To our knowledge, no instrument has adopted this technology for the communication architecture.

2.3.2 *Hybrid architectures*

Ethernet-CAN: The third proposal is a hybrid approach using a TCP/IP over ethernet network and CAN. This protocol allows higher speeds compared to CAN. On the other hand, the CDs perform as gateways. The tasks of the CD are more demanding with this architecture since the incoming packages must be processed from a TCP/IP interface, whereas the message and management of the corresponding communication is processed through a second interface. The level 1 network is TCP/IP and the level 2 networks are CAN networks (see Fig. 11).

The tasks require the CD of this architecture to have more computational capacity. However, this has several advantages: the information sent in the level 1 network can concatenate

I^2C Header				Data							I^2C Tail
Start	CD_{Addr}	R/W	ACK	FD_{Addr}	ACK	$Data_1$	ACK	...	$Data_n$	ACK	End
1 bit	7 bits	1 bit	1 bit	8 bits	1 bit	8 bits	1 bit		8 bits	1 bit	1 bit

Fig. 8 I^2C with extended addressing for both the CD (in the 7-bit header field) and the FD (the first 8-bit transmitted data).

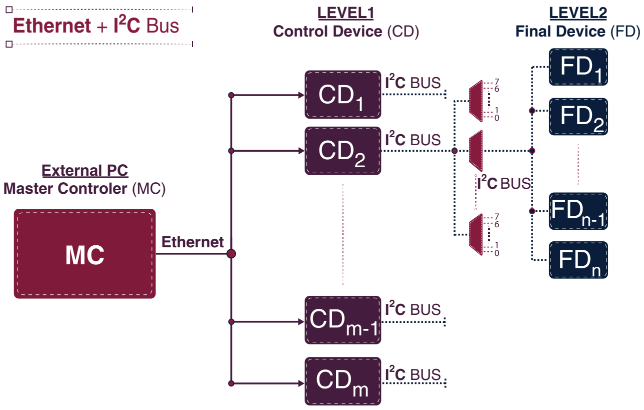

Fig. 9 Diagram of the ethernet-I^2C hybrid proposal. This diagram shows the usage of demultiplexers in the connection between a CD and multiple FDs.

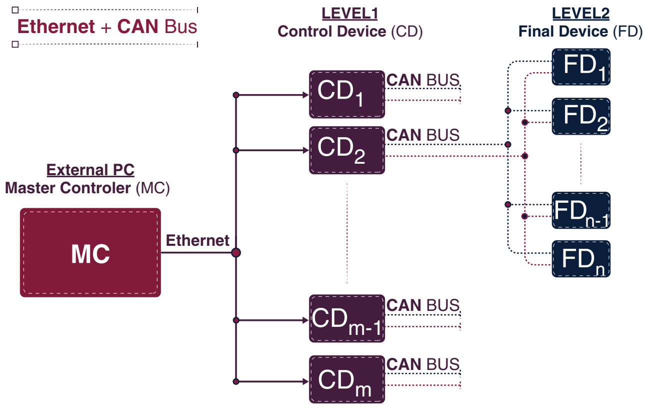

Fig. 11 Diagram of the ethernet-CAN hybrid proposal.

information to multiple FDs, optimizing the use of the TCP/IP package. Also, this protocol allows for higher transmission speeds. Hence, the global communication time is reduced. Since every CD may receive and store the information of its FDs, the different CDs can perform their transmission to their FDs in a parallel way in the lower, and slower, connections. Thus, the required time to transmit to every FD is also reduced. The package structure for this upper network is depicted in Fig. 12. This package is transmitted using TCP/IP over ethernet [Fig. 5(a)]. The TCP protocol requires a TCP ACK [Fig. 5(b)].

This architecture uses CAN networks to connect a single CD with a set of FDs. This CAN network still requires that data to be sent over multiple packages but not over multiple networks. This simplification allows for the use of CANOpen, which package structure, as used in the CAN network, is depicted in Fig. 2(a).

The communication between the MC and the CD follows a client-server scheme. This communication is managed by TCP/IP sockets. When the MC requires some information from an FD, it sends the request to the corresponding CD. The CD generates the request to the FD using CAN packages. However, the higher computational capacity of the CD permits the management of its FDs on its own initiative. The DESI project implements this solution using a BeagleBone as CD.[18]

EthernetI^2C: The last architecture discussed is the combination between TCP/IP over ethernet and I^2C (Fig. 9). Similar to the former architecures, the upper level is a high speed TCP/IP network and the CD must receive packages and create messages to the FD through an I^2C bus. The approach of this architecture also requires the CDs to perform as gateways and has higher computing capabilities to process the TCP/IP package and manage the I^2C bus.

Similar to the full I^2C network, it is possible to use I^2C demultiplexers in the lower networks. These demultiplexers will lower the bus capacitance and allow for higher speeds. Finally, the bidirectional communication is solved in the same way as in the ethernet-CAN architecture.

To our knowledge, no instrument has adopted this technology for communication architecture.

2.3.3 *General considerations*

The architectures that use a single protocol generate an end-to-end communication. This means that the MC sends the required packages to each FD addressing the corresponding CD. The CD gets the package and retransmits it to the FD. When the ACK is a different package, the FD sends the ACK to the CD, and the CD retransmits it to the MC.

In the architectures that use a combination of protocols, the CD stores all of the information for its FDs. Once all the information is received, it creates the required CAN/I^2C transmission packages for every addressed FD. This allows the different CDs to transmit information to their FDs simultaneously.

3 Architecture Comparison

This section will compare the different simulations of architecture compositions to interconnect 5000 FDs with a single MC. The number of FDs has been selected because it is the largest number of fiber positioners that will be used in a single telescope.[10,36,37]

I^2C Header				Data				I^2C Tail
Start	$demux_{Addr}$	W	ACK	Command	ACK	Selected output	ACK	End
1 bit	7 bits	1 bit	1 bit	8 bits	1 bit	8 bits	1 bit	1 bit

Fig. 10 Configuration package sent to the I^2C demultiplexer.

Instruction	Length	$Address_{FD1}$	$Data_{FD1}$	$Address_{FDn}$	$Data_{FDn}$
1 Byte	4 Bytes	2 Bytes	1–68 Bytes	2 Bytes	1–68 Bytes

Fig. 12 Data contents sent from the MC to a CD.

Table 2 Communication scenarios.

Scenario	Addressed FDs	Coordinates
1	All	16
2	All	1
3	One	16
4	One	1

3.1 *Communication Scenarios*

We discuss four different scenarios. These scenarios have been defined to represent different iterations within the plate configuration. The first iteration would probably require moving every positioner to a target. The following iterations would be made to reduce positioning errors. The number of these iterations and the affected positioners depend on their technology. The higher the positioner's precision, the fewer the number of positioners affected in each iteration and the lower the number of iterations.

Scenario 1 represents a full repositioning of the whole system with the worst-case collision avoidance solution. Therefore, it is necessary to provide all FDs with a complete set of 16 coordinates. Scenario 2 is a full repositioning of the whole system with the best-case collision avoidance solution, thereby providing all FDs with a single coordinate. This scenario represents an adjustment iteration of the whole system after a full reconfiguration. Scenario 3 represents a single FD reconfiguration with a full set of 16 coordinates. Finally, scenario 4 provides a single FD with a single coordinate. Table 2 summarizes these scenarios.

3.2 *Parameters Study*

To analyze the different architectures, two parameters have been defined to make the comparison. These parameters are (1) the required time to fulfill each scenario and (2) the power consumption of each architecture. The results of the communication time between the MC and the FDs of each architecture have been presented and explained in Sec. 3.3. Next, in Sec. 3.4, the power consumption by architecture has been estimated considering the different hardware requirements of each of them. For both parameters, the independent variable in each comparison is the number of CDs in each architecture.

The I^2C-based architectures present an addressing limitation. These architectures do not depict the required time when the number of CDs is higher than 123, because this communication protocol allows for up to 127 different devices with four used by the demultiplexers. Therefore, it is only possible to have up to 123 CDs in the same bus.

3.3 *Communication Time*

The field configuration time comprises many sequential processes, each with a particular time: the communication to the fiber positioners, collision-avoidance calculations, the fiber movement, and the verification of the fiber's position by the metrology. This sequence may be repeated in several iterations if the fibers require repositioning to achieve a defined precision. This comparison will study the required time to transmit the information defined in each scenario using the different architectures. The other processes depend on their own characteristics and are independent of the communication architecture.

For each architecture, the amount of time depends on the amount of transmitted information (the transmitted data, the overhead introduced by each protocol, the acknowledgement packages or bits, and the required configuration time) and the communication speed. The transmitted data are depicted in Fig. 1(a), where the number of coordinates depends on the scenario. This comparison will only consider communication procedures that may reduce the observation time. Other communication procedures, such as a positioner's firmware update or modifications of a positioner's configuration tables, may be done during engineering maintenance time. The overhead and acknowledgement packages are architecture-dependent and were presented in Sec. 2.2. The extra configuration required by demultiplexed I^2C buses is presented in Fig. 10.

The second element to be established prior to time comparison is the communication speed. For the TCP/IP over ethernet link, we selected a speed of 100 Mbps. However, it is possible to achieve a higher communication speed using consumer electronics. The same applies to the selection of the I^2C speed. This protocol permits the use of a 3.4 Mbps speed. However, we limited the I^2C speed to the maximum allowed for a CAN bus (1 Mbps) to simplify the architecture comparison. Table 3 summarizes the speed used for every communication protocol.

The following sections present the results for each of the four scenarios. It is necessary to notice that some of the proposed architectures are not viable for every number of CDs.

3.3.1 *Scenario 1: full configuration, worst case scenario*

As discussed above, the first scenario represents a full configuration of the whole positioner system, requiring each positioner to follow the largest possible path (a sequence of 16 coordinates). The time required by each architecture has been depicted in Fig. 13. The architectures that are composed of a single technology present a constant time, independent of the number of CDs. The hybrid approaches show an improvement when increasing the number of CDs. This improvement is due to

Table 3 Communication speed per protocol.

Communication protocol	Speed
CANOpen	1 Mbps
CAN ISO 15765-2	1 Mbps
I^2C	1 Mbps
TCP/IP	100 Mbps

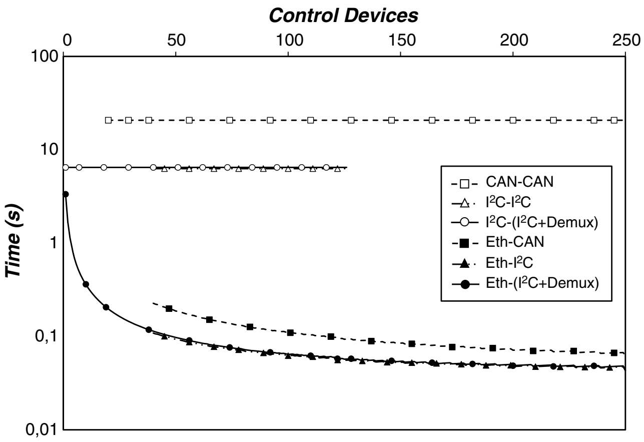

Fig. 13 Time to configure 5000 FDs with 16 coordinates each (scenario 1) as a function of the number of CDs for different architectures. Partial lines refer to the protocols limitation to the allowed maximum number of devices.

the parallelism that can be achieved by these CDs managing their bus simultaneously. However, with more than 120 CDs, the time reduction is less significant. The different steps shown in the hybrid architectures respond to the required packages segmentation over the TCP/IP link. When the number of CDs rises, the information sent to each of them lowers. Every time this information requires one less package, it saves the package overhead and the corresponding acknowledgement.

3.3.2 *Scenario 2: full reconfiguration with a single coordinate*

The second scenario represents a full reconfiguration of the whole positioner system. This circumstance happens after the first positioning iteration, when the MC uses the fiber-view camera feedback to adjust the position of each fiber positioner. In this scenario, every fiber positioner requires some adjustment but their movements will not incur any collisions.

Similar to the first scenario, Fig. 14 shows the time required by each of the architectures to fulfill this scenario. The single-protocol architectures are not influenced by the number of CDs, whereas the hybrid architectures show no performance increase over the 120 CDs.

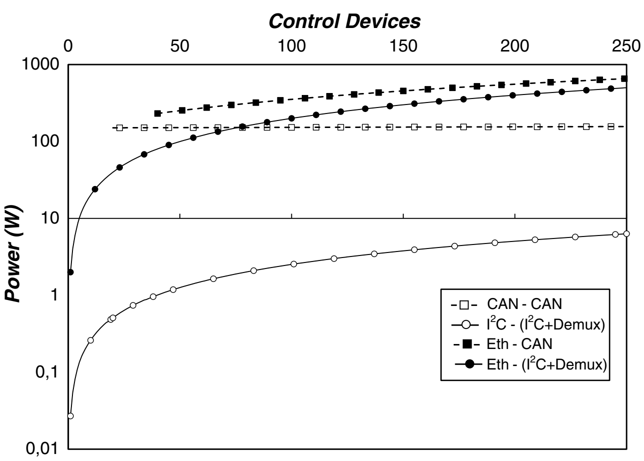

Fig. 14 Time to configure 5000 FDs with one coordinate each (scenario 2) as a function of the number of CDs for different architectures. Partial lines refer to the protocols limitation to the allowed maximum number of devices.

3.3.3 *Scenario 3: full configuration of a single positioner with 16 coordinates*

The third scenario shows a single positioner that must move to another position while following the most complex path in order to avoid colliding with its neighbors. For this scenario, the number of CDs is irrelevant. The communication to a single FD only involves one CD. The results for this scenario have been depicted in Table 4.

3.3.4 *Scenario 4: single positioner reconfiguration with a single coordinate*

The fourth scenario shows a single positioner that must move to another position but does not need to avoid its neighbors. For this scenario, the number of CDs is also irrelevant. The communication to a single FD only involves one CD. The results for this scenario are depicted in Table 5.

3.4 *Power Consumption*

Based on standard commercial components, power consumption has been calculated for each of the previously proposed architectures. This study aimed not only to compare raw power consumption but also the thermal effects in the sensors plate. Due to the fact that the communication components that are away from the plate do not have a thermal impact, their power consumption has not been evaluated. This is the case

Table 4 Scenario 3 time measurements.

Proposed architecture	Required time (s)
TCP/IP - (I^2C + Demux)	6.79×10^{-4}
TCP/IP - I^2C	6.39×10^{-4}
I^2C - (I^2C + Demux)	1.92×10^{-3}
I^2C - I^2C	1.26×10^{-3}
TCP/IP - CANOPEN	1.58×10^{-3}
CAN - CAN (ISO15762-2)	4.14×10^{-3}

Table 5 Scenario 4 time measurements.

Proposed architecture	Required time (s)
TCP/IP - (I^2C + Demux)	1.34×10^{-4}
TCP/IP - I^2C	9.41×10^{-5}
I^2C - (I^2C + Demux)	2.98×10^{-4}
I^2C - I^2C	1.75×10^{-4}
TCP/IP - CANOPEN	1.63×10^{-4}
CAN - CAN (ISO15762-2)	3.52×10^{-4}

with the TCP/IP ethernet router, which can be placed away from the plate. Also, the elements that are common to all architectures, such as the MC or the positioners themselves, have been also excluded from the comparison.

We discuss two grades of complexity for the CD: a simple routing task in the single-protocol architectures and a more complex task for the hybrid solutions. The routing task can be completed with a low power unit handling two interfaces (I^2C or CAN). This simpler unit has been denoted as CD router. The Sammy-C21[38] is a microcontroller-based module, which includes all the required electronics to perform as a CD router in all of the proposed architectures. It fulfills the requirements for this task. The hybrid architectures require the CD to convert between TCP/IP over ethernet into I^2C or CAN. This task is more demanding for the CD, denoted as CD gateway. The CD gateway requires higher computing capacity than the CD router. Therefore, it will be more power demanding. A BeagleBone has been selected to carry out this task, specifically the model Black.[39]

Every CAN interface requires a transceiver. This requirement applies to each CD and FD to physically connect to the CAN bus. As an example of a real transceiver module, the model SN65HVD1040 from Texas[40] has been selected due to its low power consumption. This power consumption value is considered only in its recessive state for every CAN node. An I^2C demultiplexer (demux) is required for the corresponding architecture related to the I^2C communication protocol. For this study, model PCA9548A from Texas[41] has been selected as an example of a real component. The selected devices are examples of those frequently used in this area that can carry out the required functionality.

Table 6 shows the elements that were considered in the study of power consumption. Different commercial devices were selected for the different elements of the communication architectures. The power consumption was obtained from their respective data sheets.

Once these devices were selected, the analysis was completed and the conclusions are as follows:

1. The main power consumption is due to CD gateways. The consumption increases when the number of CDs used in the architecture grows.
2. Comparing the different serial protocols, I^2C versus BusCAN, the necessary presence of one BusCAN driver beside each of the 5000 FDs implies an important increment of consumption for the BusCAN architectures.

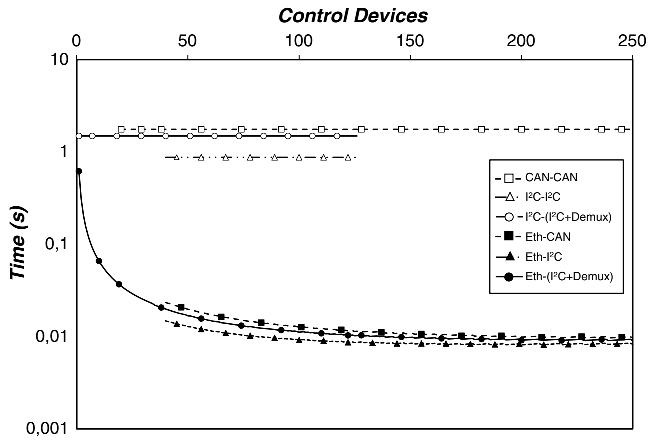

Fig. 15 Power consumption comparative for different architectures. Partial lines refer to the protocols limitation to the allowed maximum number of devices.

Figure 15 shows power consumption among the evaluated architectures. For the sake of clarity, the architectures related to I^2C protocol without demultiplexers have not been depicted, because the results are very similar to those that use them. The most efficient architecture in terms of power consumption is the $I^2C-(I^2C+\text{demux})$. It provides a solution hundreds of times lower in power consumption. The next best option, in terms of power consumption, is the hybrid TCP/IP—I^2C architecture, if the number of CDs is relatively low (fewer than 80). If using a higher number of CDs, the CAN–CAN proposal becomes a better option than TCP/IP—I^2C. Finally, the TPC/IP-CAN architecture is the most power-consuming alternative.

4 Conclusions

This paper presents a comparative analysis of different communication architectures to be used in a multifiber positioner spectrograph. These architectures allow for a bidirectional communication between a central node (a personal computer referred to as MC) and the complete set of fiber positioners

Table 6 Electrical elements included in the power consumption analysis.

	Architecture component, device, and power consumption				Power consumption	
Architecture	CD gateway Beaglebone Black 2.0 W	CD router Sammy-C21 2.5×10^{-2} W	CAN Interface SN65HVD1040 3×10^{-2} W	I^2C demux CA9548A 2.5×10^{-4} W	#CD = 1	Average ΔP/CD
TCP/IP - I^2C (+Demux)	Yes	No	No	Yes	2.0 W	2.0 W
TCP/IP - I^2C	Yes	No	No	No	2.0 W	2.0 W
I^2C - I^2C (+Demux)	No	Yes	No	Yes	$2.7\cdot10^{-2}$ W	2.5×10^{-2} W
I^2C - I^2C	No	Yes	No	No	$2.5\cdot10^{-2}$ W	2.5×10^{-2} W
TCP/IP - CANOpen	Yes	No	Yes	No	2.03 W	2.03 W
CAN - CAN (ISO15762-2)	No	Yes	Yes	No	150.085 W	8.5×10^{-2} W

of the spectrograph (FDs). The proposed communication architectures are positioner technology-independent.

This study considered the highest number of fiber positioners (5000) found in state-of-the-art technology. Time and power results in this work are simulation based. The study assumes the necessity of including a second layer of devices (CDs) to achieve bidirectional communication between the MC and every FD.

The analysis considered the number of CDs as the independent variable and evaluated the communication time in four representative scenarios and the power consumption of each of the six alternative architectures. The communication process, as part of the spectrograph configuration process, reduced time from the observation process. The power consumption of the communication architecture was related to the thermal effects that may have affected the focal plate.

The time comparison evaluated four representative scenarios in the spectrograph operative. The most common scenarios involved the full positioner system configuration with the most complex possible path (scenario 1) and the first iteration for adjusting every fiber position (scenario 2). The evaluation of the theoretical time analysis showed that the lowest time was achieved with a hybrid combination of TCP/IP over ethernet-based communication between the MC and the CDs, and an I^2C communication-based architecture between the CDs and its related FDs. The inclusion of I^2C multiplexers in this second communication not only produces a minor penalization to the time results but also reduces the capacitance penalization in a real implementation.

Every hybrid proposal improved the timing results compared to the full CAN-bus solution depicted in some state-of-the-art projects that use this solution. The hybrid ethernet-I^2C architecture showed a time reduction in the hundreds.

The communication time is one process within the field configuration procedure. After the instructions have been transmitted to the positioners, they move the optical fibers to the requested position. This mechanical movement implies an amount of time that depends on the technology. After this movement, the metrology must measure the positioning error and determine which fibers must be corrected. After the coordinates have been computed, the collision-avoidance algorithm must verify that every path is collision free or generate alternative paths. Once this procedure is finished, a set of instructions is ready to be sent to the positioner, leading to an iteration. The more time required for the whole process, the lower the time reserved for observation. For example, the DESI project[10] imposed a 20-s time frame for the whole reconfiguration process. This requirement eliminates some of the proposed architectures that exceed this value for scenario 1. Depending on the selected positioner technology that defines the actuators speed and the iterations required to achieve the required precision, the metrology system, and the collision-management algorithm, the communication architecture must be selected such that it can fulfill its task in the remaining time.

The power comparison has exclusively focused on the different components of the communication architectures that may affect the observation due to their heat generation. Any component that may be placed away from the focal plate was ignored. Taking into account that the power consumption is a theoretical approximation, other factors, such as PCB track widths, number of layers, wire size, etc., which can influence the real system's power consumption, were not considered.

The main difference between the I^2C protocol and the CAN bus is the requirement of a driver for the latter. This driver must be placed in every FD and CD CAN interface. There is also a relevant difference between the hybrid architectures and the single protocol ones: that being the complexity of the CD. The hybrid architectures require a complex CD unit (CD gateway) to analyze TCP/IP over ethernet information and handle it through the I^2C/CAN interface. In single architecture protocols, the simpler CD (CD router) allows for the use of components with lower power requirements. The absolute minimum is achieved when using the full I^2C with demultiplexers. The state-of-art CAN-based solution provides worse results than the hybrid TCP/IP over ethernet-I^2C when using fewer than 80 CDs.

The power required by the communication architecture is not the only source of heat near the focal plate. The positioner's electronics and actuators are the main source of heat near the focal plate. Fahim et al.[26] report an average power consumption of 600 mW per positioner during its action. The most power-demanding solutions, the ones based on CAN bus, would increase this value by 30 mW (i.e., 5%). This increase would not exceed the 1.2 W limit imposed in their project. Likewise, disconnecting the communication electronics can be done as this also disconnects the motors and main electronics.

The best architecture definition depends on the designer's criteria: if the communication speed is to be maximized, the best choice is a hybrid TCP/IP over ethernet-I^2C architecture. On the other hand, if the power consumption is to be minimized, the best alternative is an I^2C architecture combined with I^2C demultiplexers. The current design tendency of using a CAN bus-based communication architecture is not justified considering these results. Systems power consumption is not constant since these systems can hibernate during the observation time. Therefore, thermal alterations should be considered as a secondary choice parameter after communication speed. However, the reliance on the CAN bus' differential pair may have been the reason for the choice.

Acknowledgments

This work was funded by the Spanish MINECO, under Grant No. AYA2014-60641-C2 coordinated project of the EPS-UAM and IAA-CSIC research groups. The authors declare no conflicts of interests.

References

1. D. H. Weinberg et al., "Observational probes of cosmic acceleration," *Phys. Rep.* **530**(2), 87–255 (2013).
2. M. Levi, "Dark Energy Spectroscopic Instrument," 2016, https://desi.lbl.gov (13 March 2019).
3. R. S. de Jong, "4-Metre Multi-Object Spectroscopic Telescope," 2018, https://www.4most.eu/cms/ (19 March 2019).
4. H. Murayama et al., "Prime focus spectrograph," 2018, https://pfs.ipmu.jp/ (19 March 2019).
5. C. Fisher et al., "Developing engineering model cobra fibre positioners for the subaru telescope's prime focus spectrometer," *Proc. SPIE* **9151**, 91511Y (2014).
6. P. R. Gillingham et al., "Echidna: a multifiber positioner for the subaru prime focus," *Proc. SPIE* **4008**, 1395–1403 (2000).
7. W. Saunders et al., "'MOHAWK: a 4000-fiber positioner for DESpec," *Proc. SPIE* **8446**, 84464W (2012).
8. A. Sheinis et al., "Advances in the echidna fiber-positioning technology," *Proc. SPIE* **9151**, 91511X (2014).
9. J. Gilbert and G. Dalton, "Echidna mark II: one giant leap for 'tilting spine' fibre positioning technology," *Proc. SPIE* **9912**, 991221 (2016).

10. M. Schubnell et al., "The DESI fiber positioner system," *Proc. SPIE* **9908**, 990892 (2016).
11. Y.-H. Zhao, "Large Sky Area Multi-Object Fiber Spectroscopic Telescope," 2012, http://www.lamost.org (19 March 2019).
12. X.-Q. Cui et al., "The large sky area multi-object fiber spectroscopic telescope (lamost)," *Res. Astron. Astrophys.* **12**(9), 1197–1242 (2012).
13. R. Li, C. Liu, and F. Luo, "A design for automotive can bus monitoring system," in *2008 IEEE Vehicle Power and Propul. Conf.*, pp. 1–5 (2008).
14. W. Lawrenz, *CAN System Engineering: From Theory to Practical Applications*, 2nd ed., Springer, Dordrecht, The Netherlands (2013).
15. A. Sheinis et al., "First light results from the high efficiency and resolution multi-element spectrograph at the Anglo-Australian telescope," *J. Astron. Telesc. Instrum. Syst.* **1**, 035002 (2015).
16. F. Lang-Bardl et al., "A fibre positioner solution for the 4MOST instrument," *Proc. SPIE* **8446**, 844661 (2012).
17. T. Li et al., "Design of multi-motor distributed control system for optical fibers positioning based on CAN bus," *Proc. SPIE* **9912**, 99125I (2016).
18. M. Schubnell et al., "Desi fiber positioner testing and performance," *Proc. SPIE* **10706**, 107062A (2018).
19. A. Castillo-Morales et al., "Megara control system," *Rev. Mex. Astron. Astrofis., Ser. Conf.* **42**, 123–123 (2013).
20. K. Kaci et al., "Communication architecture system for fiber positioning of DESI spectrograph," *Proc. SPIE* **9912**, 99121V (2016).
21. C. Fisher et al., "Cobra: a two-degree of freedom fiber optic positioning mechanism," in *2009 IEEE Aerosp. Conf.*, pp. 1–11 (2009).
22. J. K. Brzeski et al., "Echidna: the engineering challenges," *Proc. SPIE* **5492**, 1228–1242 (2004).
23. A. Pérez-Calpena et al., "MEGARA focal plane subsystems," *Proc. SPIE* **8446**, 84465E (2012).
24. A. G. de Paz et al., "MEGARA: a new generation optical spectrograph for GTC," *Proc. SPIE* **9147**, 91470O (2014).
25. L. Makarem et al., "Collision avoidance in next-generation fiber positioner robotic systems for large survey spectrographs," *Astron. Astrophys.* **566**, A84 (2014).
26. N. Fahim et al., "An 8-mm diameter fibre robot positioner for massive spectroscopy surveys," *Mon. Not. R. Astron. Soc.* **450**, 794–806 (2015).
27. H. Zimmermann, "OSI reference model - the ISO model of architecture for open systems interconnection," *IEEE Trans. Commun.* **28**, 425–432 (1980).
28. ISO, "Road vehicles– controller area network (CAN)– part 1: data link layer and physical signalling," ISO 8802-3, standard, International Organization for Standardization, Geneva, CH (2015).
29. P. R. Gray et al., *Analysis and Design of Analog Integrated Circuits*, Wiley, New York (2001).
30. CiA, "Industrial communications subsystem based on iso 11898 (CAN) for controller-device interfaces - part 4: CANOpen," Technical Report, UNE-EN 50325-4, European Commitee for Standarization (2002).
31. ISO, "Road vehicles– diagnostic communication over controller area network (DoCAN)– part 5: specification for an in-vehicle network connected to the diagnostic link connector," ISO/CD 15765-5, standard, International Organization for Standadization, Geneva, CH.
32. NXP Semiconductors, "I^{2}C-bus specification, version 6.0," Technical Report, UM10204, NXP (2014).
33. L. Bacciarelli et al., "Design, testing and prototyping of a software programmable I^{2}C/SPI IP on AMBA bus," in *PhD Res. Microelectron. and Electron.*, pp. 373–376 (2006).
34. F. Zhao et al., "Design of schematic mapping system based on I^{2}C and USB bus," in *2011 First Int. Conf. Instrum., Meas., Comput., Commun. and Control*, pp. 180–183 (2011).
35. H. Saadat, *Power System Analysis*, 3rd ed., PSA Publishing, Alexandria, Virginia (2010).
36. D. J. Schlegel et al., "BigBoss: the ground-based stage-IV BAO experiment submitted by LBNL and NOAO," Technical Report, id.471.06, Cornell University Library (2009).
37. D. Schlegel et al., "The BigBoss experiment," Technical Report, SLAC-R-994, Cornell University Library (2010).
38. Chip45 GmbH & Co., *Sammy-C21 Infosheet*, Chip45 GmbH & Co., Wettenberg, Germany (2017).
39. G. Coley, "BeagleBone Black System Reference Manual, rev A5.2,," R. P. J. Day, Ed., Texas Instruments, Dallas, Texas (2013).
40. Texas Instruments, *SN65HVD1040 Low-Power CAN Bus Transceiver With Bus Wakeup*, Texas Instruments, Dallas, Texas (2015).
41. Texas Instruments, *PCA9548A Low-Voltage 8-Channel I2C Switch With Reset*, Texas Instruments, Dallas, Texas (2015).

Sofía Martínez García is an assistant professor in the Department Electronic Technology and Communications at the University Autonoma of Madrid (Spain) since 2016. She received her BS and MSc degrees in telecommunications and electronics engineering in 2006 and 2009, respectively, and her PhD degree in 2014 from the University of Granada (Spain). Her current research interests include design of electronic instrumentation for communications and digital control of switching mode power supplies.

Fernando Lopez-Colino is an associate professor at EPS-UAM since 2015. He received his BS degree in computer science in 2005, and his MS and PhD degrees in computer science and telecommunications engineering in 2007 and 2009, respectively. His current research interests are focused on embedded communication protocols and their application to wireless sensor networks and the internet of things. He has participated in the development of fiber positioner robots for the MSE project.

Javier Garrido received his BSc degree in 1974, MSc degree in 1976, and PhD degree in 1984, all in physics (electronics), from UAM. He received his current position as a full professor in 2010. From his incorporation into the EPS, his research interests are related to HW/SW applications on embedded systems as platforms for wireless sensor nets (WSNs) or robotic sensor agents. He had participated in developing of BigBOSS, DESI, and MSE.

Guillermo Glez-de-Rivera is a telecommunication engineer from UPM (Madrid, Spain) in 1995 and he received his PhD degree in computer science and telecommunications from UAM (Madrid, Spain) in 2011. He was an assistant professor first and an associated professor at UAM now, since 1997. His research interests include industrial control, microprocessors-based systems, embedded systems, robotics and co-operative systems, low power designs, sensors, and mobile nodes in wireless sensor networks. He had participated in developing of BigBOSS, DESI, and MSE.

Justo Sánchez, currently a staffed electronic engineer at the IAA-CSIC, was previously a staff astronomer at the National Astronomical Observatory. He has been a member of the electronic development team of the GIADA instrument of the ROSETTA mission. He has developed and built spectrographs for astronomy, for study of SPRITES, and for studies of RAYS at high speed. He has also collaborated in the development of fiber positioner robots for MOS systems in several telescopes.

Francisco Prada is an astrophysicist (full professor CSIC) with research interests in the fields of cosmology and astroparticle physics. He has been using cosmological simulations, optical and gamma-ray telescopes and large 3-D maps of the universe to study structure formation and the dark components of the universe. He is a member of the MultiDark, MAGIC, SDSS, DESI, and Euclid collaborations and has supervised a dozen PhD theses.