

Proyecto de investigación IMA_X+/_{SCIP} como elemento motivador en el aprendizaje del alumno

Manuel Rodriguez Valido
Dpto. Ingeniería Industrial
Universidad de La Laguna
LaLaguna, España
mrvalido@ull.edu.es

Eduardo Magdaleno Castello
Dpto. Ingeniería Industrial
Universidad de La Laguna
LaLaguna, España
emagcas@ull.edu.es

David Hernández Expósito
Grupo de Física Solar
Instituto Astrofísica de Canarias
La Laguna, España
dhdez@iaa.es

Basilio Ruiz Cobos
Grupo de Física Solar
Instituto Astrofísica de Canarias
La Laguna, España
brc@iac.es

Maria Baleguer Jimenez
Grupo de Física Solar
Instituto Astrofísica de Andalucía
Granada, España
balaguer@iaa.es

Abstract— Creemos que un elemento importante para el alumno en su proceso de aprendizaje, es hacerlo participe en Proyectos de Investigación Científico-Tecnológicos Innovadores (PCTI). Esta participación permite motivar al alumno e involucrarse más en asignaturas. Concretamente, se le plantea implementar el enlace de comunicación de alta velocidad basada en el estándar CoaXPress, entre la electrónica de lectura de un sensor de imagen y el host o frame grabber. Para ello, tienen que integrar un módulo IP CoaXPress en el sistema empotrado basado en Microblaze que posee el Firmware (FW) del instrumento IMA_X+/_{SCIP}. Desde el punto de vista metodológico, los alumnos adquirirán una visión y habilidad de trabajar en un sistema complejo, la habilidad de trabajar en equipo y desarrollarán experiencia en la metodología de diseño basado en Vivado-SDK-FPGA.

Keywords CoaXPress, GenICam, FPGA, Vivado-SDK, VHDL, Verilog TFG, TFM, Comunicaciones de alta velocidad, sensores de imagen

I. INTRODUCTION

Este trabajo, se enmarca dentro del campo de la metodología de diseño digital en FPGA dadas en varias asignaturas del grado Ingeniería electrónica y automática y del master de Industriales de la Escuela de Ingeniería y Tecnología de la Universidad de La Laguna.

El PCTI donde se desarrolla esta propuesta se denomina IMA_X+/_{SCIP} de la misión espacial Sunrise (2002-2021) [1]. Básicamente Sunrise es un observatorio solar montado sobre una góndola-globo dedicada para la investigación de los procesos magnéticos y los flujos conectivos en la atmósfera del sol. Esta misión, aprobada por plan nacional como una estrategia para el desarrollo de un demostrador tecnológico para la misión espacial Solar Orbiter (SO) [2]. SO es un satélite científico de observación de la física solar desarrollado por un consorcio internacional liderado por España y Alemania. SO fue seleccionado por la ESA como primera misión de clase media dentro del programa

“Cosmic Vision 2015-2025 y ha sido lanzado el 10 de febrero de 2020 desde Cabo Cañaveral [3] y lleva 10 instrumentos de los cuales, concretamente destacamos el magnetógrafo solar Polarimetric and Helioseismic Imager (PHI) [4] en el cual hemos participado.

Cabe destacar que IMA_X ha sido lanzado dos veces e IMA_X +/_{SCIP} instrumento es una versión mejorada de sus antecesores.

Este trabajo se estructura de la siguiente forma, en primer lugar, describiremos el sistema IMA_X+/_{SCIP} donde se detallan los dos módulos principales de la arquitectura, GSENSE400BS y CxP Control interface. A continuación, pasaremos a describir el software de control del firmware implementado y sus características principales. Y por último pasaremos a presentar los resultados y conclusiones obtenidos en este trabajo.

II. DESCRIPCION DEL SISTEMA

A. Arquitectura de sensor IMA_X+/_{SCIP}

El instrumento o magnetógrafo tiene como parte principal una cámara de altas prestaciones científicas y elegidas para obtener un buen rendimiento en las medidas de los parámetros físicos que se requieren [5]. El diagrama de bloques de la cámara IMA_X+/_{SCIP} es mostrado en la figura 1. Todo el sistema ha sido diseñado e implementado en VHDL/Verilog en una FPGA Artix 7 (xc7a50tcs325-2) y con la herramienta Vivado-SDK ambos de la empresa internacional Xilinx.

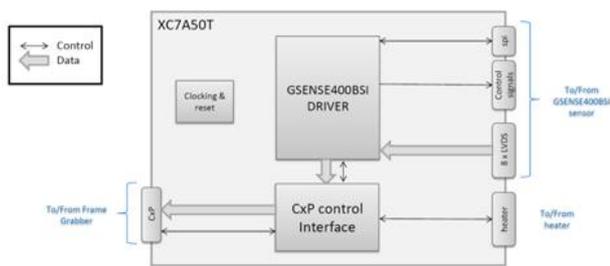


Fig. 1. Diagrama de bloques del FW IMAX+/SCIP

La arquitectura esta formada por dos grandes bloques, el driver del sensor GSENSE400BSI [6] y la interfaz de comunicaciones de alta velocidad CoaXPress [7].

El driver del sensor es el responsable tres grandes tareas, configuración del sensor, calibración de los canales de datos y controlar el proceso de adquisición de las imágenes. La tarea de configuración del sensor se hace mediante la interfaz de comunicaciones SPI. Mediante ella el driver accede a los registros del Sensor tanto para leer su estado como para cambiarlo.

El proceso de adquisición de datos de cada píxel del sensor (12 bits en modo estándar) hace uso de una interfaz de 8 canales series LVDS (Low Voltage Differential Signal). Estos datos tienen que ser previamente calibrados mediante un módulo en VHDL diseñado para tal fin. El proceso de calibración de cada canal tiene tres fases, y se basa principalmente sincronizar el reloj de lectura de los datos con los bits de cada píxel, las palabras de 12 bit de cada canal y los canales entre sí.

En cuanto al bloque CxP control interface o la interface CoaXPress, es realmente un sistema empotrado basado en Soft-Processor Microblaze. En dicho bloque hemos integrado un IP de comunicaciones bidireccional de alta velocidad para transmitir imágenes al Host y para recibir comandos desde el host a baja velocidad, todo por un cable coaxial. Este IP de alta/baja velocidad está basado en el estándar CoaXPress.

B. Gsense400BSI driver

Este subsistema fue adaptado y desarrollado a partir del diseño inicial que suministra el fabricante del sensor de imagen Gpixel. Este diseño de partida fue implementado, por esta empresa, en Verilog para una FPGA Spartan 6 con camera link interface.

A partir de las especificaciones dadas por el equipo de científicos y un análisis del diseño inicial, nosotros cambiamos dicha interface por la interface CoaXPress y modificamos distintos bloques tales como el secuenciador, bloque de calibración, el

bloque de control del sensor, los bloques de recepción y transmisión de imágenes. También, añadimos un módulo de reordenado de canales y un bloque de empaquetado de datos de tal forma que pudiéramos aprovechar el ancho de banda de transmisión (3,125 Gbits/s) de la interface CoaXPress.

La figura 2 muestra un diagrama de bloque donde se detalla el firmware del driver del sensor [8,9]. Las señales del sensor (flechas rojas) muestran la interfaz con el sensor y las verdes son las que se comunica con la interface de control CxP. Aunque no este pintado en la figura posee un sistema generador de reloj el cual permite que cada uno de los módulos que forma el sistema este perfectamente sincronizado.

El módulo de recepción de imagen tiene dos modos de funcionamiento, modo de adquisición y modo de calibración. Cuando está adquiriendo imágenes desde el sensor paraleliza a 12 bit cada uno de los 8 canales de datos serie LVDS, (*sensor_data_ser*). En modo calibración, es el encargado de determinar el offset existente en bits (con respecto al reloj de lectura), píxeles y canales para luego poder decodificar correctamente los datos cuando está en modo adquisición. Estas tareas están controladas por una máquina de 6 estados o módulo de control del sensor. Esta máquina de estado posee un registro de control para el gobierno de todo el sistema. En la tabla 1, podemos ver la asociación que tiene cada uno de estos 8 bit con el comando a realizar y la función software de sistema empotrado que se usa. De tal forma que con el solo uso o llamadas a las funciones que ahí se indica se puede controlar todos los procesos que se realizan sobre el sensor de GSENSE400BSI. Es decir, resetear el sensor, escribir o leer en los registros del sensor, adquirir imágenes e iniciar el proceso de calibración. También, hay un comando HW que se activa directamente desde el host y es usado como trigger HW para petición de un frame sin necesidad de generar el comando SW en la aplicación desarrollada (HW frame Request).

TABLE I. REGISTRO DE CONTROL Y SUS COMANDOS ASOCIADOS

Bit	Cmd	FSM State	SW function
0	SPI read	--	Spi_read()
1	SPI write	--	Spi_write
2	Reset	s RESET	resetSensor()
3	Training	S_TRAINI.	StarTrainig()

4	Frame Request	S_GRAB	requestFrame()
5	NOT USED	--	--
6	NOT USED	--	--
7	Go to IDLE	s_IDLE	--
trigger	HW-Fr-Req	s_GRAB_TG	Not SW function

El módulo de decodificación de imágenes tiene una interface Axi4 Lite para las comunicaciones entre el sistema el driver del sensor y el sistema empotrado que está en el módulo CxP. Además incluye tres bloques de memoria usados para almacenar los parámetros de configuración y las formas de ondas temporales que necesita el módulo *sequencer* en el proceso de adquisición. Los comandos enviados desde la aplicación software del sistema empotrado son almacenados en estos registros y atendiendo a sus contenidos los drivers operan o realiza las operaciones programadas.

El módulo *sequencer* haciendo uso de las formas de ondas almacenadas en el módulo codificador de imágenes direcciona cada pixel del sensor para ser leídos.

Y por último, el módulo SPI escribe y lee los datos de los registros de configuración del sensor.

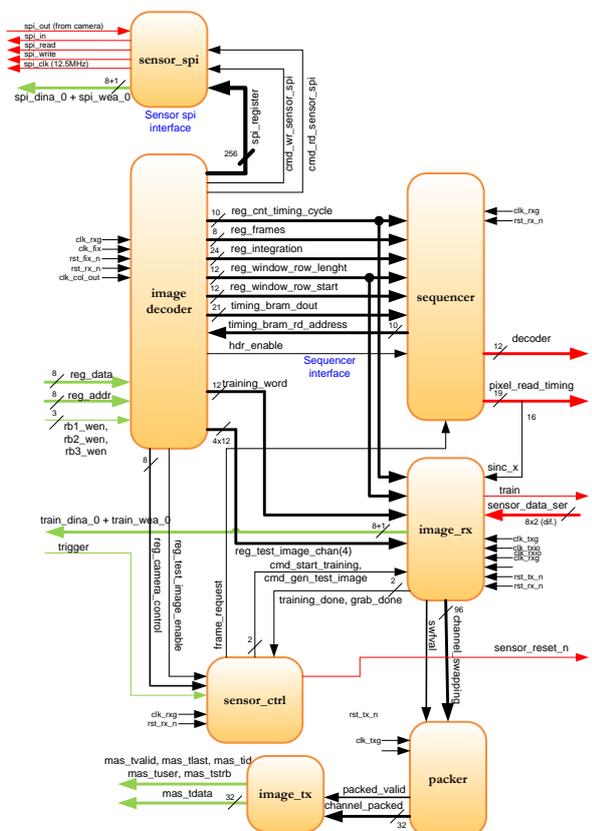


Fig. 2. Esquema del driver del sensor IMAx+/SCIP. GSENSE400BSI.

C. Interface de control CoaXPress

CoaXPress es una interface digital que fue desarrollada para transmisión de imágenes a alta velocidad y destinada principalmente a aplicaciones de visión por ordenador. La interface CoaXPress conecta dispositivos (cámaras típicamente) a un host (frame grabber). Esta combina la simplicidad del cable coaxial con el estado del arte de la tecnología de alta velocidad de datos series. Permitiendo, por el mismo cable, un data rate de hasta 6.25Gbps, enviar comandos para el control del dispositivo y la propia alimentación de toda la electrónica. Esta interface consiste de una conexión punto a punto entre el master y el dispositivo para formar un protocolo de enlace. Todos los datos son transmitidos usando paquetes codificados en 8B/10B [10]

En cada conexión el protocolo de enlace define un conjunto de canales lógicos mediante los cuales se hace la transferencia de triggers, entradas salidas de propósito general, datos de control y transmisión de streaming de alta velocidad, figura 3. Cada dispositivo es controlado por el Host vía registros y mediante el canal de control.

Fig. 3. Conexión de la interface CoaXPress y funciones transmitidas por los canales lógicos de la interfaz

La interface CoaXPress soporta el estándar GeniCam [11,12]. Normalmente esta compatibilidad con el estándar se hace mediante un fichero XML almacenado en el dispositivo. Este fichero es compartido con el Host en el momento de la conexión y de esta forma, suministra una descripción detallada, del dispositivo, compatible con la API (GenAPI) de GeniCam. Los registros de arranque almacenados en el dispositivo suministran al frame grabber una forma de acceder al fichero XML. Un aspecto importante de este fichero es que hay que usar el espacio de nombres dados por el estándar.

La interfaz de control CoaXpress en el proyecto IMAx+/SCIP es mostrada en el diagrama de bloque dado en la figura 4. Dicha interfaz de control esta basada en un diseño de bloques de Xilinx, implementado con la herramienta Vivado. La metodología de diseño con bloques, dada por esta herramienta, es una manera intuitiva y fácil de añadir IP Cores y/o módulos diseñados en VHDL, Verilog y Esquemático a un sistema empotrado.

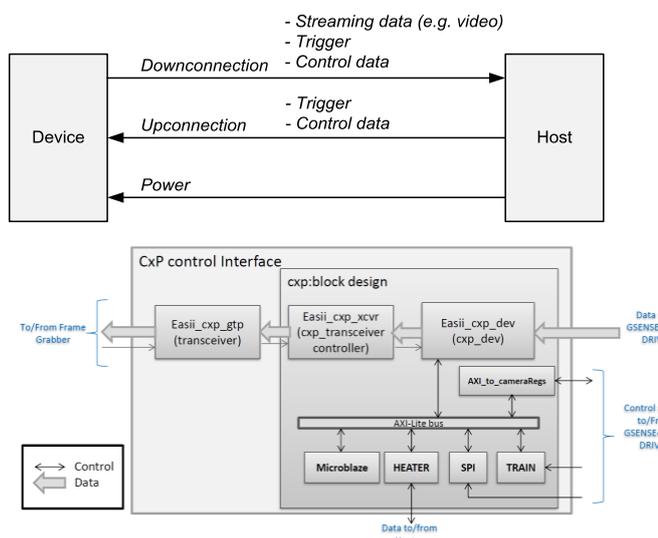


Fig. 4. Diagrama de bloques de la interfaz CoaXPress del instrumento IMaX+/SCIP

En la figura 4 podemos ver el sistema empotrado que controla tanto el driver del sensor (lado derecho) como la interfaz de comunicaciones CoaXPress (lado izquierdo).

También, se ven otros módulos (IP) necesarios para el instrumento tales como *Heater* (controlador de temperatura), *SPI*, y módulo *Train*, encargado de controlar el módulo de entrenamiento y estado definido en el driver GSENSE400BSI. Dicho sistema empotrado está basado en una arquitectura definida por un bus Axi4 Lite.

Axi4 Lite, permite, de una manera fácil acceder a todos los periféricos mediante los drivers de cada módulo y simples funciones software que están corriendo en el procesador Microblaze.

La interfaz de comunicaciones de alta velocidad también está controlada/conectada a este bus, la transferencia de datos del sensor es enviada directamente al host por la interfaz CoaXPress mediante el canal *I/O GTP transceiver* mediante el cual se puede alcanzar velocidades de transmisión de datos de hasta 6.6Gbps.

Como comentamos, este sistema empotrado, se basa en el Soft-Processor Microblaze. En él, está continuamente en ejecución una aplicación que está escuchando la interfaz CoaXPress para decodificar las ordenes/comandos procedentes del host o frame grabber y así facilitar las tareas de calibración, adquisición, configuración y transmisión el estado del instrumento. Todos los módulos de este sistema empotrado, están mapeados en el espacio de direcciones local, definido en el firmware del sensor, de tal forma que puedan ser comandados desde el host.

En cuanto al módulo IP CoaXPress *EASii_cxp_xcvr*. Es un módulo configurable diseñado por la empresa EASii-IC [13]. Este permite la transmisión de video procedentes de una o múltiples cámaras sobre un cable coaxial. Permite una velocidad de bajada (downlink) de hasta 6.25 Gbps (en nuestro caso 3.125 Gbps) y una velocidad de subida (uplink) para comunicación y control de 20.83 Mbps

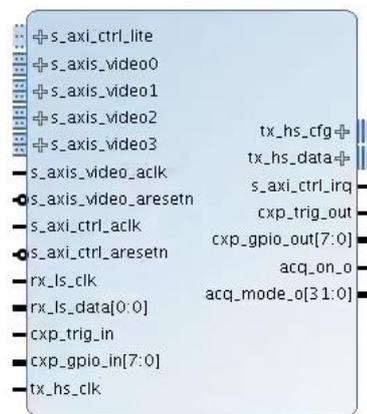


Fig. 5. EASii-IC CoaXPress IP interface

Este módulo IP ha sido adquirido con unas especificaciones concretas determinada para nuestra aplicación. La tabla 2 muestra los parámetros de las especificaciones del dispositivo. En cuanto a los parámetros de configuración, el dispositivo, dispone de una ventana de configuración de tal forma que se puede acceder desde el entorno *block design* en Vivado (figura 6). El resto de parámetros necesarios para la configuración de la interfaz CoaXPress están en el fichero XML del estándar GenICam. Estos parámetros no son editables y serán escritos por el procesador en el arranque. Es decir, antes de usar el IP, la aplicación software tiene que configurar los registros y las interrupciones. Una vez que el IP sea activado/arrancado. Este maneja todos los paquetes de control sobre la conexión hecha sobre la interface CoaXPress. El procesador no tiene actividad excepto cuando una petición del espacio del usuario es recibido. Esto significa que el host/frame Grabber pregunta por una dirección en hexadecimal por encima de 0x6000.

TABLE II. PARÁMETROS ESPECIFICOS DE CONFIGURACION DEL IP

Nombre de Parámetros	Valor
FPGA	xc7a50tcsq325-2
Single StreamID mode	True
AXI4-Stream TDATA width	32
AXI4-Streams interfaces number	1
AXI4-Lite SLVERR return	true

CoaXPress standard version	1.1
Maximal bitrate	3.125 Gbps
CoaXPress lane number	1
1,25 Gbps discovery activation	Yes
3,125 Gbps discovery activation	Yes

III. FIRMWARE. SOFTWARE DE CONTROL

El software que controla la cámara fue desarrollado en lenguaje **c** y mediante la herramienta SDK IDE de Xilinx. En general esta aplicación, tiene un funcionamiento o tareas tales como: inicialización de registros locales, y registros del sensor y establecer el enlace del sistema comunicaciones entre el frame grabber (host) y el dispositivo para la adquisición de datos (sensor). El software de control interactúa con el driver del sensor mediante tres bancos de registros, RB1, RB2, y RB3.

La figura 7 muestra el diagrama de flujo de esta aplicación software. En ella, y como etapa inicial de funcionamiento, la aplicación inicializa la interfaz CoaXPress e inicia el proceso de conexión con el frame grabber. Es esta fase, hay un proceso de descubrimiento entre el Host y el Dispositivo. En la cual se establece la velocidad de transmisión (3,125Gbps) y el fichero XML almacenado en el dispositivo es enviado al Host. Una vez establecido el enlace se inicializa los bancos de registros RB1, RB2 y RB3 y el sensor es configurado con los valores por defecto que se encuentran en el fichero.

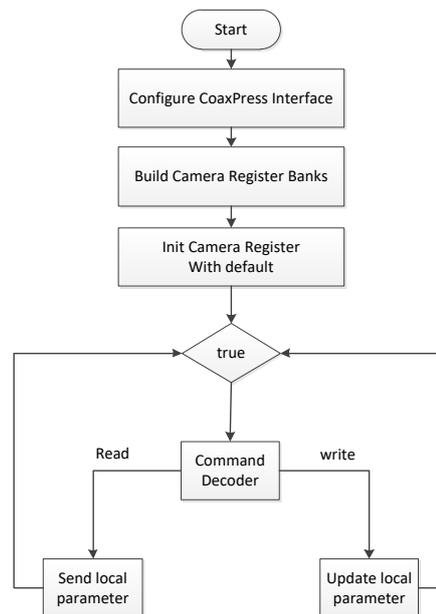


Fig. 7. Diagrama de flujo de la aplicación principal del procesador Microblaze

Una vez que estas etapas iniciales terminan, la aplicación software entra en un estado de decodificación de comandos. Es decir, cuando un comando (lectura o escritura) es detectado, el decodificador ejecuta dicha acción sobre los bancos de registros permitiendo esto interactuar con el DRIVER del sensor.

A. Vivado y diseño con bloques

Como comentamos anteriormente, el diseño presentado fue implementado en VHDL, Verilog, y en el caso del módulo IP de la interfaz CoaXPress, hemos usado uno adquirido a Easii-IC. Este módulo viene con su datasheet y un guía de configuración. Por un lado, tiene una interfaz de alta velocidad configurable mediante registros internos y por otro lado tiene una interfaz Axi4 lite de tal forma que puedas ser integrado fácilmente en una arquitectura basada en este bus (figura 4). Además, el fabricante nos proporciona un fichero de restricciones necesarias para el correcta integración y funcionamiento del mismo.

Vivado Design Suit tiene una forma fácil de integrar módulos y organizar un sistema empotrado. De forma automática, una vez que el módulo IP sea integrado, este lo sitúa en el espacio de memoria del sistema y crea las rutinas básicas (drivers) para acceder (lectura/escritura) al módulo y a todos los registros internos que este posea.

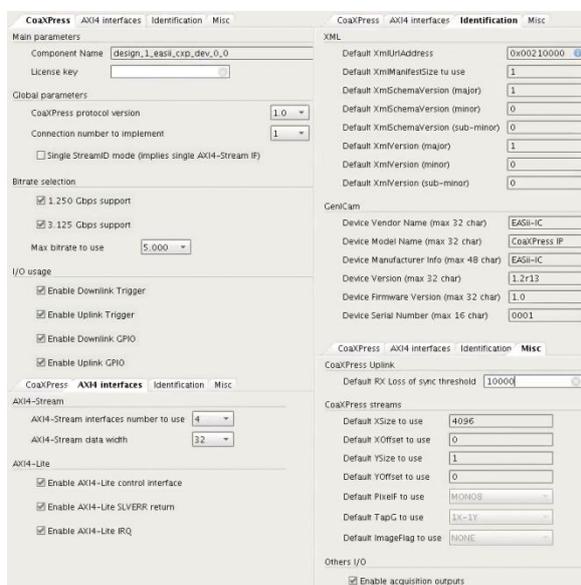


Fig. 6. Ventana de configuración del módulo IP CoaXPress de EASii-IV

RESULTADOS

Todo el sistema fue testeado satisfactoriamente. La cámara (figura 8) y su firmware fue conectado a un frame grabbe comercial con CoaXPress interface y compatible con GenICam, figura 9.

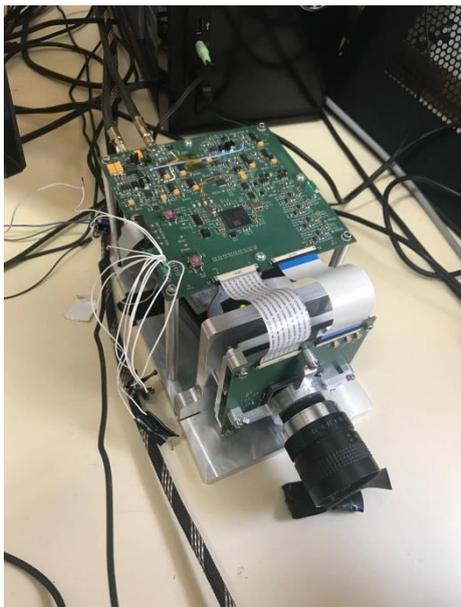


Fig. 8. Prototipo hardware de IMaX+/SCIP

Las pruebas fueron realizadas en el laboratorio, en las cuales se hicieron ajustes de todas las partes del proceso de adquisición. La figura 10 muestra una captura de una plantilla diseñada para este propósito.

El alumno que realizó este TFG, partió de conocimientos básicos ya adquiridos en metodología de diseño basada en FPGA-VHDL, sistemas empuados y sistemas de comunicaciones. Todas estas adquiridas en distintas materias del Grado en Ingeniería Electrónica y Automática Industrial. Desde este punto de vista, todo su esfuerzo del alumno, se centró en la integración del módulo IP CoaXPress de comunicaciones en el sistema empuado que le fue dado. Además, el alumno tuvo que implementar las funciones de decodificación de comandos en la aplicación software que corre sobre el procesador Microblaze.

Aunque el resto de módulos y arquitectura general del sistema empuado fue implementado por el equipo, el alumno, se enfrentó a una experiencia de metodología de diseño Hardware-Software basada en Vivado block design y SDK de Xilinx para integrar módulos en un sistema empuado sobre una FPGA.

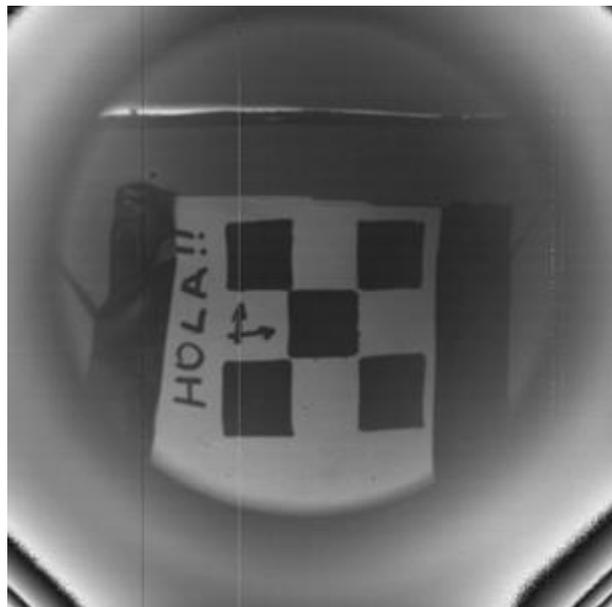


Fig. 9. 2048x2048 imagen de Test capturada por el sistema.

Además, adquirió experiencia práctica en el manejo de los drivers y software a bajo nivel para poder realizar operaciones de lectura y escritura en cada uno de los módulos/periféricos de la arquitectura propuesta.

En cuanto a la metodología para poder abordar el problema satisfactoriamente, el alumno siguió las siguientes fases de diseño.

- Estudio de los estándares CoaXPress y GenICam
- Estudio de las especificaciones o requerimientos científicos dados por el equipo de investigadores.
- Análisis del sistema HW/SW de partida proporcionado.
- Familiarizarse con la herramienta intuitiva Vivado block-design concretamente con la integración de un IP o módulo básico de los de librerías que proporciona Xilinx
- Integrar el módulo CoaXPress de EASii-IC y configurarlo según guía del usuario.
- Identificar y usar los drivers software que genera automáticamente la herramienta e integrarlos en el programa principal, con el objetivo de establecer una conexión entre el frame grabber y el sensor para la adquisición de imágenes.

A todo lo anterior, le podemos añadir que esta experiencia práctica, ha permitido al alumno enfrentarse y participar en un proyecto de investigación científico-tecnológicos innovador y multidisciplinar. Esta participación, en cierta

forma, ha obligado al alumno ser estricto con las tareas y plazos establecidos en el proyecto. Ya que las reuniones semanales de todo el equipo investigador obligaban/comprometían al alumno a seguir la disciplina del equipo. Poniendo esto en valor la importancia de estar coordinado y organizado con el grupo de investigación en todo momento.

Desde nuestro punto de vista, las tareas que resultaron más complejas para el estudiante de TFG, fue entender los conceptos y parámetros de configuración de los estándares CoaXPress y GenICam. Aunque estos elementos tecnológicos innovadores son complejos en sí, la dificultad encontrada por el alumno, básicamente fue debida, por un lado, a que son conceptos que no se trabajan en el grado, y por otra parte el estándar GenICam, que, aunque se encuentra en la frontera del diseño HW/SW se necesita conocimientos software para entender y usar dicho estándar.

Una vez entendido el funcionamiento y los conceptos asociados a estos. La integración, configuración y programación del sistema fue una tarea más llevadera y sencilla. Ya que prácticamente la herramienta Vivado-SDK realiza las tareas de conexión de periféricos y generación de drivers automáticamente.

ACKNOWLEDGMENT

Este trabajo fue financiado por el programa del plan nacional de investigación, RTI2018-096886-B-C53

REFERENCES

- [1] S. K. Solanki et al, "The Second Flight of the Sunrise Balloon-Borne Solar Orbiter: Overview of Instruments Updates, the Flight, the Data, and First Results", *Astrophysical Journal Supplement Series*, volumen 229, número 1, 2017.
- [2] ESA Science & Technology - Solar Orbiter, <https://sci.esa.int/web/solar-orbiter>
- [3] Cosmic Vision ESA program (2015-2015) <https://sci.esa.int/web/cosmic-vision/-/46510-cosmic-vision>
- [4] Achim Gandorfer, Sami K Solanki, Joachim Woch, Valentin Martínez Pillet, Alberto Álvarez Herrero and Thierry Appourchau, The Solar Orbiter Mission and its Polarimetric and Helioseismic Imager (SO/PHI). *Journal of Physics: Conference Series*, Volume 271, GONG-SoHO 24: A new era of seismology of the sun and solar-like stars 27 June – 2 July 2010.
- [5] IMAX+/SCIP Camera Requirements Document. Solar Group, IAA (Instituto Astrofísica de Andalucía).
- [6] 4 Megapixels Scientific CMOS Image Sensor. Data sheet GPIXEL400BSI.
- [7] JIIA CXP-001-2015, CoaXPress Standard, Japan Industrial Imaging Association Standard Ver 1., Revised on Dec 29, 2015.
- [8] M. Rodríguez, E. Magdaleno, "FPGA firmware description for IMaX+/SCIP", *Actas de las Jornadas Sarteco*, Cáceres, 2019.
- [9] E. Magdaleno et al., "Ordenamiento de canales del sensor GSENSE400 en modo STD para el instrumento IMaX+", *Avances en Arquitectura y Tecnología de Computadores. Actas de las Jornadas Sarteco 2019*, pp.626-631, Cáceres, 2019.
- [10] Byte oriented DC balanced 8B/10B partitioned block transmission code. US Patent 4486739.
- [11] GenICam™ standard from the European Machine Vision (EMVA) website (<http://www.emva.org>).
- [12] Generic Interface for Cameras standard, GenICam Package Version 2018.06
- [13] CoaXPress Device IP Specification. IC/130206. Hard Soft Interface Document (EASii IC).