



UNIVERSITAT DE  
BARCELONA

## **Inicialització amb Ultra96-V2**

**Nom de l'alumne: Héctor Rubio León**

**Grau: Enginyeria Electrònica de Telecomunicació**

**Nom de l'empresa o Institució on ha realitzat l'estada: Centre  
Tecnològic de Telecomunicacions de Catalunya (CTTC)**

**Nom del Tutor: Nikolaos Bartzoudis & José Rubio**

**Període de temps en que s'ha fet l'estada:**

**03/07/2024 - 16/08/2024**

**Data d'entrega de l'informe: 10/09/2024**

## Introducción

La institución a la que asistí para realizar las prácticas de la asignatura Prácticas en Empresa es el CTTC (Centre Tecnològic de Telecomunicacions de Catalunya, (<https://www.cttc.cat>), una organización privada sin ánimo de lucro financiada por la Generalitat de Catalunya.

Las actividades de investigación del CTTC, tanto fundamentales como aplicadas, se centran principalmente en tecnologías relacionadas con las capas físicas, en enlace de datos y en red de los sistemas de comunicación, y con la Geomática.

Las tareas de investigación y desarrollo en el CTTC están organizadas en nueve Unidades de Investigación (<https://www.cttc.cat/organization>). Las prácticas en empresa las he realizado en la Research Unit ADAPTative Processing Technologies (ADAPT: <https://www.cttc.cat/adapt/>).

La principal motivación de la Research Unit ADAPT es proporcionar soluciones innovadoras de procesamiento adaptativo de señales y datos para sistemas inteligentes y autónomos, por ejemplo, en las redes del operador. Sus líneas de investigación son:

- 1) Procesamiento ágil de señales y datos para tecnologías de comunicación inteligente 6G
- 2) Procesamiento asistido por IA para transmisores B5G y 6G energéticamente eficientes.

Las pruebas de concepto de los proyectos que desarrollan en ADAPT se suelen hacer sobre sistemas basados en FPGA (Field-Programmable Gate Array). Para estar al día con los nuevos dispositivos que se fabrican, cada vez que sale al mercado una nueva FPGA en ADAPT intentan adquirir un Evaluation Kit de esa FPGA. Los Evaluation Kits permiten un rápido diseño y desarrollo de proyectos destinados a pruebas de concepto. La mayoría de Evaluation Kits que se prueban en ADAPT son del fabricante AMD (antigua Xilinx).

La práctica en ADPT en el CTTC se ha realizado sobre la placa de AVNET Ultra96-V2 que es una placa de desarrollo AMD Xilinx Zynq UltraScale+™ MPSoC basada en Arm y en la especificación Linaro 96Boards Consumer Edition (CE). La principal motivación para probar esta placa de desarrollo por parte de ADAPT es que se trata de una versión 'small factor' de una placa de desarrollo que puede resultar interesante y adecuada según qué tipo de desarrollo se pretenda hacer.

## **Objetivos de la estancia**

Aprender el proceso a seguir para poner en uso una nueva placa, junto a su programación y testeo.

Conocer la metodología de trabajo en una empresa que realiza investigación que involucra desarrollos implementados en electrónica.

## **Detalle de las actividades llevadas a cabo**

Una de las primeras tareas era el comparar, tanto prestaciones como aplicaciones posibles, la FPGA con la que se iba a trabajar en el centro de investigación, la Ultra96-V2, con la utilizada en asignaturas de la carrera, la DE0-CV.

Se llevó a cabo un proceso de aprendizaje de la herramienta de diseño hardware Vivado de AMD, y se hizo una comparativa con Quartus (Intel), el software equivalente utilizado durante la carrera.

Se hizo una prueba inicial de la placa Ultra96-V2 utilizando el software de programación y depuración Vitis (AMD), para el cual también se hizo una etapa de aprendizaje.

Se exploraron y probaron aplicaciones de ejemplo ubicadas en repositorios de AMD, y Avnet (Hackster.io y element14).

Como actividad extra, se asistió a la tercera edición del "Workshop" realizado por el CTTC en Sitges el 05/07/2024 donde se hizo una exposición sobre las investigaciones que cada miembro estaba llevando a cabo. Entre ellas se encuentran: «Run-time context-driven reconfiguration of functions running in a FPGA SoC device» de José Rubio, «Micro-orchestration of RAN Functions Accelerated in FPGA SoC Devices» de Nikolaos Bartzoudis, entre muchas otras.

## Ultra96-V2 VS DE0-CV

En esta sección vamos a comparar las prestaciones de la placa ordenador utilizada en las prácticas del CTTC con la placa de desarrollo utilizada durante la carrera.

Ultra96-V2	DE0-CV
Placa ordenador	Placa desarrollo
Xilinx Zynq UltraScale+ MPSoC ZU3EG SBVA484	Cyclone V E 5CEBA4F23C7N
4 LEDs y 2 de radio controlables	10 LEDs controlables
Micro SD Card	Micro SD Card
Wi-Fi / Bluetooth (mediante UART)(ATWILC300-MR110CA)	No lo ofrece
No Displays 7-segmentos	6 Displays 7-segmentos
Ethernet mediante conectores o USB	No lo ofrece
1 botón + 1 de reset	4 botones + 1 de reset
2 interruptores para seleccionar Boot Mode	10 interruptores (switches)
Mini DisplayPort	Conector de vídeo (VGA) (Video Graphics Array)
Protocolo MIPI (Mobile Industry Processor Interface) para conectar cámaras	Puerto Serie (PS/2) para utilizar teclado y ratón a la vez
Memoria LPDDR4 de 2GB	Memoria SDRAM de 64 MB
UART mediante un header de 4 pines o utilizando el JTAG/UART Pod (accesorio) (hay que cambiar una resistencia de posición para utilizar esta función)	JTAG (UART) mediante USB y activando un switch (10)
I2C	No lo ofrece

SPI	SPI
No hace falta cristal externo (interno de 25 MHz) y genera 5 frecuencias (33.3 (para el procesador), 27, 26, 25 y 24 MHz)	Cristal de 50 MHz para 4 clocks
1 conexión USB 3.0 (4.8Gbps) para subida de datos, 2 de bajada USB 3.0 y 1 de bajada USB 2.0 (480Mbps)	1 puerto USB Blaster
Comercial e industrial	Comercial
78 I/O multiplexadas para periféricos, 22 I/O alta densidad, 31 alto rendimiento, 40 pines de baja velocidad y 60 de alta	484 pines
154.350 celdas lógicas	49.000 elementos lógicos
141.120 CLB FFs (Complex Logic Blocks Flip-Flops)	
70.560 CLB LUTs (Look Up Tables)	1 LUT con 8 entradas con la posibilidad de ocupar el 25 % de los ALMs
	18.480 ALMs (Adaptative Load Management)
7.6 Mb bloque RAM	3080 Kb bloque RAM
360 DSP	66 bloques DSP
3 CMTs (Compression Mount Technology)	0
2 System Monitor	0
	73.920 registros

*Tabla 1. Comparación recursos entre Ultra96-V2 y DE0-CV*

Nota: Los recuadros en blanco quiere decir que no se ha encontrado esa información en la documentación o no se proporciona información numérica respecto a esa característica.

### AMD Zynq™ UltraScale+™ EG VS Cyclone V

Como uno de los elementos más importantes de un sistema es su capacidad de procesamiento de información, al igual que se ha hecho con las FPGAs, se van a comparar las prestaciones de sus procesadores. En el caso de la Cyclone V se va a considerar el modelo con más prestaciones, catalogado como HPS, porque si no, no se podría realizar esta comparación, ya que directamente no tiene procesador.

AMD Zynq™ UltraScale+™	Cyclone V HPS
4 procesadores Arm Cortex-A53 y 2 Arm Cortex-R5F	Procesador dual-core Arm Cortex-A9 MPCore
GPU (Arm Mali-400 MP2 de 667 MHz)	No lo ofrece
252 GPIOs	224 GPIOs
9.4 Mb RAM	
<b>Cortex-A53</b>	<b>Cortex-A9</b>
APU (Application Processing Unit)	APU
Arquitectura de 64 bits (Arm v8-A)	Arquitectura de 32 bits (Arm v7-A)
4 SP / 2 DP FLOPs (Floating-point operations per second)	2.5 DIMPS/MHz (Los DIMPS son cálculo de enteros, no de coma flotante) por procesador (CPU)
1.5 GHz de CPU	1 GHz de CPU
32KB de caché L1 (la de datos separada de la de instrucciones, las 2 de 32KB)	32KB de caché L1 (la de datos separada de la de instrucciones, las 2 de 32KB)
1MB de caché L2	256KB de caché L2

Incluye NEON (SIMD) con instrucciones de coma flotante de precisión simple y doble	Incluye NEON (SIMD (Single Instruction Multiple Data) y unidad de procesado de datos en paralelo) y un vector de coma flotante (VFP) como coprocesadores
8 etapas dual-issue pipeline	8 etapas de pipeline
Por cada ciclo de reloj 2 instrucciones avanzan de etapa (dual-issue)	2 instrucciones por ciclo de reloj
MMU (Memory Management Unit) por procesador	MMU por procesador
<b>Cortex-R5F</b>	
RPU (Real-Time Processing Unit)	
Arquitectura de 32 bits (Armv7-A)	
600 MHz de CPU	
32KB de caché L1 (la de datos separada de la de instrucciones, las 2 de 32KB)	
128KB TCM (Tightly Coupled Memory)	
MPU (Memory Protection Unit) por procesador	

*Tabla 2. Comparación prestaciones procesadores*

Hay que mencionar que hay infinidad de información respecto a las prestaciones de los dispositivos electrónicos, aquí se ha resumido brevemente varias de ellas que han sido innovadoras para mí, características conocidas ya que se ha visto durante la carrera o que realmente son importantes. Prestaciones que no se ha incluido son como por ejemplo la medida de las instrucciones, de cuántos caminos son las distintas cachés, las interrupciones, etc.

## **Vivado VS Quartus**

La interfaz de Vivado gana respecto a Quartus ya que es un poco más compacta y permite encontrar lo que necesitas más rápidamente.

En el momento de estar escribiendo código, Vivado resalta sintaxis que detecta que es incorrecta, mientras que Quartus no tiene esa opción. Para ello, habría que descargar un Notepad con esa función como Sublime Text.

Otra de las opciones de las que dispone Vivado es el completar una línea de código previendo lo que se va a escribir, esto puede gustar más o menos, pero se puede desactivar, aunque hace más cómodo el proceso de escribir. Quartus no dispone de esta posibilidad y si se quiere se ha de buscar un Notepad con esta opción, Sublime Text dispone de esto.

Algo bueno que tiene Quartus son sus diagramas de máquinas de estado, ya que son bastante visuales.

Y una de las mayores ventajas de Vivado respecto a Quartus es sin lugar a duda la posibilidad de hacer una simulación en el propio programa en vez de tener que hacer la simulación con un programa externo como por ejemplo ModelSim, en el caso de Quartus.

Otra gran característica de Vivado es que puedes ver cuánto ocupa tu diseño en el propio dispositivo, como está enrutado, los caminos "worst case" y sus tiempos, etc. Quartus dispone de análisis temporales, pero no de una forma tan visual.

## **Vitis VS Code Composer**

Como último software comparable con uno utilizado en la carrera, encontramos la herramienta de programación y depuración.

A rasgos generales, Vitis y Code Composer tienen grandes similitudes. La interfaz es muy parecida, la capacidad de debugging, las mismas posibilidades de utilizar distintos lenguajes, ...



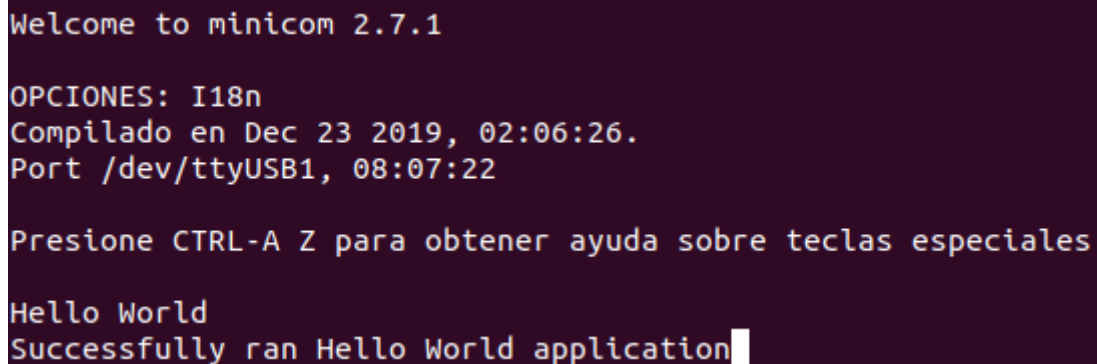
## Outputs de programas realizados

Durante la etapa de aprendizaje se fueron realizando varios programas para ir aprendiendo y acostumbrándose a los entornos de trabajo. El sistema operativo que se usó fue Linux en la versión de Ubuntu 22.04.4 LTS.

Lo primero que se llevó a cabo fue el típico programa de «Hello World» que mostraba el mensaje en cuestión por el puerto serie. Este programa sirve para probar que la placa está bien configurada y que casi todas las interfaces funcionan.

La herramienta Vitis permite conectarse mediante JTAG con la Ultra96-V2 y cargar el código desarrollado y probarlo.

En el primer programa se usó uno de los procesadores Arm Cortex-A53 utilizando un template que proporciona Vitis con la aplicación de «Hello World», que escribe en el puerto serie el mensaje. Para poder visualizar dicho mensaje se hizo uso de la aplicación ‘minicom’, que es un emulador de puerto serie para Linux.



```
Welcome to minicom 2.7.1

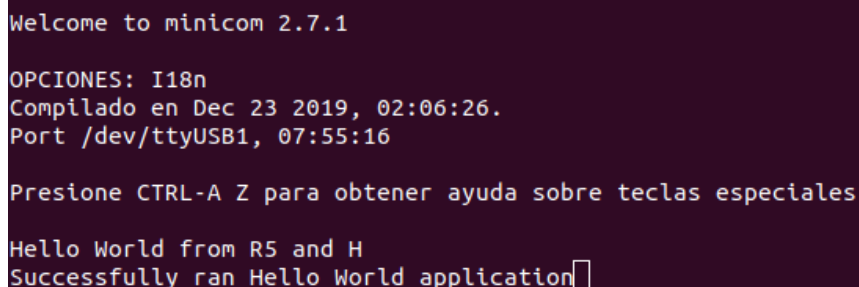
OPCIONES: I18n
Compilado en Dec 23 2019, 02:06:26.
Port /dev/ttyUSB1, 08:07:22

Presione CTRL-A Z para obtener ayuda sobre teclas especiales

Hello World
Successfully ran Hello World application
```

*Ilustración 1. Mensaje del programa Hello World utilizando el A53*

Y luego se hizo lo mismo usando uno de los procesadores Arm Cortex-R5F cambiando el código para que también escribiese «and H»:



```
Welcome to minicom 2.7.1

OPCIONES: I18n
Compilado en Dec 23 2019, 02:06:26.
Port /dev/ttyUSB1, 07:55:16

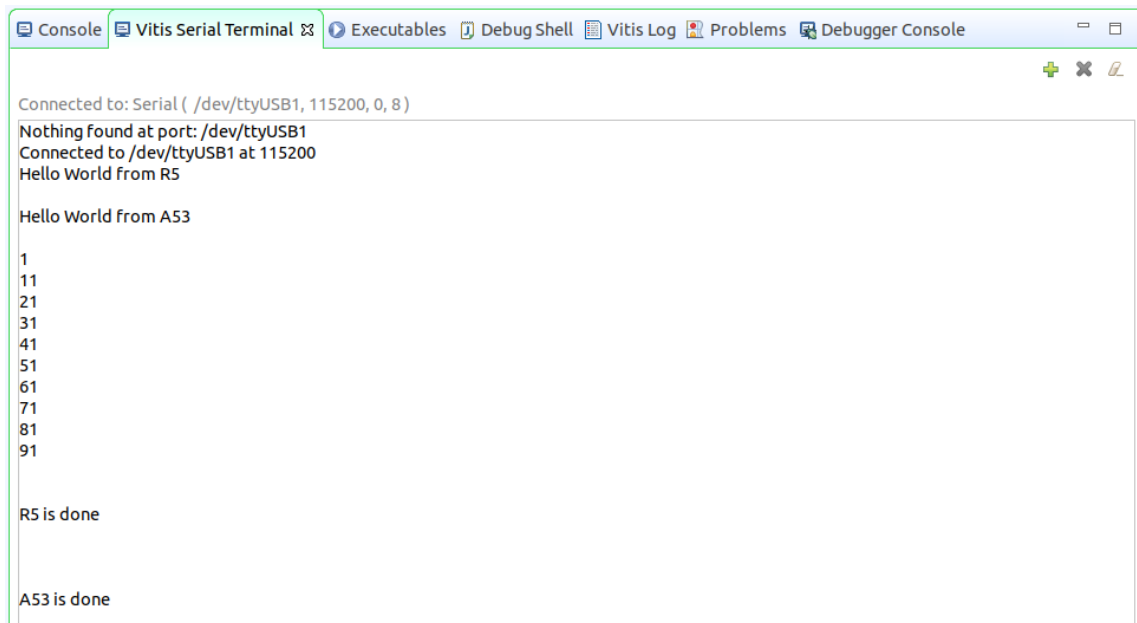
Presione CTRL-A Z para obtener ayuda sobre teclas especiales

Hello World from R5 and H
Successfully ran Hello World application
```

*Ilustración 2. Mensaje del programa Hello World modificado utilizando el R5*

Además de disponer de plantillas para un programa donde simplemente se hace un print por pantalla, también hay otros más complejos. Probamos un test de memoria y un test de periféricos.

Hicimos un mini proyecto para utilizar a la vez el A53 y el R5. En lugar de utilizar la aplicación minicom para visualizar la salida serie de los programas se uso el terminal serie de Vitis:



```
Connected to: Serial ( /dev/ttyUSB1, 115200, 0, 8 )
Nothing found at port: /dev/ttyUSB1
Connected to /dev/ttyUSB1 at 115200
Hello World from R5

Hello World from A53

1
11
21
31
41
51
61
71
81
91

R5 is done

A53 is done
```

*Ilustración 3. Mensaje por el terminal de Vitis del programa que utiliza simultáneamente el A53 y el R5*

Aquí se crearon dos dominios, uno para el A53 y otro para el R5, de modo que ambas aplicaciones se iban a ejecutar simultáneamente apuntando a la misma dirección de memoria cuyo valor inicial es cero.

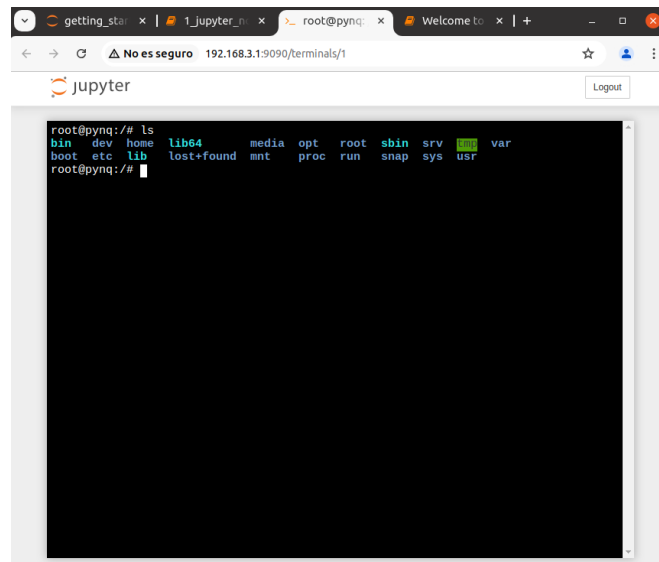
La secuencia del programa es la siguiente:

- 1) R5: suma 1 al valor en memoria
- 2) A53: hace print del valor en memoria y le suma 9

Esto se repite 10 veces. Todo esto se muestra en el propio terminal serie de Vitis.

Accediendo a la Ultra96-V2 mediante Ethernet:

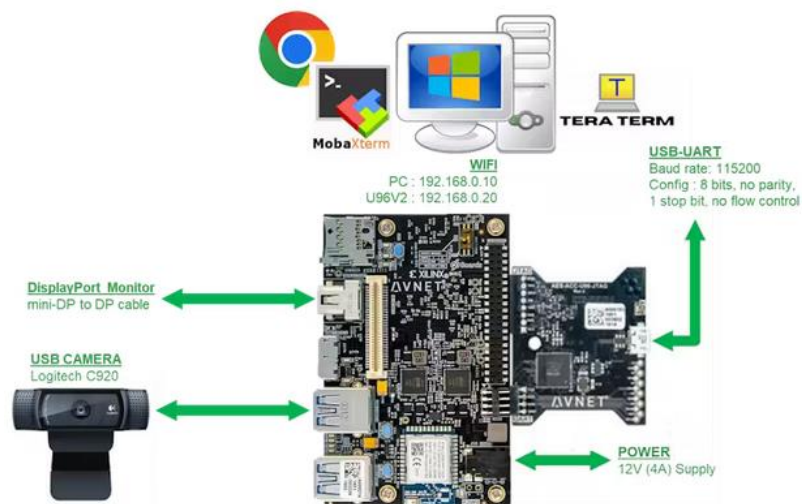
La Ultra96-V2 tiene un puerto USB 3.0 Type Micro-B upstream que sirve para conectar la placa a un navegador en un PC host. Se usó Firefox. Avnet proporciona una imagen para flashear en una tarjeta MicroSD y poder usar ese puerto. Como se puede apreciar en la fotografía, la IP asignada a la Ultra es 192.168.3.1. No solo eso, sino que también se podían ejecutar scripts de Python en la propia placa.



*Ilustración 4. Utilizando Jupyter desde Ultra96-V2*

Una vez que se comprueba que la placa y sus interfaces funcionan correctamente, se pasa a realizar proyectos más complejos.

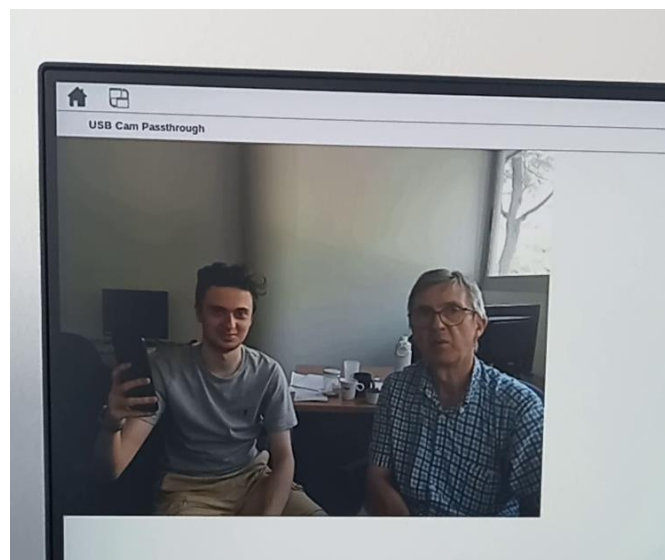
Es habitual que los fabricantes de placas de desarrollo proporcionen repositorios con proyectos de cierto interés y complejidad con instrucciones detalladas de cómo replicarlos. En concreto para esta práctica el siguiente proyecto que se desarrolló fue “*Ultra96-V2 - Building the foundational designs*” (<https://www.hackster.io/AlbertaBeef/ultra96-v2-building-the-foundational-designs-e4315f#overview>).



*Ilustración 5. Hardware setup*

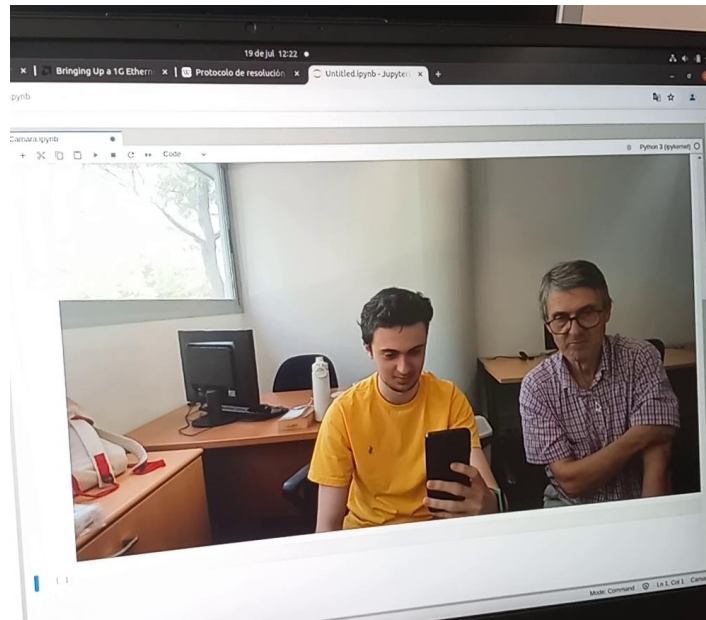
El resultado final pretendido es mostrar la imagen capturada con una cámara USB en un monitor conectado al puerto DisplayPort de la placa. Se usan las herramientas Vivado, Petalinux. Esta última herramienta de AMD ofrece todo lo necesario para personalizar, crear e implementar soluciones Linux integradas en los sistemas de procesamiento embebido AMD.

Todos los comandos Linux utilizados en este proyecto se incluyeron en un script Bash ('Ultra96-V2-Building\_the\_foundational\_designs.sh').



*Ilustración 6. Captura de la salida de vídeo de la cámara USB en el monitor conectado a la Ultra*

Modificando el proyecto PetaLinux se incluía la posibilidad de poder ejecutar en Python scripts en Jupyter Notebook. Con la nueva imagen flasheada en la tarjeta MicroSD se abría un terminal Jupyter Notebook y se ejecutaba un script Python que mostraba la salida de la cámara USB:



*Ilustración 7. Captura de la salida de vídeo de la cámara con script de Python*

Usando como base este proyecto se desarrolló uno nuevo con el objetivo de conseguir utilizar reconocimiento facial usando como base la imagen de la cámara USB. Este proyecto es:

*“Ultra96-V2 - Adding support for Vitis-AI 3.0”* (<https://www.hackster.io/AlbertaBeef/ultra96-v2-adding-support-for-vitis-ai-3-0-704799>).

Se creó una plataforma (entorno para crear software integrado y aplicaciones aceleradas) en Vitis y un overlay (librería de hardware para diseños de FPGA programables y configurables que extienden la aplicación del usuario desde el procesador hasta la lógica programable. Se pueden utilizar para acelerar una aplicación de software o para personalizar la plataforma de hardware para una aplicación en particular) para usar Vitis AI, que es el entorno de desarrollo de Xilinx para la inferencia de AI (Artificial Intelligence). Vitis AI proporciona la IP DPU (“Deep Learning Processing Unit”) y las herramientas necesarias para implementar redes neuronales estándar en plataformas de hardware de AMD.

Con la DPU se utiliza un modelo particular según el tipo de inferencia que se quiera hacer. En nuestro caso se usó el modelo “face\_mask\_detection\_pt”.

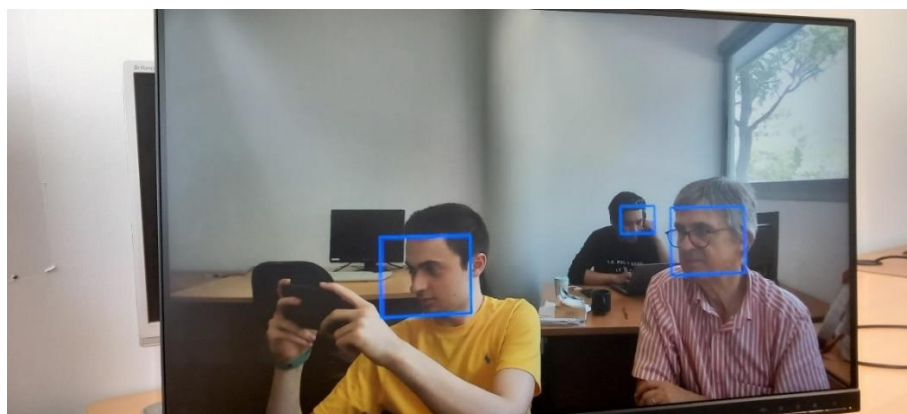
Como en el proyecto anterior, todos los comandos Linux utilizados en este proyecto se incluyeron en un script Bash ('Ultra96-V2-Adding\_support\_for\_Vitis-AI\_3.0.sh').



*Ilustración 8. Captura de la salida de vídeo de la cámara por SSH (Secure Shell) utilizando reconocimiento facial*

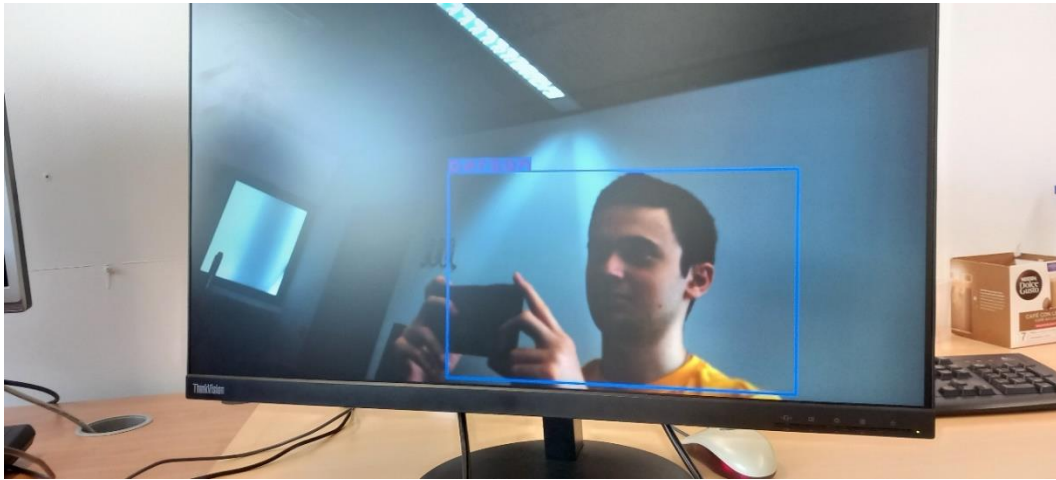
El desarrollo de los dos proyectos anteriores está basado en las plataformas que proporciona AMD para la placa de desarrollo Kria KV260. Como esta placa estaba disponible en el CTTC, se decidió probar un proyecto de reconocimiento de varios tipos en dicha placa. Este proyecto es: "Creating a Custom Kria App" (<https://community.element14.com/technologies/fpga-group/b/blog/posts/kv260-vvas-sms-2021-1-blog>).

Primero se utilizó la cámara USB utilizando un modelo para reconocimiento facial en la DPU, al igual que se hizo con la Ultra:



*Ilustración 9. Captura de la salida de vídeo de la cámara USB por monitor utilizando un modelo de reconocimiento facial en la DPU de la KV260*

También se probó un modelo de reconocimiento de cuerpo, aunque también distingue coches, autobuses, motos, etc. Pero esta vez es utilizando el protocolo de vídeo MIPI con una minicámara que dispone la KV260, en vez de la cámara USB.



*Ilustración 10. Captura de la salida de vídeo de la minicámara por monitor utilizando un modelo de reconocimiento de objetos*

## **Problemáticas**

Durante el transcurso de la estancia han surgido algunos problemas:

- La especificación del HW necesario. El cable para conectar el monitor es Mini DisplayPort, en vez de DisplayPort como suele ser común. Cualquier monitor que use DisplayPort viene con un cable normal. En cambio, con esta placa el cable no viene incluido lo cual implica una compra extra, y además no vale cualquier cable, tiene que ser activo.
- La placa a pesar de tener un disipador pasivo se calienta bastante. Se tuvo que adquirir un ventilador con conexión USB porque la placa no incluye el típico conector de dos pines para estos casos.
- Han hecho lo que se denomina un SFF (Small Form Factor). Provocando que el USB 3.0 Micro-B esté muy pegado al otro USB 3.0 A y sea difícil utilizar ambos a la vez. Además el número de puertos USB disponibles no permite utilizar teclado, ratón y cámara USB simultáneamente. Se hubo de utilizar un hub para poder conectar todos los periféricos a la vez. En la placa KV260 los puertos USB son dobles, apilados verticalmente.



- Pero sobre todo el mayor problema y fastidio ha sido que el módulo WiFi estuviese defectuoso. Para poder conectar la placa a la red hubo que utilizar un adaptador de red USB Ethernet. A continuación, se adjunta una imagen con la demostración del código de fecha y un recorte de la reseña de Avnet donde se menciona este incidente:



**Subject:** Non-Functional WiFi on AES-ULTRA96-V2-G boards.

**Products Affected:** This Errata affects only those AES-ULTRA96-V2-G boards built between December 2022 and May of 2023.

AES-ULTRA96-V2-G

**Description:**

The affected boards were built and shipped with an incompatible component installed at ref. Y1 which controls the timing on the WiFi module.

**Product Identification:**

Boards built between workweek 45 of 2022 and workweek 23 of 2023 are all affected by this errata. To determine the manufacturer date code of your board, locate the product sticker affixed to your board, and read the last 4 digits. Note that the product sticker may be located under the heatsink and require that to be removed to expose it.



YYWW (17<sup>th</sup> week of 2019)

*Ilustración 11. Izquierda: Código de fecha. Derecha: Reseña sobre el módulo WiFi*

## Conclusiones generales sobre el trabajo realizado

El trabajo realizado en el CTTC por mi parte ha sido satisfactorio y provechoso.

Con respecto al entorno de diseño de Vivado y Vitis ha sido bastante fácil la adaptación y llevar a cabo todo lo previsto con estas herramientas ha sido sencillo porque son bastante intuitivas y potentes.

Se ha logrado cumplir todos los objetivos con éxito en la franja de tiempo disponible para realizar las prácticas.

El aprendizaje conseguido ha sido bastante bueno desde mi punto de vista teniendo en cuenta las pocas horas dedicadas a las prácticas.



## Conclusiones personales

Una conclusión personal de mi estancia en el CTTC ha sido comprobar como lo aprendido en la carrera tiene una aplicación en el mercado laboral, no solo la parte teórica, los conceptos, fundamentos o leyes matemáticas, sino también la parte práctica, como es el montar, programar o soldar.

Otra conclusión a destacar es la dificultad de la investigación en términos generales, es relativamente sencillo replicar algo si tienes la receta, pero de ahí a tener que idear algo nuevo o mejorar algo ya creado, no es una tarea que todo el mundo pueda hacer con facilidad.

Como conclusión final, puedo decir que todavía me falta mucho por aprender, y eso es una motivación para mí.

## Lecciones aprendidas

Con todo el trabajo que se ha llevado a cabo durante aproximadamente un mes y medio se han aprendido muchas lecciones.

- Utilización del SSH (Secure Shell o Secure Socket Shell) para poder acceder a la Ultra96-V2 utilizando el propio PC
- Utilización de emuladores de puerto serie Linux para poder depurar las aplicaciones en la placa.
- Librería GStreamer, para generar aplicaciones para la transmisión de audio/video hasta el procesamiento complejo de audio (mezcla) y video (edición no lineal).
- Protocolo de vídeo DisplayPort
- DPU (Deep Learning Processor Unit)
- Docker
- Uso de imágenes MicroSD para utilizar la placa
- Comandos en Linux
- Petalinux
- Finalmente y posiblemente el aprendizaje más importante de todos, el buscarse la vida, ya que en reiteradas ocasiones al seguir un tutorial o un proyecto lo que sucedía es que se presentaban errores que no se comentaban o ni siquiera se mencionaban y era momento de ponerse a investigar en los ficheros en que habían problemas (con suerte, si se mencionaba el tipo de error y en qué fichero aparecía) o buscar por Internet a ver si a más usuarios les había sucedido lo mismo y los responsables en cuestión habían dado respuesta (pocas veces).

## **Repositorio GitHub**

Por indicación del turo de las prácticas, para finalizar el trabajo se recopiló y almacenó todo lo realizado durante la estancia en el CTTC en un repositorio GitHub para que así quede constancia y sea de fácil acceso para cualquier interesado que quisiera replicarlo.

El link es el siguiente: <https://github.com/hrl-112/CTTC-Internship>

## **Agradecimientos**

Para finalizar mi explicación sobre la estancia en el CTTC quería agradecer en primer lugar a la propia institución por acogerme con los brazos abiertos, ya que ha sido una estancia muy gratificante e importante para mí. Y más concretamente a aquellas personas con las que he estado en contacto directo durante estos meses. En primer lugar, a Laura Casaus, por ser la persona que se encargó de todo el proceso de ponerse en contacto conmigo y la UB de una forma fantástica y eficiente para que pudiese hacer las prácticas, además que el primer día que llegue me enseñó todo el centro. También a Mario Isaac, por ser aquel conserje que todo el mundo querría tener al lado, muy servicial a la par que gracioso, además de ser aquella persona que acudía en tu ayuda cuando te perdías por los pasillos. Finalmente, a Nikolaos Bartzoudis y a José Rubio, mis tutores, que no dudaban ningún momento en echarme una mano si tenía algún problema o no sabía/entendía qué tenía que hacer. Además de proporcionarme todos los materiales, dispositivos, programas, etc. necesarios y de mejor calidad para poder realizar las prácticas sin ningún problema y de la forma más cómoda posible.

## Referencias bibliográficas

CTTC. <https://www.cttc.cat/>

PMT-CENTRO TECNOLÓGICO DE TELECOMUNICACIONES DE CATALUÑA (CTTC). Ajuntament de Castelldefels. <https://www.castelldefels.org/es/ciudad/guia-de-la-ciudad/directorio-equipamientos-y-entidades/22587>

Ultra96-V2. AVNET. <https://www.avnet.com/wps/portal/us/products/avnet-boards/avnet-board-families/ultra96-v2/>

Xilinx Zynq UltraScale+MPSoC. AMD. <https://www.amd.com/en/products/adaptive-socs-and-fpgas/soc/zynq-ultrascale-plus-mpsoc.html#tabs-0e7c9e4014-item-89fcd7ef51-tab>

Clock System Ultra96-V2. 5P49V6975. Renesas. <https://www.renesas.com/us/en/products/clocks-timing/clock-generation/programmable-clocks/5p49v6975-versaclock-6e-programmable-clock-generator-integrated-crystal>

Zynq® UltraScale+™ MPSoC Data Sheet: Overview. AMD. <https://docs.amd.com/v/u/en-US/ds891-zynq-ultrascale-plus-overview>

Zynq 7000 SoC Technical Reference Manual (UG585). AMD. <https://docs.amd.com/r/en-US/ug585-zynq-7000-SoC-TRM/Dual/Single-Arm-Cortex-A9-MPCore-CPU-with-Arm-v7>

Zynq UltraScale+ Device Technical Reference Manual (UG1085). AMD. <https://docs.amd.com/r/en-US/ug1085-zynq-ultrascale-trm/Zynq-UltraScale-Device-Technical-Reference-Manual>

DE0-CV Board. ter asIC. <https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=163&No=921&PartNo=4>

Cyclone V. Intel. <https://cdrdv2.intel.com/v1/dl/getContent/666692?explicitVersion=true>

5CEBA4F23C7N Altera. Mouser Electronics. <https://www.mouser.es/ProductDetail/Altera/5CEBA4F23C7N?qs=w%252BhYR4jzwbUqcM9aDLI%2FA%3D%3D>

AM437x and AMIC120 ARM® Cortex™-A9 Processors Technical Reference Manual. Texas Instruments. [https://www.ti.com/lit/ug/spruhl7i/spruhl7i.pdf?ts=1720064422104&ref\\_url=https%253A%252F%252Fwww.myirtech.com%252Flist.asp%253Fid%253D510](https://www.ti.com/lit/ug/spruhl7i/spruhl7i.pdf?ts=1720064422104&ref_url=https%253A%252F%252Fwww.myirtech.com%252Flist.asp%253Fid%253D510)

ARM Cortex-A8: Whats the difference between VFP and NEON. Stack Overflow. <https://stackoverflow.com/questions/4097034/arm-cortex-a8-whats-the-difference-between-vfp-and-neon>

Cortex-A53. ARM Developer. <https://developer.arm.com/Processors/Cortex-A53>

What exactly is a dual-issue processor? Stack Overflow.

<https://stackoverflow.com/questions/8014739/what-exactly-is-a-dual-issue-processor>

Getting Started with the Vivado IDE. AMD Xilinx.

<https://www.xilinx.com/video/hardware/getting-started-with-the-vivado-ide.html>

Vivado Design Suite User Guide: Getting Started. AMD Xilinx.

<https://docs.amd.com/v/u/2018.2-English/ug910-vivado-getting-started>

Xilinx Vivado Tutorial: 1 (Basic Flow). VLSI Techno.

<https://www.youtube.com/watch?v=nBdXxRwb-Pg>

How to use vivado for Beginners | Verilog code | Testbench | Schematic View. Anand Raj.

[https://www.youtube.com/watch?v=onMmG\\_U4SVo&t=4s](https://www.youtube.com/watch?v=onMmG_U4SVo&t=4s)

Ultra96-V2 Vivado 2020.2 Basic Hardware Platform. Bryan H Fletcher.

<https://www.hackster.io/BryanF/ultra96-v2-vivado-2020-2-basic-hardware-platform-6b32b8>

Ultra96-V2 Vitis 2020.2 Hello World from ARM A53. Bryan H Fletcher.

<https://www.hackster.io/BryanF/ultra96-v2-vitis-2020-2-hello-world-from-arm-a53-2d952a>

Ultra96-V2 Vitis 2020.2 Hello World from ARM R5. Bryan H Fletcher.

<https://www.hackster.io/BryanF/ultra96-v2-vitis-2020-2-hello-world-from-arm-r5-4548d5>

Ultra96-V2 Vitis 2020.2 Test Applications. Bryan H Fletcher.

<https://www.hackster.io/BryanF/ultra96-v2-vitis-2020-2-test-applications-9afcbe>

OpenAMP in Xilinx MPSoC FPGA- Running Petalinux & Baremetal. LogicTronix [FPGA Design + Machine Learning Company]. <https://www.hackster.io/LogicTronix/openamp-in-xilinx-mpsoc-fpga-running-petalinux-baremetal-e025ca>

Vitis-AI 3.0 designs for Ultra96-V2 and ZUBoard. Mario Bergeron.

<https://community.element14.com/technologies/fpga-group/b/blog/posts/vitis-ai-3-0-designs-for-ultra96-v2-and-zuboard>

Ultra96v2, Zynq UltraScale+ MPSoC - Flashing LEDs. High Level Synthesis.

<https://www.youtube.com/watch?v=08f23pOl-Tk>

PetaLinux 101 - Getting Started Quickly. Adaptive Computing Developer.

<https://www.youtube.com/watch?v=k03r2Ud42jY>

Path to Programmable III: Ultra96v2 dual-core baremetal AMP (Asymmetric Multi Processing) design - A53, R5. saadtiwana\_int. <https://community.element14.com/challenges-projects/design-challenges/pathprogrammable3/b/blog/posts/path-to-programmable-iii-ultra96v2-dual-core-a53-r5-baremetal-operation---vitis-2022-2>

Ultra96-V2 - Building the foundational designs. Mario Bergeron.

<https://www.hackster.io/AlbertaBeef/ultra96-v2-building-the-foundational-designs-e4315f#overview>