
This is the **published version** of the bachelor thesis:

Sillero Franco, Òscar. *Connectivitat Wi-Fi i Bluetooth per a una placa docent*.
Treball de Final de Grau (Universitat Autònoma de Barcelona), 2026 (Grau en
Enginyeria Electrònica de Telecomunicació)

This version is available at <https://ddd.uab.cat/record/336113>

under the terms of the  license.



TREBALL DE FI DE GRAU

GRAU EN ENGINYERIA ELECTRÒNICA DE TELECOMUNICACIÓ

CONNECTIVITAT WI-FI I BLUETOOTH PER A UNA PLACA DOCENT

Òscar Sillero Franco

DIRECTOR: Màrius Montón Macian

DEPARTAMENT DE MICROELECTRÒNICA I SISTEMES ELECTRÒNICS

UNIVERSITAT AUTÒNOMA DE BARCELONA

Bellaterra, Juny 28, 2026

Abstract

This Bachelor's Final Project focuses on the integration and validation of wireless connectivity on a teaching printed circuit board designed for use in different subjects at the Escola d'Enginyeria of the Universitat Autònoma de Barcelona. The work is part of the redesign of a previous teaching board prototype and aims to expand its capabilities by incorporating Wi-Fi and Bluetooth Low Energy connectivity.

To achieve this objective, a study of different ESP32 modules has been carried out in order to identify which option best met the technical and functional requirements of the new board. The analysis considered several technical parameters of the modules studied, as well as their compatibility with the Teensy 3.6 and Giant Gecko Starter Kit development platforms. Based on this study, the ESP32-C6-WROOM-1 module was selected, mainly due to its support for Wi-Fi, Bluetooth Low Energy and communication interfaces suitable for integration into the teaching platform.

The project includes the analysis of the module connections, the study of its pinout and the definition of its integration with the rest of the board. In addition, the necessary firmware has been developed and validated to verify the operation of the ESP32-C6 module in different communication scenarios.

Different development environments and tools have been used to carry out these tasks. ESP-IDF was used for programming and validating the ESP32-C6 module, while Simplicity Studio was used to perform tests with the Giant Gecko platform. nRF Connect was also used to verify Bluetooth Low Energy communication with a mobile phone, as well as serial terminals to test communication with the module through AT commands. Finally, Altium Designer was used as a support tool to review the hardware integration of the module within the overall board design.

Thus, this work combines the study and selection of the communication module, its integration into an existing teaching platform and the performance of functional tests aimed at validating its use in future educational applications.

Resum

Aquest Treball de Fi de Grau se centra en la integració i validació de connectivitat sense fils en una placa de circuit imprès docent pensada per ser utilitzada en diferents assignatures de l'Escola d'Enginyeria de la Universitat Autònoma de Barcelona. El treball s'emmarca dins del redisseny d'un prototip anterior de placa docent i té com a objectiu ampliar-ne les capacitats mitjançant la incorporació de connectivitat Wi-Fi i Bluetooth Low Energy.

Per assolir aquest objectiu, s'ha dut a terme un estudi de diferents mòduls ESP32 amb la finalitat d'identificar quina opció s'adaptava millor als requisits tècnics i funcionals de la nova placa. L'anàlisi ha tingut en compte diversos paràmetres tècnics dels mòduls estudiats, així com la seva compatibilitat amb les plataformes de desenvolupament Teensy 3.6 i Giant Gecko Starter Kit. A partir d'aquest estudi, s'ha seleccionat el mòdul ESP32-C6-WROOM-1, principalment pel seu suport per a Wi-Fi, Bluetooth Low Energy i interfícies de comunicació adequades per a la integració dins la plataforma docent.

El projecte inclou l'anàlisi de les connexions del mòdul, l'estudi del seu pinout i la definició de la seva integració amb la resta de la placa. A més, s'ha desenvolupat i validat el firmware necessari per comprovar el funcionament del mòdul ESP32-C6 en diferents escenaris de comunicació.

Per dur a terme aquestes tasques s'han utilitzat diferents entorns i eines de desenvolupament. ESP-IDF s'ha emprat per a la programació i validació del mòdul ESP32-C6, mentre que Simplicity Studio s'ha utilitzat per realitzar proves amb la plataforma Giant Gecko. També s'ha fet servir nRF Connect per verificar la comunicació Bluetooth Low Energy amb un telèfon mòbil, així com terminals sèrie per comprovar la comunicació amb el mòdul mitjançant comandes AT. Finalment, Altium Designer s'ha utilitzat com a eina de suport per revisar la integració hardware del mòdul dins del disseny general de la placa.

D'aquesta manera, el treball combina l'estudi i selecció del mòdul de comunicació, la seva integració dins d'una plataforma docent existent i la realització de proves funcionals orientades a validar-ne l'ús en futures aplicacions educatives.

Índex

Abstract	i
Resum	ii
Llista de figures	x
Llista de taules	xi
1 Introducció	1
1.1 Context i motivació	1
1.2 Abast del projecte	2
1.3 Estructura del document	3
2 Objectius del treball	5
2.1 Objectiu general	5
2.2 Objectius específics	5
3 Estat de l'art	8
3.1 Placa docent de partida	8
3.1.1 Descripció del prototip anterior	9
3.1.2 Limitacions i aspectes a millorar	10
3.1.3 Treball previ i projecte complementari	10
3.2 Plataformes de desenvolupament	11

3.2.1	Teensy 3.6	11
3.2.2	Giant Gecko Starter Kit	12
3.3	Família ESP32 i tecnologies de comunicació	13
3.4	Comparació de les variants ESP32	14
3.4.1	Sèrie ESP8266	14
3.4.2	Sèrie ESP32	15
3.4.3	Sèrie ESP32-S	15
3.4.4	Sèrie ESP32-C	16
3.4.5	Sèrie ESP32-H	17
3.4.6	Sèrie ESP32-P	18
3.5	Selecció del mòdul ESP32-C6-WROOM-1	18
3.5.1	Criteris de selecció	19
3.5.2	Comparació de formats disponibles	19
3.5.3	Elecció del format WROOM-1	20
3.5.4	Cost i disponibilitat	21
4	Implementació	22
4.1	Arquitectura general del sistema	22
4.1.1	Funció de l'ESP32-C6 dins de la placa docent	24
4.1.2	Relació amb la Teensy 3.6 i la Giant Gecko	24
4.2	Integració hardware de l'ESP32-C6-WROOM-1	25
4.2.1	Anàlisi del pinout	26
4.2.2	Alimentació i connexions de massa	28
4.2.3	Senyals de reset i boot	29
4.2.4	Interfície UART	29
4.2.5	Interfície USB i programació	30
4.2.6	Connexions incorporades a l'esquemàtic	30
4.3	Entorns i eines de desenvolupament	31

4.3.1	ESP-IDF	31
4.3.2	Simplicity Studio	32
4.3.3	nRF Connect	32
4.3.4	ESP-AT i terminals sèrie	32
4.3.5	Altium Designer	33
4.4	Desenvolupament del firmware	33
4.4.1	Proves bàsiques de funcionament	33
4.4.2	Connexió Wi-Fi en mode Station	34
4.4.3	Configuració Wi-Fi en mode SoftAP	34
4.4.4	Comunicació Bluetooth Low Energy	35
4.4.5	Aplicació BLE amb servei GATT	35
4.4.6	Funcionament com a mòdem mitjançant ESP-AT	36
4.4.7	Adaptació de la UART per a les comandes AT	37
5	Entregables	39
5.1	Materials desenvolupats	39
5.2	Carpeta de materials del projecte	40
5.3	Manual docent	41
6	Pla de proves i resultats	42
6.1	Metodologia de validació	42
6.2	Proves bàsiques del mòdul	45
6.2.1	Compilació, programació i monitorització	45
6.2.2	Proves de GPIO	46
6.3	Validació de la connectivitat Wi-Fi	47
6.3.1	Resultats de la prova Wi-Fi Station	47
6.3.2	Resultats de la prova Wi-Fi SoftAP	47
6.4	Validació de Bluetooth Low Energy	48

6.4.1	Detecció i connexió des de nRF Connect	49
6.4.2	Intercanvi de dades mitjançant GATT	50
6.5	Validació del funcionament com a mòdem	51
6.5.1	Càrrega del firmware ESP-AT	51
6.5.2	Configuració del port de comandes	52
6.5.3	Resposta a les comandes AT	52
6.5.4	Connexió a una xarxa Wi-Fi	53
6.5.5	Obertura de la connexió TCP	54
6.5.6	Enviament de dades al servidor	55
6.6	Validació amb les plataformes de desenvolupament	56
6.6.1	Proves amb la Giant Gecko	56
6.6.2	Compatibilitat amb la Teensy 3.6	57
6.7	Resum dels resultats	58
7	Conclusions	60
7.1	Conclusions generals	60
7.2	Grau d'assoliment dels objectius	61
7.3	Línies futures	62
A	Manual docent de configuració i proves	64
A.1	Introducció	64
A.2	Organització dels materials lliurats	65
A.3	Preparació dels entorns de desenvolupament	65
A.3.1	Instal·lació i configuració d'ESP-IDF	66
A.3.2	Instal·lació de nRF Connect	67
A.3.3	Instal·lació i configuració de Simplicity Studio	68
A.3.4	Preparació de Python i del terminal sèrie	69
A.4	Connexió i identificació de les plaques	69

A.4.1	Connexió de la placa ESP32-C6	69
A.4.2	Identificació del port COM a Windows	70
A.4.3	Connexió de la Giant Gecko Starter Kit	70
A.4.4	Comprovacions prèvies	71
A.5	Execució dels projectes de l'ESP32-C6	71
A.5.1	Projecte Hello World	71
A.5.2	Projecte GPIO Blink	72
A.5.3	Projecte Wi-Fi Station	73
A.5.4	Projecte Wi-Fi SoftAP	73
A.5.5	Projecte Bluetooth Low Energy amb GATT	74
A.5.6	Projecte ESP-AT	75
A.5.7	Connexió Wi-Fi mitjançant comandes AT	76
A.5.8	Connexió TCP i enviament de dades	77
A.6	Execució dels projectes de la Giant Gecko	77
A.6.1	Projecte GPIO, LED i polsadors	78
A.6.2	Projecte ADC	78
A.7	Comandes principals	79
A.7.1	Comandes generals d'ESP-IDF	79
A.7.2	Comandes del terminal sèrie i de Python	80
A.7.3	Comandes principals d'ESP-AT	80
A.8	Paràmetres que cal adaptar	81
A.8.1	Port de comunicació	81
A.8.2	Rutes dels projectes	82
A.8.3	Configuració de la xarxa Wi-Fi	82
A.8.4	Configuració del punt d'accés Wi-Fi	82
A.8.5	Adreça IP i port del servidor TCP	83
A.8.6	Longitud del missatge TCP	83

A.8.7	Pins GPIO	84
A.8.8	Paràmetres Bluetooth Low Energy	84
A.8.9	Configuració UART d'ESP-AT	84
A.8.10	Paràmetres de la Giant Gecko	85
A.8.11	Enllaç als materials	85
Bibliografia		87

Llista de figures

1	Primera versió de la placa docent basada en la Teensy 3.5.	9
2	Símbol i connexions principals de la Teensy 3.6 dins de l'esquemàtic de la placa docent.	12
3	Giant Gecko Starter Kit utilitzada durant el desenvolupament i les proves. . . .	13
4	Mòdul ESP32-C6-WROOM-1-N8 seleccionat per a la nova placa docent.	18
5	Distribució i numeració dels pins de l'ESP32-C6-WROOM-1.	27
6	Programació de l'ESP32-C6, obertura del monitor sèrie i execució correcta del projecte Hello World.	45
7	Execució de la prova de GPIO mitjançant el control del LED integrat de l'ESP32-C6. . . .	46
8	Connexió de l'ESP32-C6 a una xarxa Wi-Fi en mode Station i obtenció correcta d'una adreça IP.	47
9	Connexió del telèfon mòbil al punt d'accés Wi-Fi creat per l'ESP32-C6.	48
10	Creació d'un punt d'accés Wi-Fi amb l'ESP32-C6 i connexió d'un dispositiu client, al qual s'assigna una adreça IP mitjançant DHCP.	48
11	Inicialització de la pila BLE i establiment correcte de la connexió amb l'ESP32-C6. . . .	49
12	Detecció i connexió amb l'ESP32-C6 mitjançant l'aplicació nRF Connect.	49
13	Enviament del missatge "Comunicació BLE correcta" des de l'aplicació nRF Connect mitjançant una operació d'escriptura GATT.	50
14	Recepció i visualització del missatge enviat mitjançant GATT al monitor sèrie de l'ESP32-C6.	50

15	Inicialització correcta del firmware ESP-AT i configuració del port de comandes de l'ESP32-C6.	51
16	Resposta de l'ESP32-C6 a les comandes AT i AT+GMR mitjançant el firmware ESP-AT.	53
17	Connexió de l'ESP32-C6 a una xarxa Wi-Fi i consulta de l'adreça IP mitjançant comandes AT.	54
18	Obertura correcta d'una connexió TCP des de l'ESP32-C6 mitjançant la comanda AT+CIPSTART.	55
19	Connexió de l'ESP32-C6 registrada pel servidor TCP executat a l'ordinador. . .	55
20	Comandes utilitzades per establir la connexió TCP i iniciar l'enviament de dades des de l'ESP32-C6.	56
21	Recepció correcta del missatge enviat per l'ESP32-C6 al servidor TCP.	56
22	Lectura dels valors del potenciòmetre i del termistor mitjançant l'ADC de la Giant Gecko Starter Kit.	57

Llista de taules

1	Comparació dels formats de mòdul disponibles per a l'ESP32-C6.	20
2	Nom, numeració, tipus i funcions dels pins de l'ESP32-C6-WROOM-1.	28
3	Comparació entre la configuració original i la configuració final del port de comandes ESP-AT.	38
4	Resum general de les proves de validació realitzades.	44
5	Proves i exemples realitzats sobre cada placa.	59

Capítol 1

Introducció

1.1 Context i motivació

En els darrers anys, les plataformes docents basades en sistemes electrònics programables han estat una eina molt útil per a l'aprenentatge pràctic dins l'àmbit de l'enginyeria. Aquest tipus de plaques permeten treballar conceptes relacionats amb la programació de microcontroladors, la comunicació entre dispositius, l'adquisició de dades, el control de perifèrics i la integració de diferents tecnologies dins d'un mateix sistema.

Disposar d'una plataforma flexible, ampliable i compatible amb diferents entorns de desenvolupament facilita la preparació de pràctiques i activitats formatives per a diverses assignatures. A més, permet que els estudiants treballin amb dispositius i tecnologies similars als que es poden trobar en aplicacions professionals.

Aquest Treball de Fi de Grau se centra en el redisseny d'una placa docent desenvolupada per a l'Escola d'Enginyeria de la Universitat Autònoma de Barcelona. El projecte parteix d'un prototip anterior i busca ampliar-ne les funcionalitats mitjançant la incorporació de connectivitat sense fils. Aquesta millora permet que la placa no només pugui treballar amb sensors, actuadors i altres perifèrics locals, sinó també comunicar-se amb dispositius externs mitjançant Wi-Fi i Bluetooth Low Energy.

La incorporació d'aquestes tecnologies amplia les possibilitats d'ús de la plataforma i permet plantejar noves pràctiques relacionades amb comunicacions sense fils, transmissió de dades, control remot i interacció amb telèfons mòbils, ordinadors o altres sistemes electrònics.

El projecte s'ha desenvolupat en paral·lel amb el redisseny complet de la placa de circuit imprès, dut a terme en un Treball de Fi de Grau complementari. Aquest altre treball se centra principalment en la modificació dels esquemàtics, el desenvolupament del layout i la preparació dels fitxers necessaris per a la fabricació i el muntatge de la PCB. En canvi, aquest treball se

centra en l'estudi, la selecció, la integració i la validació funcional del mòdul de comunicació sense fils.

D'aquesta manera, els dos projectes contribueixen a un mateix objectiu general: obtenir una nova versió de la placa docent més completa, versàtil i adequada per a futures aplicacions educatives.

1.2 Abast del projecte

El treball comprèn l'estudi de diferents dispositius de la família ESP32 amb la finalitat d'identificar quina opció s'adapta millor als requisits tècnics i funcionals de la nova placa docent. A partir d'aquesta comparació, s'ha seleccionat el mòdul ESP32-C6-WROOM-1, que incorpora connectivitat Wi-Fi i Bluetooth Low Energy i disposa d'interfícies adequades per comunicar-se amb altres dispositius.

Un cop seleccionat el mòdul, s'ha analitzat el seu pinout, les connexions d'alimentació, els senyals de control, els pins d'arrencada i les interfícies de comunicació disponibles. També s'ha estudiat la manera d'integrar-lo amb les plataformes Teensy 3.6 i Giant Gecko Starter Kit, que actuen com a controladors principals de la placa.

Una part especialment rellevant del projecte és la definició d'una comunicació UART entre l'ESP32-C6 i aquestes plataformes. Aquesta interfície permet que la Teensy o la Giant Gecko enviïn instruccions i dades al mòdul, mentre que l'ESP32-C6 s'encarrega de gestionar les comunicacions sense fils.

El treball també inclou el desenvolupament i la validació del firmware necessari per comprovar el funcionament del mòdul en diferents escenaris. S'han realitzat proves bàsiques de programació i execució, connexió a una xarxa Wi-Fi, creació d'un punt d'accés, comunicació Bluetooth Low Energy i intercanvi de dades amb un telèfon mòbil.

A més, s'ha estudiat l'ús del firmware ESP-AT per convertir l'ESP32-C6 en un mòdul de comunicació controlat mitjançant comandes AT. Aquesta part ha requerit adaptar la configuració de la UART per permetre l'enviament de comandes, la connexió a una xarxa Wi-Fi i la transmissió de dades mitjançant una connexió TCP.

Per desenvolupar aquestes tasques s'han utilitzat diferents eines i entorns de treball. ESP-IDF s'ha emprat per programar i validar l'ESP32-C6, mentre que Simplicity Studio s'ha utilitzat per realitzar proves amb la plataforma Giant Gecko. L'aplicació nRF Connect ha permès verificar la comunicació Bluetooth Low Energy, i els terminals sèrie s'han utilitzat per monitorar el funcionament del dispositiu i enviar comandes AT. Altium Designer també s'ha fet servir com a eina de suport per revisar la integració del mòdul dins del disseny general de la placa.

L'abast del present TFG no inclou el desenvolupament complet del layout ni la generació de tots els fitxers de fabricació de la PCB, ja que aquestes tasques formen part del projecte complementari. No obstant això, sí que inclou la definició i revisió de les connexions necessàries per integrar correctament el mòdul ESP32-C6-WROOM-1 dins de la nova placa docent.

1.3 Estructura del document

Aquest document s'organitza en set capítols i un annex, cadascun dels quals presenta una part concreta del procés seguit durant el desenvolupament del projecte.

El primer capítol introdueix el context i la motivació del treball, en delimita l'abast i descriu l'organització general de la memòria.

El segon capítol presenta els objectius del treball, diferenciant entre l'objectiu general del projecte i els objectius específics definits per assolir-lo.

El tercer capítol correspon a l'estat de l'art. En aquest apartat es descriu la placa docent de partida, els treballs previs i les plataformes de desenvolupament Teensy 3.6 i Giant Gecko Starter Kit. També es presenta la família ESP32, es comparen les seves principals variants i es justifica la selecció del mòdul ESP32-C6-WROOM-1-N8.

El quart capítol descriu la implementació del sistema. S'hi presenta l'arquitectura general de la placa docent, la funció de l'ESP32-C6 i la seva relació amb les plataformes Teensy 3.6 i Giant Gecko. També s'explica la integració hardware del mòdul, incloent-hi l'anàlisi del pinout, l'alimentació, els senyals de reset i boot, les interfícies UART i USB Serial/JTAG i les connexions incorporades a l'esquemàtic.

En aquest mateix capítol es descriuen els entorns i les eines utilitzades durant el desenvolupament, així com l'estructura i el funcionament general del firmware preparat per implementar les funcionalitats Wi-Fi, Bluetooth Low Energy i ESP-AT. En aquest apartat no s'inclouen els procediments complets d'instal·lació ni els codis font, ja que aquesta informació es recull de manera específica a l'annex.

El cinquè capítol presenta els materials generats durant el desenvolupament del projecte i la manera com es posen a disposició del professorat i de futurs usuaris de la plataforma. S'hi descriuen de manera general els projectes de firmware preparats, la documentació relacionada amb la integració del mòdul i els recursos necessaris per continuar el treball.

També s'explica l'organització de la carpeta compartida que actua com a punt d'accés als codis, als fitxers de configuració i a la resta de materials desenvolupats.

El sisè capítol presenta el pla de proves i els resultats obtinguts. S'hi descriu la metodologia de

validació i s'analitzen els resultats de les proves bàsiques de programació i GPIO, la connectivitat Wi-Fi en els modes Station i SoftAP, la comunicació Bluetooth Low Energy, l'intercanvi de dades mitjançant GATT, el funcionament de l'ESP32-C6 com a mòdem mitjançant comandes AT i l'enviament de dades a través d'una connexió TCP.

També s'hi presenten les proves realitzades amb la Giant Gecko Starter Kit i es diferencia entre les funcionalitats validades experimentalment i aquelles que han quedat preparades a nivell de disseny per a una validació posterior.

El setè capítol recull les conclusions generals del treball, el grau d'assoliment dels objectius i les possibles línies de continuació i millora del projecte.

Finalment, l'Annex A conté el manual docent de configuració i proves. Aquest annex està destinat principalment al professorat i als futurs usuaris de la plataforma i recull els procediments necessaris per preparar els entorns de desenvolupament, configurar els projectes, programar les plaques i comprovar-ne el funcionament.

Els codis font complets, els scripts auxiliars i els fitxers de configuració es proporcionen en una carpeta externa accessible mitjançant l'enllaç indicat al mateix annex. Al llarg del manual s'especifica la ubicació de cada projecte dins d'aquesta carpeta, els paràmetres que cal adaptar, les comandes necessàries i el resultat esperat de cada prova.

D'aquesta manera, la memòria principal presenta el desenvolupament, les decisions tècniques i els resultats del projecte, mentre que l'annex funciona com una guia pràctica per reproduir les proves utilitzant els materials lliurats.

Capítol 2

Objectius del treball

Aquest capítol presenta l'objectiu general del projecte i els diferents objectius específics definits per assolir-lo. Aquests objectius inclouen tant l'estudi i la selecció del mòdul de comunicació com la seva integració hardware, el desenvolupament del firmware i la validació de les funcionalitats incorporades.

2.1 Objectiu general

L'objectiu general d'aquest Treball de Fi de Grau és integrar i validar un mòdul de comunicació sense fils en una placa docent destinada a diferents assignatures de l'Escola d'Enginyeria de la Universitat Autònoma de Barcelona.

La incorporació d'aquest mòdul té com a finalitat ampliar les funcionalitats de la plataforma mitjançant connectivitat Wi-Fi i Bluetooth Low Energy. Al mateix temps, es busca mantenir la compatibilitat amb les plataformes de desenvolupament Teensy 3.6 i Giant Gecko Starter Kit, de manera que el mòdul pugui treballar de manera autònoma o actuar com a element auxiliar de comunicació controlat per un altre microcontrolador.

La solució desenvolupada ha de permetre utilitzar la placa en pràctiques relacionades amb sistemes encastats, comunicacions sense fils, transmissió de dades i interacció amb dispositius externs. També es pretén que la integració sigui flexible i ampliable, facilitant el desenvolupament de noves aplicacions docents en el futur.

2.2 Objectius específics

Per assolir l'objectiu general del projecte, s'han definit els objectius específics següents:

- Estudiar les principals variants i mòduls disponibles dins de la família ESP32, tenint en compte les seves característiques tècniques, les tecnologies de comunicació incorporades, el cost i la facilitat d'integració.
- Seleccionar el mòdul que s'adapti millor als requisits tècnics i funcionals de la nova placa docent.
- Analitzar detalladament el mòdul ESP32-C6-WROOM-1 seleccionat, incloent-hi el pinout, l'alimentació, els senyals de control, els pins d'arrencada i les interfícies de comunicació disponibles.
- Definir les connexions necessàries per integrar l'ESP32-C6-WROOM-1 dins del disseny general de la placa.
- Estudiar la comunicació entre l'ESP32-C6 i les plataformes Teensy 3.6 i Giant Gecko Starter Kit, especialment mitjançant una interfície UART.
- Analitzar els ports USB disponibles i determinar les seves funcions de programació, comunicació sèrie i depuració.
- Preparar l'entorn de desenvolupament necessari per programar el mòdul ESP32-C6 mitjançant ESP-IDF.
- Desenvolupar proves bàsiques de firmware per comprovar la correcta programació i execució del mòdul.
- Validar la connectivitat Wi-Fi del dispositiu, tant mitjançant la connexió a una xarxa existent com mitjançant la creació d'un punt d'accés propi.
- Validar la comunicació Bluetooth Low Energy mitjançant la detecció i connexió del mòdul des d'un telèfon mòbil.
- Desenvolupar una aplicació BLE senzilla que permeti intercanviar dades entre el telèfon mòbil i l'ESP32-C6 mitjançant un servei GATT.
- Configurar i adaptar el firmware ESP-AT perquè el mòdul pugui ser controlat mitjançant comandes AT des d'una interfície sèrie accessible.
- Comprovar el funcionament de l'ESP32-C6 com a mòdem Wi-Fi, incloent-hi la connexió a una xarxa, l'obtenció d'una adreça IP, l'obertura d'una connexió TCP i l'enviament de dades.
- Desenvolupar proves bàsiques amb la plataforma Giant Gecko Starter Kit per comprovar-ne el funcionament i facilitar-ne la integració amb la placa docent.

- Elaborar un pla de proves que permeti verificar de manera ordenada les funcionalitats implementades.
- Generar manuals i documentació tècnica que facilitin la configuració, programació, validació i ús del sistema per part d'altres usuaris.

El compliment d'aquests objectius ha de permetre disposar d'una integració funcional i documentada del mòdul ESP32-C6-WROOM-1, preparada per ser incorporada a la nova versió de la placa docent i validada experimentalment sobre aquesta plataforma.

Capítol 3

Estat de l'art

Aquest capítol presenta el context tecnològic i els antecedents necessaris per entendre les decisions adoptades durant el desenvolupament del projecte. En primer lloc, es descriu la placa docent de partida i es presenten les principals necessitats que han motivat el seu redisseny. També s'introdueixen les plataformes Teensy 3.6 i Giant Gecko Starter Kit, amb les quals la nova placa ha de ser compatible.

A continuació, es presenta la família de dispositius ESP32 i les principals tecnologies de comunicació que incorpora. Posteriorment, es comparen les variants més representatives de la família per determinar quines s'adapten millor als requisits del projecte. Finalment, es justifica la selecció de l'ESP32-C6 i del mòdul ESP32-C6-WROOM-1 com a solució per incorporar connectivitat Wi-Fi i Bluetooth Low Energy a la nova placa docent.

3.1 Placa docent de partida

El present projecte parteix d'una primera versió de placa docent desenvolupada a l'Escola d'Enginyeria de la Universitat Autònoma de Barcelona. La plataforma va sorgir com una iniciativa conjunta dels departaments de Microelectrònica i Sistemes Electrònics, i Arquitectura de Computadors i Sistemes Operatius, amb la col·laboració de l'UAB Open Lab.

L'objectiu d'aquesta primera versió era disposar d'una plataforma oberta i ampliable que pogués ser utilitzada en diferents assignatures relacionades amb microcontroladors, electrònica digital i sistemes encastats. D'aquesta manera, es buscava reduir la dependència de muntatges específics per a cada assignatura, simplificar la preparació del material docent i oferir als estudiants una plataforma comuna.

La placa es va desenvolupar com una extensió de la Teensy 3.5 i incorporava diferents sensors, actuadors i interfícies de comunicació. Els resultats del projecte anterior van permetre demostrar

la viabilitat del concepte i establir una base funcional per al desenvolupament d'una nova versió de la plataforma [1].

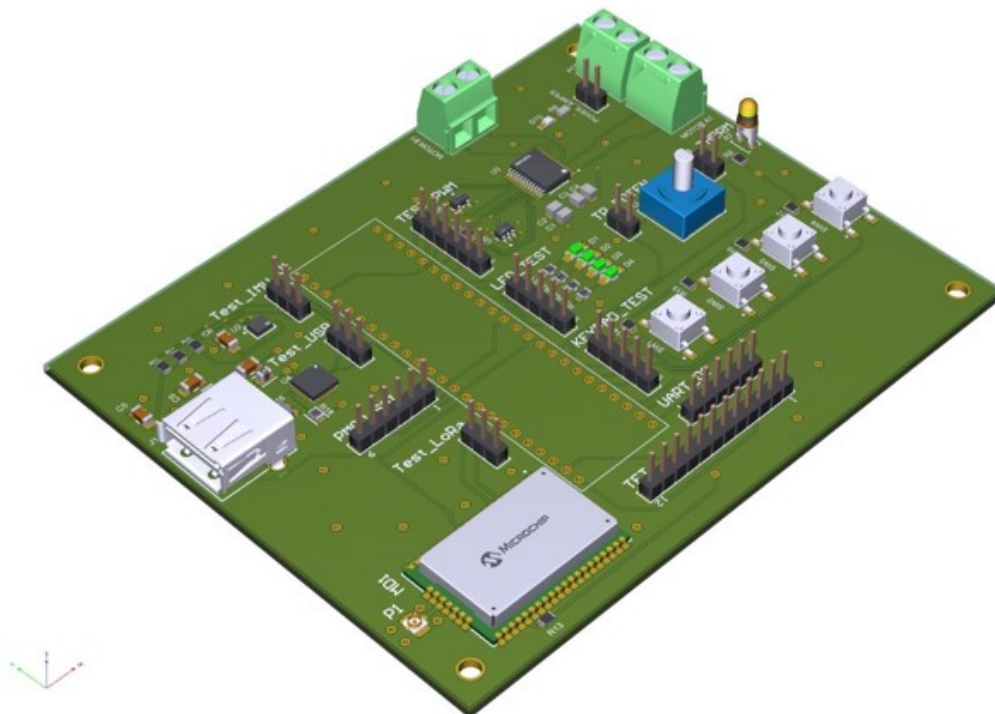


Figura 1: Primera versió de la placa docent basada en la Teensy 3.5.

Font: Treball de Fi de Grau d'Adrià Rafecas Rubió [1].

3.1.1 Descripció del prototip anterior

La primera versió de la placa utilitzava la Teensy 3.5 com a microcontrolador principal. Aquesta plataforma s'encarregava de controlar els diferents elements incorporats a la PCB i d'executar els programes desenvolupats durant les pràctiques.

La placa integrava diferents busos i interfícies, com I2C, SPI, UART, PWM, entrades i sortides digitals, entrades analògiques, USB i connectors PMOD i TFT. Aquesta varietat permetia treballar conceptes relacionats amb la comunicació entre dispositius, l'adquisició de dades i el control de perifèrics.

Entre els components incorporats hi havia polsadors, LED, un potenciòmetre, un termistor, una unitat de mesura inercial, un controlador de motors, un convertidor USB-UART i un mòdul de comunicació LoRa. La selecció d'aquests elements es va realitzar amb la finalitat de disposar de perifèrics representatius de les diferents funcionalitats i protocols disponibles.

El disseny també tenia en compte l'ús docent de la plataforma. Per aquest motiu, es va prioritzar una distribució visual clara, la identificació dels components i una mida suficient per facilitar-ne el muntatge, la inspecció i les possibles reparacions. La placa es va fabricar amb dues capes per mantenir un cost reduït i facilitar la producció de diverses unitats.

3.1.2 Limitacions i aspectes a millorar

Tot i que la primera versió va permetre validar el concepte de placa docent, durant el seu desenvolupament i les proves posteriors es van identificar diferents aspectes que podien ser revisats.

Una de les principals limitacions era que la plataforma estava dissenyada específicament per treballar amb la Teensy 3.5. Aquesta dependència reduïa la possibilitat d'utilitzar altres plaques de desenvolupament i dificultava l'adaptació de la plataforma a diferents assignatures.

La placa disposava de comunicació LoRa i d'un convertidor USB-UART, però no incorporava un mòdul capaç d'oferir simultàniament connectivitat Wi-Fi i Bluetooth Low Energy. La integració d'aquestes tecnologies permetria connectar la plataforma amb xarxes locals, ordinadors i telèfons mòbils, així com desenvolupar noves pràctiques de comunicacions sense fils.

També es va considerar necessari revisar els esquemàtics, les connexions entre blocs i la selecció d'alguns components. El redisseny havia de mantenir les funcionalitats útils de la primera versió, corregir les incidències detectades i millorar la compatibilitat amb les plataformes Teensy 3.6 i Giant Gecko Starter Kit.

3.1.3 Treball previ i projecte complementari

El treball desenvolupat per Adrià Rafecas [1] constitueix el punt de partida de la nova versió de la placa docent. En aquell projecte es van definir els requisits inicials de la plataforma, es van seleccionar els principals components i es va realitzar el disseny, la fabricació i la validació d'un primer prototip compatible amb la Teensy 3.5.

A partir d'aquesta base, el redisseny actual s'ha dividit en dos Treballs de Fi de Grau complementaris. Aquesta divisió permet abordar de manera més detallada tant el desenvolupament físic de la nova PCB com la incorporació de les noves funcionalitats de comunicació.

El projecte complementari se centra principalment en la revisió dels esquemàtics, la selecció i incorporació dels components, la creació dels footprints, el desenvolupament del layout i la preparació dels fitxers necessaris per fabricar i muntar la placa.

El present treball se centra en l'estudi i la selecció del mòdul de comunicació sense fils, l'anàlisi de les seves connexions i la definició de la seva integració amb la Teensy 3.6 i la Giant

Gecko. També inclou el desenvolupament del firmware i les proves necessàries per validar les funcionalitats Wi-Fi, Bluetooth Low Energy i el funcionament del mòdul com a mòdem mitjançant comandes AT.

Tot i que els dos treballs tenen objectius específics diferents, formen part d'un mateix projecte general: obtenir una nova versió de la placa docent més flexible, completa i preparada per a futures aplicacions educatives.

3.2 Plataformes de desenvolupament

La nova placa docent s'ha dissenyat per treballar principalment amb dues plataformes de desenvolupament: la Teensy 3.6 i la Giant Gecko Starter Kit. Aquestes plataformes poden actuar com a sistemes principals de control i comunicar-se amb els diferents sensors, actuadors i mòduls integrats a la placa.

La incorporació de l'ESP32-C6 no pretén substituir aquestes plataformes, sinó complementar-les amb funcionalitats de comunicació sense fils. Per aquest motiu, és necessari disposar d'una interfície que permeti intercanviar ordres i dades entre el microcontrolador principal i el mòdul ESP32.

3.2.1 Teensy 3.6

La Teensy 3.6 és una placa de desenvolupament basada en un microcontrolador ARM Cortex-M4F de 32 bits [2, 3]. Es tracta d'una plataforma compacta amb una elevada capacitat de processament i un nombre ampli de pins d'entrada i sortida [4].

La disponibilitat de diferents interfícies de comunicació permet connectar la Teensy amb sensors, actuadors, memòries i altres microcontroladors. Entre aquestes interfícies es troben UART, SPI i I2C, que resulten especialment útils per establir comunicació amb mòduls externs.

En el context de la placa docent, la Teensy 3.6 pot actuar com a controlador principal del sistema. D'aquesta manera, pot executar el programa principal de la pràctica, adquirir dades dels sensors i enviar informació a l'ESP32-C6 perquè aquest gestioni la comunicació Wi-Fi o Bluetooth Low Energy.

La comunicació entre la Teensy i l'ESP32-C6 s'ha definit mitjançant una interfície UART, connectant la sortida TX d'un dispositiu amb l'entrada RX de l'altre. Aquesta solució permet separar la programació de l'aplicació principal de la gestió de les comunicacions sense fils.

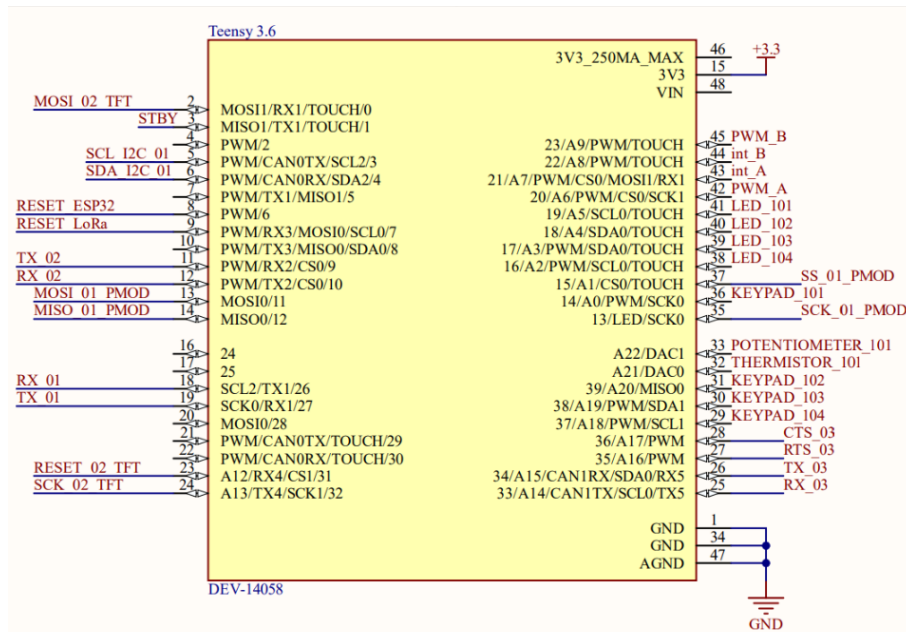


Figura 2: Símbol i connexions principals de la Teensy 3.6 dins de l'esquemàtic de la placa docent.

Font: Altium Designer.

3.2.2 Giant Gecko Starter Kit

La Giant Gecko Starter Kit és una plataforma de desenvolupament basada en microcontroladors EFM32 de Silicon Labs [5]. La placa incorpora diferents sensors i perifèrics que permeten comprovar les funcionalitats del microcontrolador i desenvolupar aplicacions de prova.

La plataforma disposa d'un depurador integrat que permet programar i depurar el microcontrolador sense necessitat d'utilitzar eines externes. El seu entorn principal de desenvolupament és Simplicity Studio, que proporciona les eines necessàries per crear projectes, configurar perifèrics, compilar el codi i programar la placa [6].

Durant el projecte s'han desenvolupat proves bàsiques amb la Giant Gecko per comprovar el funcionament dels GPIO, els botons, els LED i altres perifèrics. Aquestes proves serveixen com a punt de partida per estudiar posteriorment la comunicació entre la Giant Gecko i el mòdul ESP32-C6.

Igual que en el cas de la Teensy, la Giant Gecko pot actuar com a controlador principal de la placa docent. La comunicació prevista amb l'ESP32-C6 s'ha definit mitjançant una interfície UART. Aquesta configuració permet utilitzar l'ESP32 com a mòdul auxiliar de comunicació sense haver de traslladar tota l'aplicació principal al mateix ESP32.



Figura 3: Giant Gecko Starter Kit utilitzada durant el desenvolupament i les proves.

Font: Silicon Labs [5].

3.3 Família ESP32 i tecnologies de comunicació

L'ESP32 és una família de microcontroladors de tipus SoC desenvolupada per Espressif Systems, utilitzada en projectes de sistemes encastats i comunicacions sense fils [7, 8]. Aquests dispositius integren en un mateix xip capacitat de processament, memòria, mecanismes de seguretat per hardware i una gran varietat de perifèrics, com ADC, DAC, GPIO i busos de comunicació.

Una de les característiques més importants de l'ESP32 és que combina capacitat de control i comunicació en un únic dispositiu. Això permet executar programes propis i, al mateix temps, comunicar-se amb xarxes sense fils o amb altres elements del sistema.

Les interfícies de comunicació, com UART, SPI i I2C, faciliten la connexió amb sensors, pantalles, mòduls externs o altres microcontroladors. Aquesta flexibilitat permet utilitzar l'ESP32 tant com a controlador principal d'una aplicació com en forma de mòdul auxiliar de comunicació.

També cal diferenciar entre xips, mòduls i plaques de desenvolupament. El xip és el circuit integrat principal que conté el processador i els perifèrics. Els mòduls, com les famílies WROOM o WROVER, incorporen el xip juntament amb memòria flash, antena i els components principals de radiofreqüència [9]. Finalment, les plaques de desenvolupament afegeixen elements com reguladors de tensió, connectors USB, botons de reset i boot, LED i pins accessibles.

En aquest projecte, la família ESP32 resulta adequada perquè permet incorporar comunicació Wi-Fi i Bluetooth a la placa docent sense necessitat d'afegir sistemes de radiofreqüència independents. A més, Espressif proporciona l'entorn ESP-IDF, documentació tècnica, exemples oficials i el firmware ESP-AT, fet que facilita el desenvolupament i la validació del sistema [10, 11].

Abans de seleccionar un model concret, és necessari comparar les variants disponibles i

valorar quina ofereix el millor equilibri entre connectivitat, prestacions, facilitat d'integració, disponibilitat i cost.

3.4 Comparació de les variants ESP32

La família ESP està formada per diferents sèries i models orientats a necessitats diverses. Tot i que comparteixen un mateix ecosistema de desenvolupament, existeixen diferències importants pel que fa a l'arquitectura del processador, la connectivitat disponible, el consum, el nombre de perifèrics i l'aplicació principal.

En aquest apartat es presenten les sèries ESP8266, ESP32, ESP32-S, ESP32-C, ESP32-H i ESP32-P [12, 7]. Dins de cada sèrie s'analitzen els models més representatius amb l'objectiu d'identificar les opcions que compleixen els requisits del projecte.

Els criteris principals de comparació són la disponibilitat de Wi-Fi i Bluetooth Low Energy, les interfícies de comunicació disponibles, la facilitat d'integració, la capacitat de processament i les possibilitats d'ampliació de la plataforma docent.

3.4.1 Sèrie ESP8266

La sèrie ESP8266 es pot considerar una de les primeres famílies de microcontroladors d'Espressif orientades a incorporar connectivitat Wi-Fi en projectes electrònics de baix cost. Aquests dispositius van tenir una gran difusió en aplicacions de prototipatge, automatització i sistemes de sensors, ja que permetien afegir comunicació sense fils de manera senzilla i econòmica.

3.4.1.1 ESP8266

L'ESP8266 integra connectivitat Wi-Fi i un microcontrolador capaç d'executar aplicacions bàsiques de control i comunicació. Tot i això, en comparació amb les variants posteriors de la família ESP32, presenta unes prestacions més limitades.

Aquest dispositiu no incorpora Bluetooth, disposa de menys perifèrics i ofereix menys flexibilitat per a projectes que requereixen diferents tecnologies de comunicació.

Per aquest motiu, encara que l'ESP8266 continua sent útil en aplicacions senzilles basades únicament en Wi-Fi, no resulta l'opció més adequada per a aquest projecte, ja que la placa docent requereix Wi-Fi i Bluetooth Low Energy en un mateix dispositiu.

3.4.2 Sèrie ESP32

La sèrie ESP32 representa una evolució respecte a l'ESP8266, ja que incorpora en un mateix xip connectivitat Wi-Fi i Bluetooth, una major capacitat de processament i un conjunt més ampli de perifèrics.

3.4.2.1 ESP32

L'ESP32 original va ser introduït per Espressif Systems l'any 2016 i utilitza un microcontrolador de doble nucli Xtensa LX6. Aquest dispositiu integra Wi-Fi, Bluetooth clàssic i Bluetooth Low Energy.

La combinació de connectivitat, capacitat de processament i cost reduït va contribuir a la seva àmplia utilització en projectes d'automatització, prototipatge i sistemes encastats.

Tot i continuar sent una opció més que vàlida, l'ESP32 original és una variant anterior a altres models que incorporen arquitectures i tecnologies de comunicació més recents. Per aquest motiu, es pren com a punt de referència dins de la comparació, però es valoren altres opcions que poden adaptar-se millor als requisits de la placa docent.

3.4.3 Sèrie ESP32-S

La sèrie ESP32-S agrupa variants orientades a aplicacions que requereixen connectivitat Wi-Fi, USB i, en alguns casos, una major capacitat de processament.

3.4.3.1 ESP32-S2

L'ESP32-S2 incorpora Wi-Fi en la banda de 2,4 GHz i connectivitat USB OTG. Aquesta característica el fa interessant en aplicacions que requereixen comunicació directa amb ordinadors o dispositius USB.

La principal limitació per al projecte és que no incorpora Bluetooth. Per tant, encara que sigui una opció vàlida per a aplicacions centrades exclusivament en Wi-Fi, no compleix un dels requisits principals de la placa docent.

3.4.3.2 ESP32-S3

L'ESP32-S3 incorpora Wi-Fi i Bluetooth Low Energy, a més d'un processador de doble nucli i recursos orientats a aplicacions que requereixen una major capacitat de processament.

Aquest model és adequat per a interfícies d'usuari, tractament de senyal, veu, imatge i altres aplicacions que poden necessitar més recursos de càlcul.

En el context d'aquest projecte, l'ESP32-S3 compleix els requisits de Wi-Fi i Bluetooth Low Energy. No obstant això, les seves prestacions estan orientades a aplicacions més exigents, mentre que la placa docent necessita principalment una solució de comunicació equilibrada, moderna i fàcil d'integrar.

3.4.4 Sèrie ESP32-C

La sèrie ESP32-C està formada per variants basades en arquitectura RISC-V i orientades a comunicacions sense fils modernes. Aquesta sèrie inclou opcions de baix cost i models més complets amb Wi-Fi 6, Bluetooth Low Energy i compatibilitat amb IEEE 802.15.4.

3.4.4.1 ESP32-C2

L'ESP32-C2 és una variant orientada a aplicacions senzilles i de baix cost. Incorpora Wi-Fi i Bluetooth Low Energy, però disposa de recursos i interfícies més limitats que altres models de la mateixa sèrie.

Pot ser adequat en dispositius compactes on el cost sigui el factor prioritari. No obstant això, per al present projecte es busca una solució amb més possibilitats d'ampliació i una major varietat d'interfícies.

3.4.4.2 ESP32-C3

L'ESP32-C3 incorpora un processador RISC-V, connectivitat Wi-Fi i Bluetooth Low Energy. Representa una opció moderna, de baix consum i compatible amb les eines de desenvolupament d'Espressif.

Aquest model compleix els requisits bàsics de connectivitat del projecte, però no disposa de Wi-Fi 6 ni de compatibilitat amb l'estàndard IEEE 802.15.4. Això impedeix utilitzar directament protocols com Zigbee o Thread, habituals en xarxes IoT de baix consum.

Per aquest motiu, tot i ser una alternativa vàlida, ofereix menys possibilitats de comunicació i ampliació que l'ESP32-C6.

3.4.4.3 ESP32-C5

L'ESP32-C5 incorpora Wi-Fi 6 de doble banda, permetent treballar en 2,4 GHz i 5 GHz. També disposa de Bluetooth Low Energy i altres tecnologies de comunicació modernes.

Aquestes característiques el converteixen en una opció molt completa. No obstant això, el Wi-Fi en la banda de 5 GHz no és un requisit imprescindible per a la placa docent i pot incrementar la complexitat o el cost sense aportar un benefici directe a les pràctiques previstes.

3.4.4.4 ESP32-C6

L'ESP32-C6 combina diverses tecnologies de comunicació sense fils en un mateix dispositiu. Integra un processador principal RISC-V de 32 bits, Wi-Fi 6 en la banda de 2,4 GHz, Bluetooth Low Energy i és compatible amb IEEE 802.15.4 [13].

La compatibilitat amb IEEE 802.15.4 permet utilitzar tecnologies com Thread i Zigbee, orientades a xarxes de dispositius de baix consum. Tot i que aquestes tecnologies no són l'objectiu principal del projecte, proporcionen possibilitats d'ampliació per a futures pràctiques.

En comparació amb l'ESP32-C2 i l'ESP32-C3, l'ESP32-C6 incorpora tecnologies de comunicació més recents. Al mateix temps, ofereix una solució més ajustada als requisits del projecte que l'ESP32-C5, ja que proporciona Wi-Fi 6 sense necessitat de treballar en la banda de 5 GHz.

Per aquests motius, l'ESP32-C6 destaca com una de les opcions que millor s'adapten a la nova placa docent.

3.4.5 Sèrie ESP32-H

La sèrie ESP32-H està orientada principalment a comunicacions de baix consum basades en Bluetooth Low Energy i IEEE 802.15.4.

3.4.5.1 ESP32-H2

L'ESP32-H2 incorpora Bluetooth Low Energy i suport per a tecnologies com Thread i Zigbee. Aquestes característiques el fan interessant per a xarxes de sensors, sistemes domòtics i comunicacions mallades de baix consum.

No obstant això, l'ESP32-H2 no incorpora Wi-Fi. Aquesta limitació fa que no compleixi un dels requisits principals del projecte i, per tant, queda descartat com a opció per a la placa docent.

3.4.6 Sèrie ESP32-P

La sèrie ESP32-P està orientada a aplicacions d'alt rendiment que requereixen una major capacitat de processament.

3.4.6.1 ESP32-P4

L'ESP32-P4 està basat en arquitectura RISC-V i està pensat per a aplicacions com interfícies gràfiques, tractament d'imatge, sistemes multimèdia i aplicacions HMI.

Tot i les seves prestacions, aquest dispositiu no integra connectivitat Wi-Fi ni Bluetooth. Per tant, no s'ajusta a l'objectiu principal del projecte i queda descartat com a mòdul de comunicació.

3.5 Selecció del mòdul ESP32-C6-WROOM-1

Després de comparar les principals variants de la família ESP32, s'ha seleccionat l'ESP32-C6 com el dispositiu que ofereix el millor equilibri entre connectivitat, prestacions, possibilitats d'ampliació i facilitat d'integració.

Posteriorment, s'han analitzat els diferents formats de mòdul disponibles basats en aquest xip. El model finalment seleccionat és l'ESP32-C6-WROOM-1-N8 [14, 15].

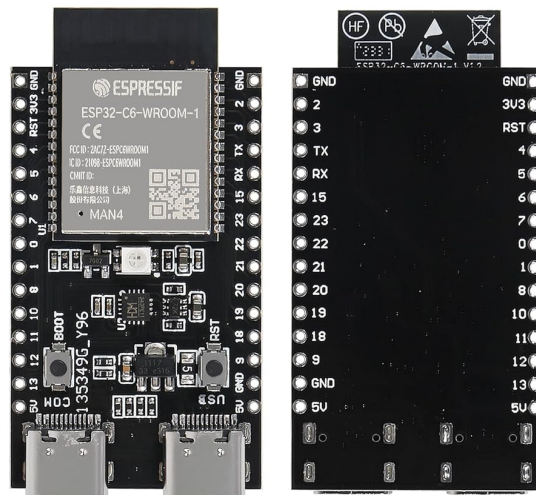


Figura 4: Mòdul ESP32-C6-WROOM-1-N8 seleccionat per a la nova placa docent.

Font: Espressif Systems [14].

3.5.1 Criteris de selecció

Els principals criteris considerats durant la selecció han estat els següents:

- Disponibilitat de Wi-Fi i Bluetooth Low Energy en un mateix dispositiu.
- Existència d'una interfície UART per comunicar-se amb la Teensy 3.6 i la Giant Gecko.
- Disponibilitat d'altres interfícies, com SPI, I2C i USB.
- Facilitat d'integració dins de la PCB.
- Disponibilitat de documentació i exemples oficials.
- Compatibilitat amb l'entorn ESP-IDF.
- Possibilitat d'utilitzar el firmware ESP-AT.
- Disponibilitat comercial.
- Cost adequat per fabricar diverses unitats de la placa docent.
- Possibilitat d'ampliar les funcionalitats en futures pràctiques.

La presència d'una UART accessible és especialment important, ja que permet utilitzar l'ESP32-C6 com a mòdul de comunicació controlat per una altra plataforma. D'aquesta manera, la Teensy o la Giant Gecko poden executar l'aplicació principal i enviar ordres a l'ESP32 perquè gestioni les comunicacions sense fils.

3.5.2 Comparació de formats disponibles

Un cop seleccionat l'ESP32-C6, s'han comparat els formats ESP32-C6-MINI-1, ESP32-C6-WROOM-1 i ESP32-C6-WROOM-1U [14, 16].

El format MINI presenta unes dimensions reduïdes i permet disminuir la superfície ocupada a la PCB. No obstant això, incorpora un nombre superior de contactes en un espai més petit, fet que incrementa la densitat del footprint i la complexitat del rutatge i del procés de fabricació.

El format WROOM-1 presenta unes dimensions més grans, però disposa d'una distribució de contactes més adequada per a una placa docent. Aquesta característica facilita la integració, la fabricació i la inspecció de les connexions.

La versió WROOM-1 incorpora una antena PCB integrada, mentre que la versió WROOM-1U utilitza un connector per a una antena externa.

Taula 1: Comparació dels formats de mòdul disponibles per a l'ESP32-C6.

Mòdul	Dimensions aproximades	Contactes	Antena	Integració
ESP32-C6-MINI-1	$13,2 \times 16,6 \times 2,4$ mm	53	Antena PCB integrada	Format compacte, però amb una densitat de contactes elevada i un footprint més complex.
ESP32-C6-WROOM-1	$18,0 \times 25,5 \times 3,1$ mm	29	Antena PCB integrada	Integració més senzilla, amb menys contactes i una distribució més accessible.
ESP32-C6-WROOM-1U	$18,0 \times 19,2 \times 3,2$ mm	29	Connector per a antena externa	Integració similar al WROOM-1, però requereix una antena externa.

Font: Elaboració pròpia a partir dels datasheets oficials d'Espressif Systems.

Com es pot observar a la Taula 1, el format MINI-1 permet reduir l'espai ocupat a la PCB, però presenta un nombre més elevat de contactes i una major densitat en el footprint. En canvi, els formats WROOM disposen de menys contactes i faciliten la integració dins de la placa docent. Entre aquests dos formats, el WROOM-1 evita la necessitat d'incorporar una antena externa, fet que simplifica el disseny i redueix el nombre d'elements necessaris.

3.5.3 Elecció del format WROOM-1

S'ha escollit el format ESP32-C6-WROOM-1 perquè ofereix una integració més senzilla i robusta que el format MINI. Tot i ocupar una superfície més gran, aquesta diferència no representa una limitació important en el context de la placa docent.

La major separació entre els contactes facilita el disseny del footprint i el rutatge de les pistes durant el desenvolupament de la PCB. A més, permet una inspecció visual més clara de les connexions un cop fabricada la placa, fet que facilita la detecció de possibles defectes durant les fases de verificació i proves.

La fabricació i el muntatge dels components de la placa es van encarregar a una empresa

especialitzada. Per tant, l'elecció del format WROOM-1 no es va basar en la realització manual de la soldadura, sinó en la facilitat d'integració del mòdul en el disseny i en la seva robustesa dins del procés de fabricació.

S'ha seleccionat la versió WROOM-1 en lloc de la WROOM-1U perquè incorpora una antena PCB integrada. El projecte no requereix una antena externa, de manera que la versió integrada permet reduir el nombre de components, evitar l'ús d'un connector d'antena i simplificar el disseny general de la placa.

La variant concreta seleccionada és l'ESP32-C6-WROOM-1-N8, equipada amb 8 MB de memòria flash. Aquesta capacitat és suficient per executar el firmware i les aplicacions previstes en el projecte, incloses les proves de Wi-Fi, Bluetooth Low Energy i comandes AT.

3.5.4 Cost i disponibilitat

El cost és un criteri important perquè la placa està destinada a un entorn docent i pot ser necessari fabricar diverses unitats.

El preu orientatiu de l'ESP32-C6-WROOM-1-N8 se situa al voltant de 4 o 5 euros per unitat en compres de poques unitats. Aquest cost es considera adequat tenint en compte que el mòdul ja incorpora el microcontrolador, la memòria flash, els circuits principals de radiofreqüència i l'antena PCB.

L'ús d'un mòdul complet també redueix el nombre de components externs i evita haver de dissenyar de manera independent la part de radiofreqüència, fet que simplifica la fabricació de la placa.

Capítol 4

Implementació

Aquest capítol descriu el procés seguit per integrar el mòdul ESP32-C6-WROOM-1-N8 dins de la nova placa docent. En primer lloc, es presenta l'arquitectura general del sistema i el paper que assumeix l'ESP32-C6 en relació amb les plataformes Teensy 3.6 i Giant Gecko Starter Kit. A continuació, s'analitzen les connexions necessàries per incorporar el mòdul al disseny, incloent-hi l'alimentació, els senyals de control, les interfícies UART i USB i els pins utilitzats a l'esquemàtic.

Posteriorment, es descriuen els entorns i les eines emprades durant el desenvolupament, així com les aplicacions preparades per implementar les funcionalitats Wi-Fi, Bluetooth Low Energy i ESP-AT.

Els procediments detallats de descàrrega, instal·lació i configuració dels entorns, juntament amb els codis complets i les instruccions necessàries per compilar-los, programar-los i executar-los, es recullen a l'Annex A. D'aquesta manera, el present capítol se centra en les decisions adoptades i en el funcionament general de les solucions desenvolupades.

4.1 Arquitectura general del sistema

La nova placa docent s'ha plantejat com una plataforma capaç de treballar amb diferents microcontroladors, sensors, actuadors i interfícies de comunicació. Dins d'aquesta arquitectura, la Teensy 3.6 o la Giant Gecko Starter Kit poden actuar com a controladors principals, mentre que l'ESP32-C6-WROOM-1-N8 aporta les funcionalitats de comunicació sense fils.

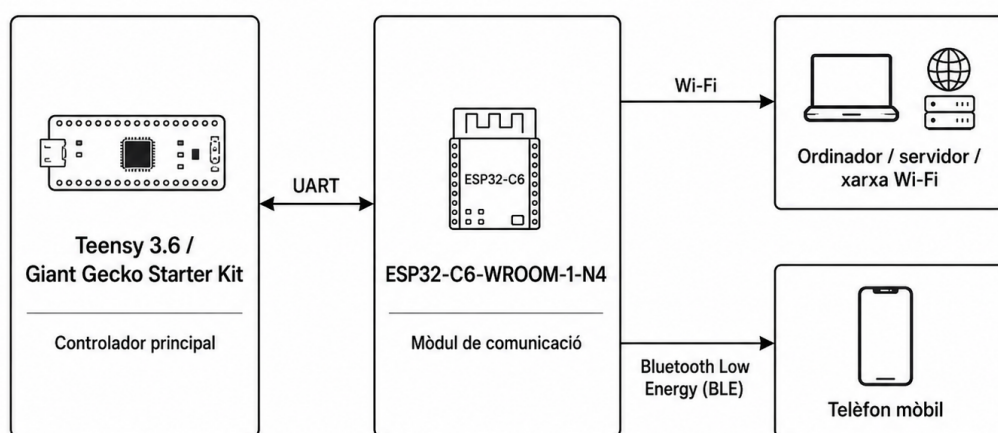
Aquesta distribució permet que el microcontrolador principal s'encarregui de l'adquisició de dades, el processament de la informació i el control dels perifèrics. Paral·lelament, l'ESP32-C6 pot gestionar la connexió Wi-Fi, la comunicació Bluetooth Low Energy i la transmissió de dades a dispositius externs.

L'arquitectura contempla dos modes generals de funcionament:

- **Funcionament autònom:** l'ESP32-C6 executa directament el firmware de l'aplicació i gestiona tant el processament com les comunicacions.
- **Funcionament com a mòdul de comunicació:** la Teensy 3.6 o la Giant Gecko executen l'aplicació principal i controlen l'ESP32-C6 mitjançant una interfície UART. Aquest és el mode d'ús previst principalment per a la placa docent, ja que permet mantenir la plataforma principal com a responsable de l'aplicació i reservar l'ESP32-C6 per a la gestió de les comunicacions sense fils.

Aquesta flexibilitat és especialment útil en un entorn docent. En les activitats inicials, l'ESP32-C6 es pot programar de manera independent. En pràctiques més avançades, es pot estudiar la comunicació entre microcontroladors, la distribució de tasques i el control d'un mòdul extern mitjançant una interfície sèrie.

Arquitectura general del sistema



Esquema 1: Arquitectura general del sistema de comunicació de la placa docent.

Font: Elaboració pròpia, amb assistència d'eines d'intel·ligència artificial per a la millora visual del diagrama.

4.1.1 Funció de l'ESP32-C6 dins de la placa docent

La funció principal de l'ESP32-C6-WROOM-1-N8 és incorporar connectivitat sense fils a la placa docent. El mòdul permet connectar la plataforma a una xarxa Wi-Fi existent, crear un punt d'accés propi, comunicar-se amb dispositius Bluetooth Low Energy i transmetre dades mitjançant connexions de xarxa.

En mode Wi-Fi Station, l'ESP32-C6 actua com a client d'una xarxa existent i obté una adreça IP. Aquesta configuració permet comunicar la placa amb ordinadors, servidors o altres dispositius connectats a la mateixa xarxa.

En mode SoftAP, el mòdul crea una xarxa Wi-Fi pròpia. Aquesta opció permet establir una comunicació directa amb un ordinador o un telèfon mòbil sense dependre de la infraestructura de xarxa del laboratori.

Mitjançant Bluetooth Low Energy, l'ESP32-C6 pot anunciar la seva presència, acceptar connexions i intercanviar informació a través de serveis i característiques GATT.

Finalment, el firmware ESP-AT permet utilitzar l'ESP32-C6 com a mòdul de comunicació controlat externament. En aquest mode, un altre microcontrolador pot enviar comandes AT per configurar el dispositiu, connectar-lo a una xarxa Wi-Fi, consultar-ne l'adreça IP, obrir connexions TCP i transmetre dades.

Les funcionalitats principals implementades són:

- Connexió a una xarxa Wi-Fi en mode Station.
- Creació d'una xarxa pròpia en mode SoftAP.
- Comunicació Bluetooth Low Energy.
- Intercanvi de dades mitjançant un servei GATT.
- Funcionament com a mòdul de comunicació mitjançant ESP-AT.
- Establiment de connexions TCP.
- Comunicació amb altres microcontroladors mitjançant UART.

4.1.2 Relació amb la Teensy 3.6 i la Giant Gecko

La Teensy 3.6 i la Giant Gecko Starter Kit poden actuar com a controladors principals de la placa docent. En aquesta configuració, l'ESP32-C6 funciona com un element auxiliar especialitzat en les comunicacions sense fils.

La interfície per establir aquesta comunicació és la UART. La connexió requereix dues línies de dades i una referència comuna de massa:

- El TX de l'ESP32-C6 es connecta al RX del microcontrolador principal.
- El RX de l'ESP32-C6 es connecta al TX del microcontrolador principal.
- Els dos dispositius comparteixen la mateixa connexió de GND.

A través d'aquesta interfície, la Teensy o la Giant Gecko poden enviar ordres, dades adquirides pels sensors o paràmetres de configuració. L'ESP32-C6 pot respondre amb informació sobre l'estat de la connexió, les dades rebudes o el resultat de les operacions sol·licitades.

Quan s'utilitza ESP-AT, el controlador principal no ha d'implementar directament tota la gestió de la comunicació Wi-Fi. En lloc d'això, envia comandes de text a l'ESP32-C6, que interpreta les ordres i executa les operacions corresponents.

Tal com es representa a l'Esquema 1, la Teensy 3.6 o la Giant Gecko Starter Kit es comuniquen amb l'ESP32-C6 mitjançant una interfície UART bidireccional, mentre que el mòdul gestiona les comunicacions Wi-Fi i Bluetooth Low Energy amb els dispositius externs.

4.2 Integració hardware de l'ESP32-C6-WROOM-1

La integració hardware ha requerit analitzar les característiques elèctriques i funcionals del mòdul, identificar els senyals necessaris i definir-ne les connexions amb la resta de la placa docent.

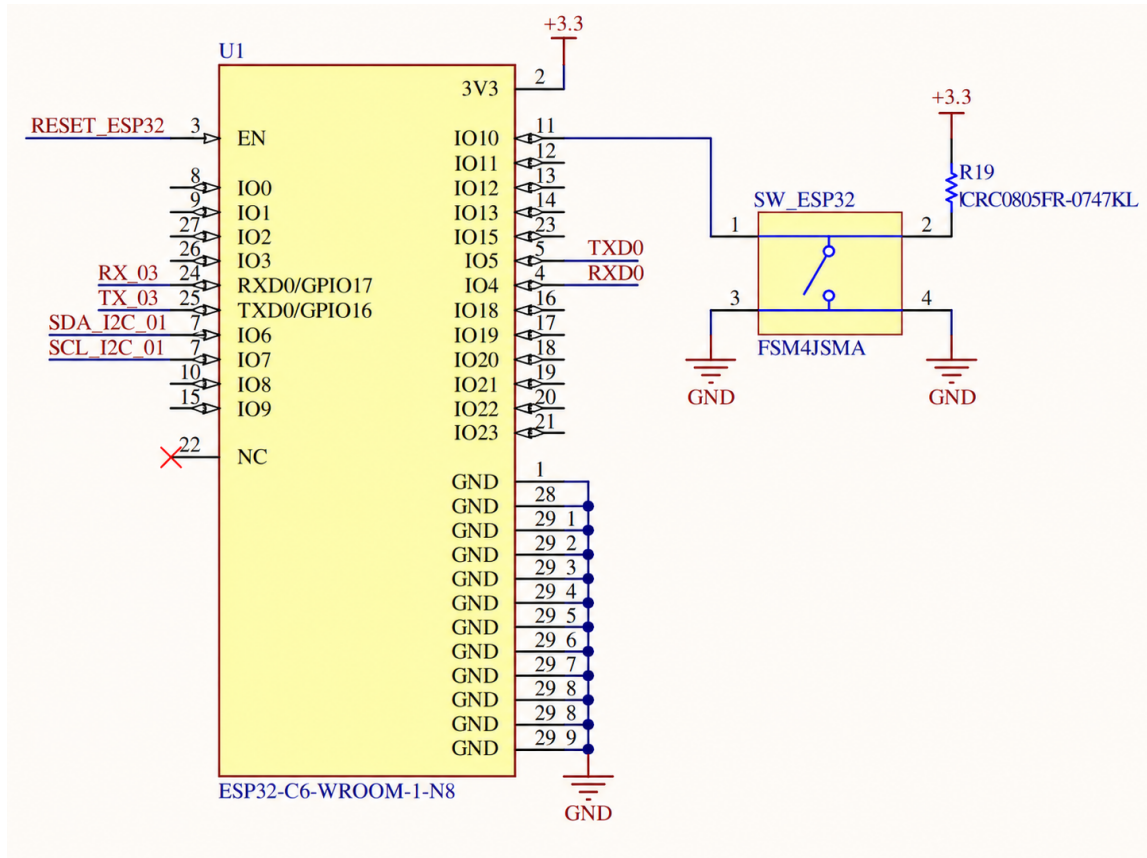
S'han prioritzat els senyals indispensables per alimentar, reiniciar, programar i comunicar el dispositiu. També s'han tingut en compte les restriccions associades als pins de configuració d'arrencada i a la zona lliure necessària al voltant de l'antena integrada.

Els elements principals incorporats al disseny són:

- Alimentació a 3,3 V.
- Connexions de massa.
- Senyal d'habilitació i reset.
- Senyals de configuració d'arrencada.
- Interfície UART0.
- Interfície USB Serial/JTAG.

- Connector de programació.
- GPIO disponibles per a possibles ampliacions.

L'Esquema 2 mostra de manera conjunta les connexions principals incorporades al bloc de l'ESP32-C6 dins del disseny de la placa docent.



Esquema 2: Connexions principals de l'ESP32-C6-WROOM-1-N8 dins de l'esquemàtic de la nova placa docent.

Font: Elaboració pròpia mitjançant Altium Designer.

4.2.1 Anàlisi del pinout

L'ESP32-C6-WROOM-1 disposa de 29 pins en total: 28 contactes laterals i un pad de massa exposat a la part inferior [14]. Aquests contactes inclouen les connexions d'alimentació, els senyals de control, els GPIO i les diferents interfícies de comunicació.

Per definir correctament la integració, s'ha analitzat per a cada pin:

- El número físic.

- El nom del senyal.
- El tipus elèctric.
- Les funcions alternatives disponibles.
- L'ús previst dins de la placa docent.
- La xarxa assignada a l'esquemàtic.

Aquest estudi ha permès seleccionar els senyals necessaris i evitar incompatibilitats amb els pins relacionats amb l'arrencada, la interfície USB o altres funcions internes del dispositiu.

La Figura 5 mostra la distribució física i la numeració dels contactes. La Taula 2 recull el nom, el tipus i les funcions disponibles per a cadascun dels pins.

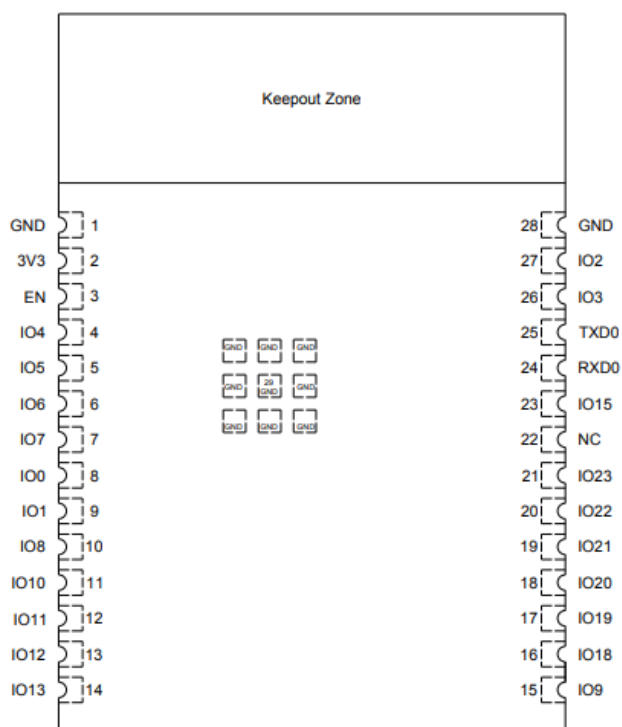


Figura 5: Distribució i numeració dels pins de l'ESP32-C6-WROOM-1.

Font: Datasheet oficial de l'ESP32-C6-WROOM-1 d'Espressif Systems [14].

Name	No.	Type ¹	Function
GND	1	P	Ground
3V3	2	P	Power supply
EN	3	I	High: on, enables the chip. Low: off, the chip powers off. Note: Do not leave the EN pin floating.
IO4	4	I/O/T	MTMS, GPIO4, LP_GPIO4, LP_UART_RXD, ADC1_CH4, FSPIHD
IO5	5	I/O/T	MTDI, GPIO5, LP_GPIO5, LP_UART_TXD, ADC1_CH5, FSPIWP
IO6	6	I/O/T	MTCK, GPIO6, LP_GPIO6, LP_I2C_SDA, ADC1_CH6, FSPICLK
IO7	7	I/O/T	MTDO, GPIO7, LP_GPIO7, LP_I2C_SCL, FSPID
IO0	8	I/O/T	GPIO0, XTAL_32K_P, LP_GPIO0, LP_UART_DTRN, ADC1_CH0
IO1	9	I/O/T	GPIO1, XTAL_32K_N, LP_GPIO1, LP_UART_DSRN, ADC1_CH1
IO8	10	I/O/T	GPIO8
IO10	11	I/O/T	GPIO10
IO11	12	I/O/T	GPIO11
IO12	13	I/O/T	GPIO12, USB_D-
IO13	14	I/O/T	GPIO13, USB_D+
IO9	15	I/O/T	GPIO9
IO18	16	I/O/T	GPIO18, SDIO_CMD, FSPICS2
IO19	17	I/O/T	GPIO19, SDIO_CLK, FSPICS3
IO20	18	I/O/T	GPIO20, SDIO_DATA0, FSPICS4
IO21	19	I/O/T	GPIO21, SDIO_DATA1, FSPICS5
IO22	20	I/O/T	GPIO22, SDIO_DATA2
IO23	21	I/O/T	GPIO23, SDIO_DATA3
NC	22	—	NC
IO15	23	I/O/T	GPIO15
RXD0	24	I/O/T	UORXD, GPIO17, FSPICS1
TXD0	25	I/O/T	UOTXD, GPIO16, FSPICS0
IO3	26	I/O/T	GPIO3, LP_GPIO3, LP_UART_CTSN, ADC1_CH3
IO2	27	I/O/T	GPIO2, LP_GPIO2, LP_UART_RTSN, ADC1_CH2, FSPIQ
GND	28	P	Ground
EPAD	29	P	Ground

Taula 2: Nom, numeració, tipus i funcions dels pins de l'ESP32-C6-WROOM-1.

Font: Datasheet oficial de l'ESP32-C6-WROOM-1 d'Espressif Systems [14].

4.2.2 Alimentació i connexions de massa

El mòdul s'alimenta mitjançant el pin 3V3, connectat a la línia de 3,3 V de la placa. Aquesta alimentació ha de mantenir-se estable, especialment durant les transmissions sense fils, ja que el consum instantani pot augmentar quan s'utilitzen les ràdios Wi-Fi o Bluetooth Low Energy.

Els pins laterals GND i el pad exposat inferior s'han connectat al pla de massa de la PCB. Aquesta connexió proporciona una referència elèctrica comuna, contribueix a la integritat dels senyals i facilita la dissipació tèrmica del mòdul.

Les connexions de massa utilitzades són:

- Pin 1: GND.
- Pin 28: GND.
- Pad exposat inferior: connexió al pla de massa.

Tal com es mostra a l'Esquema 2, el pin 3V3 es connecta a la línia d'alimentació de 3,3 V, mentre que els pins GND i el pad exposat es connecten a la massa comuna de la placa.

4.2.3 Senyals de reset i boot

El senyal de reset de l'ESP32-C6 s'ha implementat a través del pin EN del mòdul, connectat a la xarxa `RESET_ESP32`. Aquesta xarxa incorpora un pulsador de reset, identificat com a `SW_ESP32`, i una resistència de polarització R19 connectada a 3,3 V. D'aquesta manera, el senyal es manté en nivell alt durant el funcionament normal i, en prémer el pulsador, es porta temporalment a nivell baix per reiniciar el mòdul [14].

Pel que fa als senyals de boot, els pins GPIO8 i GPIO9 formen part dels senyals relacionats amb l'arrencada de l'ESP32-C6. Aquests pins no s'han connectat a cap senyal externa de la placa docent ni s'han assignat a cap funcionalitat addicional. Aquesta decisió evita interferir amb el comportament d'arrencada del mòdul i deixa aquests pins reservats.

4.2.4 Interfície UART

La comunicació sèrie entre l'ESP32-C6 i les plataformes principals s'ha definit mitjançant la interfície UART0 [17].

Els senyals utilitzats són:

- GPIO16 com a U0TXD.
- GPIO17 com a U0RXD.

GPIO16 transmet les dades generades per l'ESP32-C6, mentre que GPIO17 rep les dades enviades pel dispositiu extern. La connexió amb el microcontrolador principal s'ha de realitzar de manera creuada, connectant la sortida TX d'un dispositiu amb l'entrada RX de l'altre.

Aquesta interfície es pot utilitzar amb un protocol propi o amb el firmware ESP-AT. En aquest últim cas, la Teensy 3.6 o la Giant Gecko poden controlar el mòdul enviant comandes de text i processant les respostes rebudes.

A l'Esquema 2 es poden identificar les línies TX 03 i RX 03, associades respectivament a GPIO16 i GPIO17, així com el connector destinat a la programació i a la comunicació sèrie.

4.2.5 Interfície USB i programació

L'ESP32-C6 integra un controlador USB Serial/JTAG que permet connectar directament el xip a un ordinador sense necessitat d'incorporar obligatòriament un convertidor USB-UART extern [18]. No obstant això, en el disseny final de la placa docent no s'ha utilitzat la interfície USB nativa de l'ESP32-C6. La comunicació i la programació del mòdul es gestionen mitjançant la interfície UART0 i un convertidor USB-UART CP2102 integrat a la placa, compartit amb la resta del sistema. Per aquest motiu, els pins GPIO12 i GPIO13 no s'han connectat a un connector USB, sinó que queden disponibles com a GPIO de propòsit general per a futures ampliacions.

Cal diferenciar, doncs, els dos camins possibles de comunicació. En el cas de la USB nativa, les línies D+ i D- arribarien directament al controlador USB intern de l'ESP32-C6. En el cas adoptat, basat en el pont USB-UART, la comunicació USB de l'ordinador es converteix en senyals UART mitjançant el CP2102, que es connecten a la UART0 del mòdul (GPIO16 i GPIO17).

La ubicació de tots aquests pins es pot consultar a la Figura 5 i a la Taula 2.

4.2.6 Connexions incorporades a l'esquemàtic

A partir de l'anàlisi del mòdul i de l'esquemàtic final de la placa docent, s'han incorporat les connexions necessàries per alimentar-lo, reiniciar-lo, programar-lo i comunicar-lo amb la resta del sistema.

Els pins principals utilitzats són:

- Pin 1 i pins GND: connexió al pla de massa comú de la placa.
- Pin 2: alimentació de 3,3 V del mòdul.
- Pin 3: senyal EN / reset.
- Pin 6: GPIO6, línia SDA_I2C_01 del bus I2C.
- Pin 7: GPIO7, línia SCL_I2C_01 del bus I2C.

- Pin 24: GPIO17, recepció de la UART0 de comunicació, connectada a la línia RX_03.
- Pin 25: GPIO16, transmissió de la UART0 de comunicació, connectada a la línia TX_03.
- Pins GPIO4 i GPIO5: línies sèrie addicionals RXD0 i TXD0, destinades al connector de programació del mòdul.

A més de la UART principal de comunicació, l'esquemàtic incorpora una segona parella de línies sèrie, TXD0 i RXD0, associades als pins GPIO5 i GPIO4. Aquestes línies es destinen al connector de programació del mòdul i permeten disposar d'un accés sèrie addicional durant la configuració, depuració o càrrega de firmware.

Els pins relacionats amb la configuració d'arrencada del mòdul no s'han assignat a cap funcionalitat addicional de la placa docent. D'aquesta manera, s'evita interferir amb el procés de boot de l'ESP32-C6.

4.3 Entorns i eines de desenvolupament

Per implementar i validar les diferents funcionalitats s'han utilitzat diversos entorns i eines de desenvolupament. Cada eina s'ha destinat a una part concreta del projecte.

En aquest apartat se'n descriu únicament la funció dins del procés de desenvolupament. Els passos d'instal·lació, configuració i ús es documenten de manera completa a l'Annex A.

4.3.1 ESP-IDF

ESP-IDF és l'entorn oficial de desenvolupament d'Espressif per als dispositius de la família ESP32 [10]. En aquest projecte s'ha utilitzat per configurar, compilar, programar i monitorar les aplicacions executades per l'ESP32-C6.

L'entorn proporciona biblioteques, eines de configuració i exemples oficials relacionats amb Wi-Fi, Bluetooth Low Energy, GPIO i altres perifèrics [19, 20].

Les funcions principals utilitzades han estat:

- Selecció i configuració de l'ESP32-C6 com a dispositiu de destinació.
- Compilació dels projectes.
- Programació de la memòria flash.
- Monitorització dels missatges del dispositiu.

- Configuració dels paràmetres de Wi-Fi.
- Desenvolupament i modificació de l'aplicació BLE.

4.3.2 Simplicity Studio

Simplicity Studio és l'entorn de desenvolupament de Silicon Labs utilitzat per treballar amb la Giant Gecko Starter Kit [6, 5].

L'eina permet detectar la placa, crear o importar projectes, configurar perifèrics, compilar el codi, programar el microcontrolador i executar sessions de depuració.

Durant el projecte s'ha utilitzat per realitzar proves bàsiques de GPIO, LED i polsadors. Aquestes comprovacions permeten validar el funcionament de la plataforma i preparar-ne la futura comunicació amb l'ESP32-C6.

4.3.3 nRF Connect

nRF Connect és una aplicació per a dispositius mòbils que s'ha utilitzat per verificar la comunicació Bluetooth Low Energy [21].

L'aplicació permet:

- Detectar dispositius BLE propers.
- Establir una connexió amb l'ESP32-C6.
- Consultar els serveis i les característiques GATT.
- Llegir i escriure valors.
- Activar notificacions o indicacions.

El seu ús ha permès validar el servidor GATT sense haver de desenvolupar una aplicació mòbil específica.

4.3.4 ESP-AT i terminals sèrie

ESP-AT és un firmware oficial d'Espressif que permet controlar les funcions de comunicació de l'ESP32 mitjançant comandes AT [22, 23].

En aquest projecte s'ha utilitzat per convertir l'ESP32-C6 en un mòdul de comunicació controlable a través d'una interfície UART. Els terminals sèrie han permès visualitzar els missatges d'arrencada i enviar les comandes corresponents.

Les instruccions completes de descàrrega, instal·lació, compilació, programació i ús d'ESP-AT es recullen a l'Annex A.

4.3.5 Altium Designer

Altium Designer s'ha utilitzat com a eina de suport per revisar la integració hardware de l'ESP32-C6 dins del disseny general de la placa.

El treball realitzat amb aquesta eina s'ha centrat en:

- Revisar el símbol i el footprint del mòdul.
- Comprovar les connexions d'alimentació i massa.
- Revisar els senyals de reset i configuració d'arrencada.
- Verificar les connexions UART i USB.
- Coordinar la integració amb el projecte complementari encarregat del layout.

4.4 Desenvolupament del firmware

El desenvolupament del firmware s'ha organitzat de manera progressiva. En primer lloc, s'han preparat programes bàsics per verificar l'entorn i el procés de programació. Posteriorment, s'han implementat les funcions Wi-Fi i Bluetooth Low Energy. Finalment, s'ha adaptat el firmware ESP-AT per comprovar el funcionament del dispositiu com a mòdul de comunicació.

En aquest capítol es descriu l'objectiu i el funcionament general de cada desenvolupament. Els codis complets, les comandes i els procediments necessaris per reproduir-los s'inclouen a l'Annex A. Els resultats obtinguts es presenten al capítol dedicat al pla de proves.

4.4.1 Proves bàsiques de funcionament

La primera fase ha consistit a comprovar que l'ESP32-C6 es podia configurar, compilar, programar i monitorar correctament.

S'ha utilitzat un projecte bàsic de tipus Hello World [24] per verificar:

- La selecció correcta de l'ESP32-C6.
- El funcionament del compilador.
- La comunicació entre l'ordinador i la placa.
- La programació de la memòria flash.
- La visualització de missatges per consola.

També s'han preparat proves de GPIO per comprovar el control de les entrades i sortides digitals i el funcionament dels LED disponibles.

4.4.2 Connexió Wi-Fi en mode Station

La funcionalitat Wi-Fi Station permet que l'ESP32-C6 es connecti com a client a una xarxa Wi-Fi existent. Dins del projecte, aquest mode s'ha utilitzat per validar la capacitat del mòdul d'integrar-se en una infraestructura de xarxa real i establir una base de comunicació IP amb altres dispositius.

El desenvolupament d'aquesta prova ha consistit a preparar un firmware capaç d'inicialitzar el subsistema Wi-Fi, configurar el dispositiu en mode Station i gestionar el procés de connexió amb el punt d'accés. El programa controla els diferents estats de la connexió, incloent-hi l'inici del servei Wi-Fi, l'autenticació amb la xarxa, la gestió de possibles desconnexions i l'obtenció d'una adreça IP mitjançant DHCP.

Durant l'execució, l'estat del procés es mostra pel monitor sèrie, fet que permet verificar de manera directa si el mòdul s'ha connectat correctament i si ha obtingut una configuració de xarxa vàlida. Aquesta informació ha estat utilitzada per comprovar que l'ESP32-C6 queda operatiu dins de la xarxa i preparat per establir comunicacions posteriors mitjançant protocols basats en TCP/IP.

Aquest mode ofereix una funcionalitat essencial per a la placa docent, ja que permet connectar-la a una xarxa Wi-Fi existent i comunicar-la amb ordinadors, servidors o altres dispositius connectats a la mateixa xarxa. D'aquesta manera, es poden plantejar aplicacions docents relacionades amb l'enviament de dades de sensors, la recepció d'ordres externes o la supervisió remota del sistema.

4.4.3 Configuració Wi-Fi en mode SoftAP

El mode SoftAP permet que l'ESP32-C6 actuï com a punt d'accés i generi una xarxa Wi-Fi pròpia. En aquest treball, aquest mode s'ha utilitzat per validar la capacitat del mòdul

de crear una xarxa local independent, sense necessitat de disposar d'un encaminador o d'una infraestructura Wi-Fi externa.

El firmware preparat configura el dispositiu perquè inicialitzi el subsistema Wi-Fi, estableixi el mode SoftAP i generi una xarxa amb els paràmetres definits per a la prova. Aquesta configuració inclou el nom de la xarxa, les condicions d'accés, el canal de comunicació, el nombre màxim de dispositius clients i el mecanisme de seguretat utilitzat.

Un cop activat el punt d'accés, s'ha comprovat que la xarxa creada per l'ESP32-C6 apareixia disponible des d'un telèfon mòbil i que era possible establir-hi connexió correctament. També s'ha verificat que el dispositiu client rebia una adreça IP assignada pel servidor DHCP del mateix ESP32-C6, confirmant així el funcionament bàsic de la xarxa local creada pel mòdul.

Aquest mode ofereix una alternativa de comunicació especialment útil en entorns docents, ja que permet establir una connexió directa entre la placa i un dispositiu extern sense dependre d'una xarxa prèvia. Això facilita el desenvolupament de pràctiques en què un telèfon mòbil o un ordinador es connectin directament a la placa per consultar dades, enviar instruccions o comprovar el funcionament d'un sistema de comunicació local.

4.4.4 Comunicació Bluetooth Low Energy

La primera validació de Bluetooth Low Energy s'ha realitzat mitjançant un projecte basat en la pila NimBLE.

L'ESP32-C6 actua com a dispositiu perifèric, mentre que el telèfon mòbil actua com a dispositiu central. Aquesta configuració permet detectar el mòdul, establir-hi una connexió i consultar els serveis disponibles.

La implementació comprova les funcions bàsiques següents:

- Inicialització de la pila BLE.
- Inici del procés d'advertising.
- Detecció del dispositiu des del telèfon mòbil.
- Establiment de la connexió.
- Accés als serveis i característiques GATT.

4.4.5 Aplicació BLE amb servei GATT

A partir del projecte BLE inicial s'ha desenvolupat una aplicació capaç d'intercanviar missatges amb un telèfon mòbil.

L'ESP32-C6 actua com a servidor GATT i el telèfon com a client. S'ha configurat un servei amb una característica que admet operacions de lectura i escriptura.

Quan el telèfon envia un missatge, la funció associada a la característica rep les dades, n'emmagatzema el contingut i el mostra a través de la consola sèrie. Aquesta aplicació permet comprovar un intercanvi real d'informació i no únicament la detecció del dispositiu.

4.4.6 Funcionament com a mòdem mitjançant ESP-AT

En aquest mode, l'ESP32-C6 deixa d'executar l'aplicació principal i passa a funcionar com un mòdul de comunicació controlat externament. La idea és que el microcontrolador principal (la Teensy 3.6 o la Giant Gecko) no hagi d'implementar tota la pila Wi-Fi i TCP/IP, sinó que delegui aquestes tasques a l'ESP32-C6 enviant-li ordres de text senzilles a través de la UART.

Per aconseguir-ho s'ha utilitzat el firmware oficial ESP-AT [22, 23], que interpreta comandes AT rebudes per la UART i executa les operacions de comunicació corresponents. D'aquesta manera, l'ESP32-C6 ofereix al controlador principal un conjunt de funcions de xarxa accessibles mitjançant text, sense haver de programar directament les ràdios ni la gestió de connexions.

El flux de treball comprovat durant el projecte segueix l'ordre següent, i utilitza les comandes recollides a la documentació oficial d'Espressif [25, 26, 27]:

- **AT**: comprova que el mòdul respon correctament i que el port de comandes està ben configurat. La resposta esperada és OK.
- **AT+GMR**: consulta la versió del firmware ESP-AT instal·lat, fet que permet verificar que el binari programat és el correcte.
- **AT+CWMODE**: configura el mode Wi-Fi del dispositiu. En aquest cas s'ha seleccionat el mode Station, perquè el mòdul actuï com a client d'una xarxa existent.
- **AT+CWJAP**: connecta l'ESP32-C6 a una xarxa Wi-Fi indicant el nom de la xarxa (SSID) i la contrasenya. Un cop establerta la connexió, el mòdul obté una adreça IP del servidor DHCP de la xarxa.
- **AT+CIFSR**: consulta l'adreça IP assignada, cosa que confirma que la connexió a la xarxa s'ha completat correctament.
- **AT+CIPSTART**: obre una connexió TCP cap a un servidor remot, indicant el tipus de connexió, l'adreça IP del servidor i el port.
- **AT+CIPSEND**: indica el nombre de bytes que es volen transmetre i envia les dades a través de la connexió TCP oberta.

Aquesta seqüència demostra el conjunt complet de funcionalitats que l'ESP32-C6 posa a disposició del controlador principal en mode mòdem: comprovació de l'estat, configuració del Wi-Fi, connexió a una xarxa, obtenció d'adreça IP, obertura d'una connexió TCP i enviament de dades. Tot el procés es controla únicament mitjançant comandes de text per la UART, sense necessitat que el microcontrolador principal implementi la pila de comunicacions.

Per comprovar l'enviament de dades s'ha preparat un servidor TCP executat en un ordinador de la mateixa xarxa. Aquest servidor rep i mostra els missatges transmesos per l'ESP32-C6, fet que permet validar de punta a punta el funcionament del mòdul com a mòdem de comunicació. Els valors exactes utilitzats en cada comanda, així com els resultats obtinguts, es presenten al capítol dedicat al pla de proves, i el procediment complet es documenta a l'Annex A.

4.4.7 Adaptació de la UART per a les comandes AT

En la configuració inicial d'ESP-AT, els missatges de registre del sistema i el port de comandes utilitzaven interfícies diferents. El port destinat a les comandes AT estava assignat a UART1, amb GPIO7 com a TX, GPIO6 com a RX i GPIO5 i GPIO4 com a senyals de control de flux.

Durant les primeres proves, l'ordinador permetia visualitzar els missatges d'arrencada, però no enviar comandes AT al port corresponent sense utilitzar un adaptador USB-UART extern.

Com que no es disposava d'aquest adaptador i es volia controlar el mòdul mitjançant la interfície accessible, es va modificar el fitxer de paràmetres de fàbrica del projecte ESP-AT.

Els canvis aplicats van ser:

- Canvi del port de comandes de UART1 a UART0.
- Canvi del pin TX de GPIO7 a GPIO16.
- Canvi del pin RX de GPIO6 a GPIO17.
- Desactivació del senyal CTS.
- Desactivació del senyal RTS.

La configuració final utilitza UART0, GPIO16 com a TX i GPIO17 com a RX.

Després de modificar els paràmetres, el firmware es va recompilar i es va generar un nou fitxer binari adaptat a la configuració del sistema. Els valors exactes, el fitxer modificat i les comandes de compilació i programació es documenten pas a pas a l'Annex A.

Taula 3: Comparació entre la configuració original i la configuració final del port de comandes ESP-AT.

Paràmetre	Configuració original	Configuració final
Port de comandes	UART1	UART0
Pin TX	GPIO7	GPIO16
Pin RX	GPIO6	GPIO17
Pin CTS	GPIO5	Desactivat
Pin RTS	GPIO4	Desactivat
Control de flux	Hardware	No utilitzat
Línies necessàries	TX, RX, CTS, RTS i GND	TX, RX i GND

Font: Elaboració pròpia a partir de la configuració del projecte ESP-AT.

Capítol 5

Entregables

Aquest capítol presenta els materials generats durant el desenvolupament del treball i la manera com es posen a disposició del professorat i de futurs usuaris de la placa docent. El procés d'integració del mòdul i el desenvolupament de les diferents aplicacions s'han descrit al capítol anterior, mentre que la validació i els resultats obtinguts es presenten al capítol següent. Per aquest motiu, en aquest apartat no es tornen a explicar les característiques tècniques ni el funcionament detallat de cada prova.

Els principals resultats lliurats són els projectes de firmware desenvolupats, els fitxers de configuració necessaris, el firmware ESP-AT adaptat, els programes preparats per a la Giant Gecko Starter Kit i la documentació que permet reproduir el treball. El codi font es troba disponible mitjançant una carpeta compartida de materials, mentre que les instruccions detallades de configuració i ús es recullen al manual docent de l'Annex A.

5.1 Materials desenvolupats

Durant el projecte s'han preparat diferents aplicacions destinades a comprovar les funcionalitats de l'ESP32-C6-WROOM-1-N8 i de les plataformes relacionades amb la placa docent.

Entre aquests materials es troben els projectes bàsics utilitzats per verificar la programació i l'execució del mòdul, les aplicacions de connexió Wi-Fi en mode Station i SoftAP, i els projectes de comunicació Bluetooth Low Energy. També s'ha desenvolupat una aplicació basada en un servei GATT que permet intercanviar missatges entre un telèfon mòbil i l'ESP32-C6.

Un altre dels materials generats és el projecte ESP-AT adaptat a la configuració prevista per a la placa docent. Aquesta adaptació inclou la modificació del port de comandes i de les línies UART utilitzades.

També s'han preparat diferents projectes per a la Giant Gecko Starter Kit mitjançant Simplicity Studio. Aquests programes permeten comprovar el funcionament dels GPIO, els LED, els pulsadors i altres perifèrics de la plataforma, i constitueixen una base per a la seva futura comunicació amb l'ESP32-C6.

La documentació relacionada amb el disseny inclou la informació necessària per interpretar les connexions del mòdul dins de la placa, especialment els senyals d'alimentació, reset, boot, UART i USB. El desenvolupament complet de l'esquemàtic i del layout forma part del projecte complementari, de manera que en aquest treball es lliura principalment la informació vinculada amb la selecció, integració i validació de l'ESP32-C6.

5.2 Carpeta de materials del projecte

Els projectes, els fitxers de configuració i la documentació desenvolupats durant el treball es proporcionen mitjançant una carpeta compartida de Google Drive. Aquesta carpeta actua com a punt central d'accés als materials necessaris per reproduir les proves descrites a la memòria i continuar el desenvolupament de la placa docent.

El contingut s'ha organitzat en carpetes independents segons la plataforma i la funcionalitat corresponent. S'hi inclouen els projectes desenvolupats amb ESP-IDF, les aplicacions de comunicació Bluetooth Low Energy, els fitxers necessaris per compilar i programar el firmware ESP-AT, els projectes preparats amb Simplicity Studio per a la Giant Gecko Starter Kit i els scripts auxiliars utilitzats durant les proves.

Cada projecte incorpora els fitxers font i de configuració necessaris, mentre que les instruccions detallades per preparar els entorns, adaptar els paràmetres i executar les aplicacions es recullen a l'Annex A. No s'inclouen les carpetes de compilació generades automàticament, ja que es poden regenerar a partir dels fitxers font i augmentarien innecessàriament la mida del material lliurat.

Els materials es poden consultar i descarregar mitjançant l'enllaç següent:

Accés a la carpeta de materials del projecte

Per garantir-ne l'accés, la carpeta s'ha configurat amb permisos de lectura per a qualsevol persona que disposi de l'enllaç.

5.3 Manual docent

Per complementar la carpeta de materials del projecte, s'ha preparat un manual docent destinat principalment al professorat que utilitzarà la placa en futures pràctiques. Aquest document s'inclou a l'Annex A i concentra tota la informació pràctica necessària per preparar els entorns i executar els projectes.

El manual explica com descarregar, instal·lar i configurar ESP-IDF, Simplicity Studio, nRF Connect, ESP-AT i els terminals sèrie utilitzats durant el desenvolupament. També descriu com identificar el port de comunicació, seleccionar l'ESP32-C6 com a dispositiu objectiu, compilar els projectes, programar les plaques i obrir els monitors corresponents.

A continuació, es presenta cada projecte de manera independent. Per a cada aplicació s'explica la seva finalitat, els fitxers principals que la formen, l'estructura general del programa i les funcions més importants. També s'indiquen els paràmetres que s'han de modificar, com les credencials de la xarxa Wi-Fi, els pins utilitzats o les opcions de configuració del firmware.

Les comandes utilitzades durant la compilació, la programació i l'execució també es documenten al manual. En lloc d'únicament presentar una llista d'ordres, s'explica què fa cadascuna, en quin directori s'ha d'executar i en quin moment del procés és necessària.

En el cas del firmware ESP-AT, el manual descriu la descàrrega del projecte, la modificació de la configuració UART i la compilació del firmware. També inclou les principals comandes AT utilitzades per comprovar el funcionament del mòdul, connectar-lo a una xarxa Wi-Fi, consultar-ne l'adreça IP, obrir una connexió TCP i enviar dades.

Pel que fa a les aplicacions Bluetooth Low Energy, el document explica l'estructura del servei GATT, les característiques utilitzades i el procés necessari per connectar-se al dispositiu mitjançant nRF Connect. També s'hi descriu com enviar un missatge des del telèfon mòbil i com comprovar-ne la recepció al monitor sèrie.

Finalment, el manual inclou els programes desenvolupats per a la Giant Gecko Starter Kit i les instruccions necessàries per importar-los, compilar-los i programar-los mitjançant Simplicity Studio.

D'aquesta manera, la carpeta compartida proporciona els projectes i els fitxers originals, mentre que l'Annex A actua com una guia detallada per entendre'ls i utilitzar-los. Aquesta separació evita repetir el codi dins de la memòria principal i facilita que el professorat pugui reproduir o modificar els programes utilitzats.

Capítol 6

Pla de proves i resultats

Aquest capítol presenta les proves realitzades per comprovar el funcionament de les solucions desenvolupades i els resultats obtinguts durant el procés de validació. L'objectiu és verificar que les funcionalitats implementades proporcionen el comportament previst, sense tornar a descriure l'estructura interna dels programes.

Les comprovacions s'han dut a terme principalment sobre una placa de desenvolupament equipada amb un mòdul ESP32-C6-WROOM-1. Aquesta plataforma ha permès validar el firmware i les funcions de comunicació abans de disposar de la versió final de la placa docent.

Els procediments complets per preparar els entorns, configurar els projectes, compilar-los, programar-los i executar-los es recullen al manual docent de l'Annex A.

6.1 Metodologia de validació

La validació s'ha organitzat de manera progressiva. En primer lloc, s'ha comprovat que l'entorn de desenvolupament permetia detectar, compilar, programar i monitoritzar correctament l'ESP32-C6. Un cop verificat el funcionament bàsic del dispositiu, s'han provat les funcionalitats Wi-Fi i Bluetooth Low Energy. Finalment, s'ha validat el funcionament de l'ESP32-C6 com a mòdem mitjançant ESP-AT i la transmissió de dades a través d'una connexió TCP.

Per a cada prova s'ha establert una configuració inicial, una funcionalitat concreta que calia comprovar i un resultat esperat. La validació s'ha considerat correcta quan el comportament observat coincidia amb el previst i es podia verificar mitjançant el monitor sèrie, una aplicació externa o la resposta d'un altre dispositiu.

Les proves s'han agrupat en els blocs següents:

-
- Funcionament bàsic i programació de l'ESP32-C6.
 - Connectivitat Wi-Fi en mode Station i SoftAP.
 - Comunicació Bluetooth Low Energy.
 - Funcionament com a mòdem mitjançant ESP-AT.
 - Proves amb la Giant Gecko Starter Kit i estudi de compatibilitat amb la Teensy 3.6.

Taula 4: Resum general de les proves de validació realitzades.

Funcionalitat	Resultat esperat	Resultat obtingut	Estat
Compilació i programació	Compilar el projecte, programar la memòria flash i iniciar correctament l'aplicació.	El projecte s'ha compilat i programat correctament, i els missatges s'han visualitzat al monitor sèrie.	Superada
Control de GPIO	Configurar i controlar correctament les entrades i sortides digitals.	S'ha comprovat l'encesa i l'apagada dels LED segons el comportament definit al firmware.	Superada
Wi-Fi Station	Connectar l'ESP32-C6 a una xarxa existent i obtenir una adreça IP.	El mòdul s'ha connectat al punt d'accés i ha obtingut una adreça IP correctament.	Superada
Wi-Fi SoftAP	Crear una xarxa Wi-Fi pròpia i permetre la connexió d'un dispositiu extern.	La xarxa creada per l'ESP32-C6 s'ha detectat i s'hi ha pogut establir una connexió.	Superada
Connexió BLE	Detectar l'ESP32-C6 des de nRF Connect i establir-hi una connexió.	El dispositiu s'ha detectat, connectat i ha mostrat els serveis i característiques GATT.	Superada
Intercanvi GATT	Enviar un missatge des del telèfon mòbil i rebre'l a l'ESP32-C6.	El missatge enviat des de nRF Connect s'ha rebut i mostrat correctament al monitor sèrie.	Superada
Funcionament amb ESP-AT	Enviar comandes AT, connectar el mòdul a una xarxa i obrir una connexió TCP.	El firmware ha respost a les comandes, s'ha connectat a la xarxa i ha establert la connexió TCP.	Superada
Enviament de dades TCP	Transmetre un missatge des de l'ESP32-C6 fins a un servidor executat en un ordinador.	El servidor ha rebut i mostrat correctament el missatge enviat pel mòdul.	Superada
Proves amb Giant Gecko	Compilar i executar programes bàsics de GPIO, LED i polsadors.	Els programes s'han executat correctament i el comportament observat ha coincidit amb el previst.	Superada
Comunicació amb les plataformes principals	Establir una comunicació UART entre l'ESP32-C6 i la Teensy 3.6 o la Giant Gecko.	La comunicació ha quedat definida a nivell de disseny, pendent de validació conjunta sobre la PCB final.	Pendent fins que arribi la PCB definitiva.

Font: Elaboració pròpia a partir dels resultats obtinguts durant les proves.

6.2 Proves bàsiques del mòdul

Abans de validar les funcions de comunicació, s'ha comprovat que l'ESP32-C6 es podia programar i executar correctament. Aquestes primeres proves permeten descartar problemes relacionats amb l'entorn de desenvolupament, el port de comunicació, la memòria flash o el mateix hardware utilitzat.

Durant les primeres comprovacions es va detectar una incidència amb una de les tres plaques de desenvolupament adquirides. Des del primer moment, aquesta unitat presentava un sobreescalfament fora del normal i no era reconeguda correctament per l'ordinador, ja que el port de comunicació no apareixia disponible.

Inicialment, es van revisar la connexió USB, els controladors instal·lats i la configuració del port. Com que el problema es mantenia, es va substituir la placa per una altra unitat del mateix conjunt. Amb la nova placa, l'ordinador va detectar correctament el port de comunicació i es van poder realitzar les operacions de programació i monitorització sense incidències.

6.2.1 Compilació, programació i monitorització

La primera comprovació ha consistit a configurar un projecte per a l'ESP32-C6 i compilar-lo mitjançant ESP-IDF. El procés de compilació s'ha completat sense errors i ha generat els fitxers necessaris per programar el dispositiu.

Posteriorment, el firmware s'ha transferit a la memòria flash de la placa. Un cop finalitzada la programació, s'ha iniciat el monitor sèrie i s'han visualitzat els missatges d'arrencada del sistema.

Aquest resultat confirma que l'entorn de desenvolupament estava correctament configurat, que la placa era detectada per l'ordinador i que el procés de compilació, programació i monitorització funcionava adequadament.

```
Hard resetting via RTS pin...
Executing action: monitor
Running idf_monitor in directory C:\esp32\projects\hello_world_esp32c6

Hello world!
This is esp32c6 chip with 1 CPU core(s), WiFi/BLE, 802.15.4 (Zigbee/Thread), silicon revision v0.2, 2MB external flash
Minimum free heap size: 480040 bytes
Restarting in 10 seconds...
```

Figura 6: Programació de l'ESP32-C6, obertura del monitor sèrie i execució correcta del projecte Hello World.

Font: Elaboració pròpia a partir del terminal d'ESP-IDF.

6.2.2 Proves de GPIO

Les proves de GPIO s'han utilitzat per comprovar la configuració i el control de les entrades i sortides digitals.

En el cas de l'ESP32-C6, s'ha executat un programa bàsic que permetia controlar l'estat dels LED connectats als GPIO seleccionats. El firmware modificava el nivell lògic de les sortides, fent possible encendre i apagar els LED de manera controlada.

Durant l'execució, els LED responien als canvis programats, alternant entre l'estat encès i apagat segons la seqüència definida al codi. Aquest comportament confirma que els pins es podien configurar correctament com a sortides digitals i que el programa era capaç de controlar-ne l'estat.

Aquesta prova ha permès verificar el funcionament bàsic dels GPIO abans d'utilitzar-los en aplicacions més complexes o assignar-los a altres funcions del sistema.

```
I (233) example: Example configured to blink addressable LED!  
I (243) example: Turning the LED OFF!  
I (1243) example: Turning the LED ON!
```

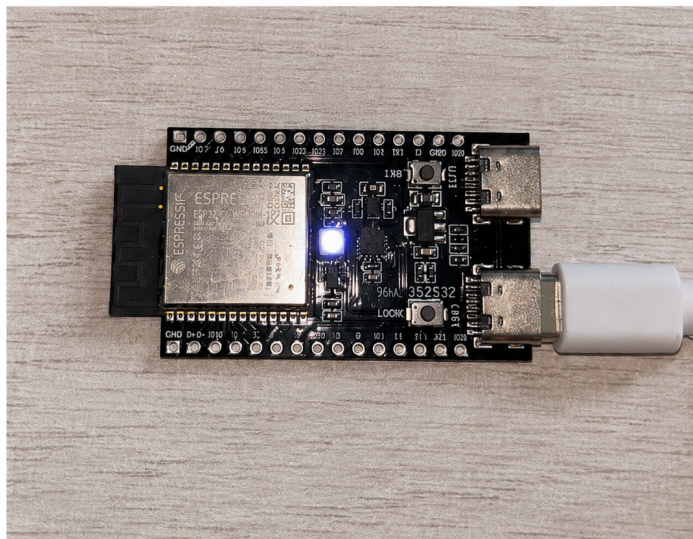


Figura 7: Execució de la prova de GPIO mitjançant el control del LED integrat de l'ESP32-C6.

Font: Elaboració pròpia a partir del terminal d'ESP-IDF i de la prova realitzada sobre la placa.

6.3 Validació de la connectivitat Wi-Fi

La connectivitat Wi-Fi s'ha comprovat en els dos modes principals previstos per a la placa docent. En primer lloc, s'ha validat la connexió de l'ESP32-C6 a una xarxa existent. Posteriorment, s'ha configurat el mòdul perquè generés una xarxa pròpia.

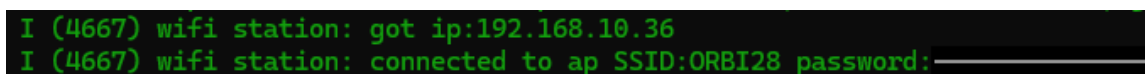
6.3.1 Resultats de la prova Wi-Fi Station

En aquesta prova, l'ESP32-C6 s'ha configurat com a estació per connectar-se a un punt d'accés existent.

Després d'iniciar el firmware, el monitor sèrie ha mostrat la inicialització del subsistema Wi-Fi i els intents de connexió amb la xarxa configurada. Un cop completada l'autenticació, el dispositiu ha rebut una adreça IP.

L'obtenció d'aquesta adreça confirma que el mòdul s'ha associat correctament al punt d'accés i que pot comunicar-se amb altres dispositius connectats a la mateixa xarxa.

També s'ha observat que, en cas de pèrdua temporal de la connexió, l'aplicació intenta restablir-la mitjançant el mecanisme de reconexió incorporat al projecte.



```
I (4667) wifi station: got ip:192.168.10.36
I (4667) wifi station: connected to ap SSID:ORBI28 password:_____
```

Figura 8: Connexió de l'ESP32-C6 a una xarxa Wi-Fi en mode Station i obtenció correcta d'una adreça IP.

Font: Elaboració pròpia a partir del monitor sèrie d'ESP-IDF.

6.3.2 Resultats de la prova Wi-Fi SoftAP

La prova SoftAP ha consistit a configurar l'ESP32-C6 com a punt d'accés.

Després d'iniciar el firmware, la xarxa creada pel mòdul ha aparegut entre les xarxes disponibles d'un telèfon mòbil. La connexió s'ha pogut establir utilitzant el nom i la contrasenya definits a la configuració del projecte.

El monitor sèrie ha registrat la connexió del dispositiu extern i ha mostrat la informació associada al nou client. Aquest resultat demostra que l'ESP32-C6 pot generar una xarxa local pròpia sense dependre d'un encaminador extern.

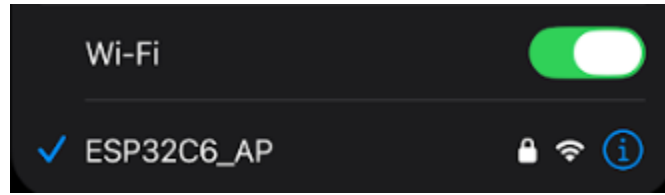


Figura 9: Connexió del telèfon mòbil al punt d'accés Wi-Fi creat per l'ESP32-C6.

Font: elaboració pròpia a partir de la configuració Wi-Fi del telèfon mòbil.

Tal com es mostra a la Figura 9, el telèfon mòbil detecta la xarxa creada i s'hi connecta correctament. Aquesta evidència complementa els missatges del monitor sèrie i confirma el funcionament del mòdul com a punt d'accés Wi-Fi.

```
I (738) wifi:mode : softAP (10:51:db:0d:24:e1)
I (838) wifi:Total power save buffer number: 16
I (838) wifi:Init max length of beacon: 752/752
I (838) wifi:Init max length of beacon: 752/752
I (838) esp_netif_lwip: DHCP server started on interface WIFI_AP_DEF with IP: 192.168.4.1
I (848) wifi softAP: wifi_init_softap finished. SSID:ESP32C6_AP password:12345678 channel:1
I (848) main_task: Returned from app_main()
I (15418) wifi:new:<1,0>, old:<1,1>, ap:<1,0>, sta:<255,255>, prof:1, snd_ch_cfg:0x0
I (15418) wifi:(trc)phytype:CBW20-SGI, snr:56, maxRate:0, highestRateIdx:0
W (15418) wifi:(trc)band:2G, phymode:3, highestRateIdx:0, lowestRateIdx:11, dataSchedTableSize:14
I (15428) wifi:(trc)band:2G, rate(S-MCS7, rateIdx:0), ampdn(rate:S-MCS7, schedIdx(0, stop:8)), snr:56, ampdnState:wait operational
I (15438) wifi:ifidx:1, rssi:-42, nf:-98, phytype(0x3, CBW20-SGI), phymode(0x3, 11bgn), max_rate:0, he:0, vht:0, ht:1
I (15448) wifi:(ht)max.RxAMPDULenExponent:3(65535 bytes), MMSS:6(8 us)
I (15458) wifi:station: 4e:2a:29:cd:29:17 join, AID=1, bgn, 20
I (15488) wifi softAP: station 4e:2a:29:cd:29:17 join, AID=1
I (16618) esp_netif_lwip: DHCP server assigned IP to a client, IP is: 192.168.4.2
```

Figura 10: Creació d'un punt d'accés Wi-Fi amb l'ESP32-C6 i connexió d'un dispositiu client, al qual s'assigna una adreça IP mitjançant DHCP.

Font: Elaboració pròpia a partir del monitor sèrie d'ESP-IDF.

Tal com es mostra a la Figura 10, l'ESP32-C6 inicia correctament el mode SoftAP amb l'adreça IP local 192.168.4.1. Posteriorment, el monitor sèrie registra la connexió d'un dispositiu client i mostra que el servidor DHCP li assigna l'adreça IP 192.168.4.2. Aquestes línies confirmen tant la creació del punt d'accés com la connexió correcta del telèfon mòbil.

6.4 Validació de Bluetooth Low Energy

La comunicació Bluetooth Low Energy s'ha comprovat mitjançant l'aplicació nRF Connect. Aquesta eina ha permès detectar l'ESP32-C6, establir una connexió i interactuar amb els serveis i les característiques proporcionats pel servidor GATT.

6.4.1 Detecció i connexió des de nRF Connect

Després de programar el projecte BLE, l'ESP32-C6 ha iniciat el procés d'advertising. En Bluetooth Low Energy, l'advertising és el mecanisme mitjançant el qual un dispositiu anuncia periòdicament la seva presència i permet que altres equips propers el puguin detectar.

Durant l'escaneig realitzat amb nRF Connect, el dispositiu ha aparegut entre els perifèrics disponibles. Això confirma que l'ESP32-C6 estava enviant correctament els paquets d'advertising i que podia ser identificat des del telèfon mòbil.

Posteriorment, s'ha establert la connexió i l'aplicació ha mostrat els serveis i les característiques definits al servidor GATT. Aquesta comprovació confirma que la pila Bluetooth Low Energy s'ha inicialitzat adequadament i que el mòdul pot actuar com a dispositiu perifèric.

```
I (435) NimBLE_BLE_PRPH: BLE Host Task Started
I (445) NimBLE: Device Address:
I (445) NimBLE: 10:51:db:0d:24:e2
I (9315) NimBLE: connection established; status=0
```

Figura 11: Inicialització de la pila BLE i establiment correcte de la connexió amb l'ESP32-C6.

Font: Elaboració pròpia a partir del monitor sèrie d'ESP-IDF.

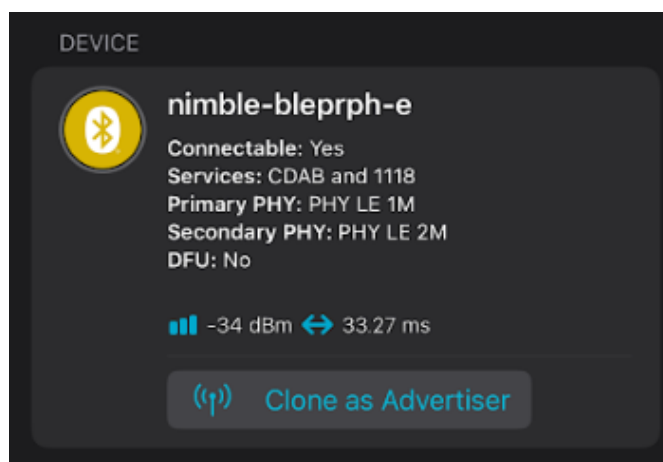


Figura 12: Detecció i connexió amb l'ESP32-C6 mitjançant l'aplicació nRF Connect.

Font: Elaboració pròpia a partir de l'aplicació nRF Connect.

Tal com es mostra a les Figures 11 i 12, l'ESP32-C6 inicia correctament la pila BLE, anuncia el dispositiu i permet establir una connexió des del telèfon mòbil. El missatge *connection established; status=0* confirma que la connexió s'ha completat sense errors.

6.4.2 Intercanvi de dades mitjançant GATT

Un cop establerta la connexió, s'ha utilitzat la característica GATT modificada per enviar un missatge des del telèfon mòbil cap a l'ESP32-C6.

El valor s'ha introduït des de nRF Connect i s'ha transmès mitjançant una operació d'escriptura. L'ESP32-C6 ha rebut les dades i n'ha mostrat el contingut al monitor sèrie.

Aquesta prova demostra que la comunicació Bluetooth Low Energy no es limita a la detecció i connexió del dispositiu, sinó que permet transmetre informació entre el telèfon mòbil i el mòdul.

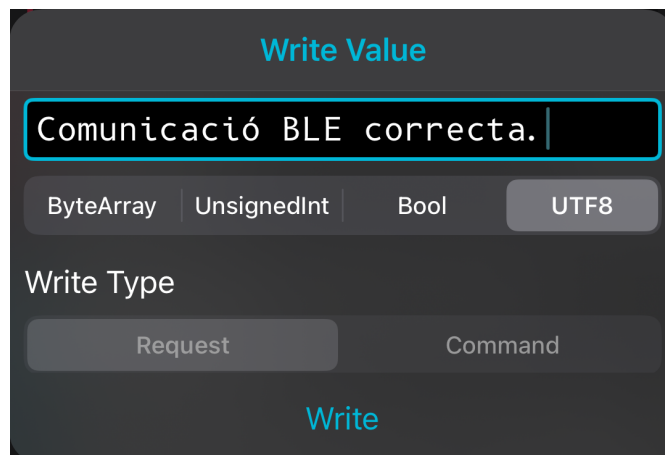


Figura 13: Enviament del missatge “Comunicació BLE correcta” des de l'aplicació nRF Connect mitjançant una operació d'escriptura GATT.

Font: Elaboració pròpia a partir de l'aplicació nRF Connect.

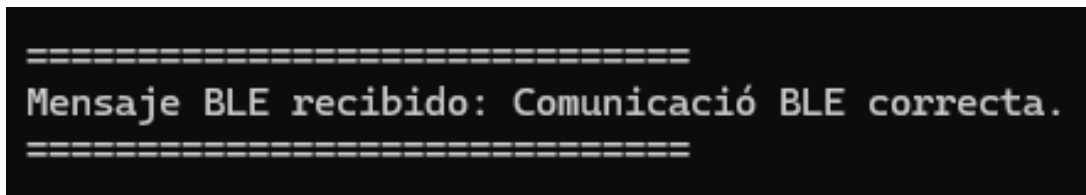


Figura 14: Recepció i visualització del missatge enviat mitjançant GATT al monitor sèrie de l'ESP32-C6.

Font: Elaboració pròpia a partir del monitor sèrie d'ESP-IDF.

Tal com es mostra a les Figures 13 i 14, el missatge s'ha introduït en format UTF-8 des de nRF Connect i s'ha enviat mitjançant una operació d'escriptura sobre la característica GATT. L'ESP32-C6 l'ha rebut correctament i n'ha mostrat el contingut al monitor sèrie, fet que confirma la comunicació de dades entre el telèfon mòbil i el mòdul.

6.5 Validació del funcionament com a mòdem

El funcionament com a mòdem s'ha validat mitjançant el firmware ESP-AT. En aquesta configuració, l'ESP32-C6 rep comandes a través d'una interfície sèrie i executa les operacions de comunicació corresponents.

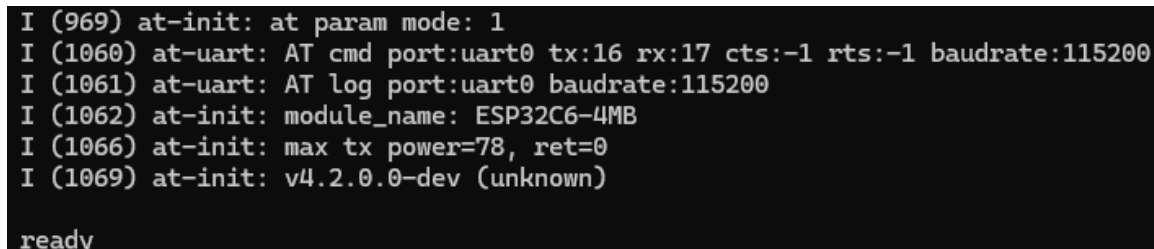
La validació s'ha dividit en diferents fases: programació del firmware, comprovació del port de comandes, connexió a una xarxa Wi-Fi i transmissió de dades a un servidor TCP.

6.5.1 Càrrega del firmware ESP-AT

Després de modificar la configuració de la UART i recompilar el projecte ESP-AT, s'ha generat un fitxer binari amb el firmware complet, preparat per ser gravat a la memòria flash de l'ESP32-C6.

La programació del fitxer s'ha completat correctament. Un cop reiniciat el dispositiu, el terminal sèrie ha mostrat els missatges d'inicialització d'ESP-AT, fet que confirma que el firmware s'ha carregat i s'ha executat adequadament.

Aquest resultat també permet comprovar que els canvis introduïts a la configuració de la UART són compatibles amb el mòdul i no impedeixen l'arrencada del sistema.



```
I (969) at-init: at param mode: 1
I (1060) at-uart: AT cmd port:uart0 tx:16 rx:17 cts:-1 rts:-1 baudrate:115200
I (1061) at-uart: AT log port:uart0 baudrate:115200
I (1062) at-init: module_name: ESP32C6-4MB
I (1066) at-init: max tx power=78, ret=0
I (1069) at-init: v4.2.0.0-dev (unknown)

ready
```

Figura 15: Inicialització correcta del firmware ESP-AT i configuració del port de comandes de l'ESP32-C6.

Font: Elaboració pròpia a partir del terminal sèrie.

Tal com es mostra a la Figura 15, el firmware ESP-AT s'inicia correctament després del reinici del dispositiu. El terminal mostra la configuració del port de comandes a la UART0, amb els pins GPIO16 i GPIO17 i una velocitat de 115200 bauds. Finalment, el missatge *ready* confirma que el mòdul està preparat per rebre comandes AT.

6.5.2 Configuració del port de comandes

En la configuració inicial del firmware ESP-AT, la UART0 s'utilitzava per mostrar els missatges de registre del sistema, mentre que la UART1 estava reservada per rebre les comandes AT. El port de comandes utilitzava GPIO7 com a TX, GPIO6 com a RX i els pins GPIO5 i GPIO4 per als senyals CTS i RTS del control de flux hardware.

Durant les primeres proves, l'ordinador només tenia accés al port associat als missatges de registre. Això permetia visualitzar correctament l'arrencada del firmware, però no introduir comandes AT, ja que aquestes es rebien a través de la UART1.

Per utilitzar la configuració original hauria estat necessari connectar un adaptador USB-UART extern als pins GPIO7 i GPIO6. Com que no es disposava d'aquest adaptador i es volia efectuar la prova directament des de l'ordinador, es va modificar la configuració del firmware.

Tal com es mostra a la Taula 3, el port de comandes es va traslladar de la UART1 a la UART0. Els pins de transmissió i recepció es van canviar de GPIO7 i GPIO6 a GPIO16 i GPIO17, respectivament. També es van desactivar els senyals CTS i RTS, de manera que la comunicació només requereix les línies TX, RX i GND.

Després de recompilar i programar novament el firmware, el terminal sèrie va mostrar la nova configuració del port de comandes. Aquesta configuració va permetre enviar comandes AT des de l'ordinador i obtenir les respostes corresponents, fet que confirma que l'adaptació de la UART es va aplicar correctament.

6.5.3 Resposta a les comandes AT

La primera ordre enviada al mòdul ha estat la comanda **AT**. El dispositiu ha retornat la resposta **OK**, fet que confirma que la comunicació sèrie funciona correctament i que el firmware interpreta les ordres rebudes.

A continuació, s'ha utilitzat la comanda **AT+GMR** per consultar la informació de versió. El terminal ha mostrat les dades corresponents al firmware ESP-AT instal·lat i ha finalitzat novament amb la resposta **OK**.

Aquestes dues comandes han permès confirmar el funcionament bidireccional de la interfície abans de continuar amb les operacions de xarxa.

```
ready
AT

OK
AT+GMR
AT version:4.2.0.0-dev(01b2e1d - ESP32C6 - May  8 2026 09:05:10)
SDK version:v5.4.4-dirty
compile time(14544e41):Jun  2 2026 18:52:24
Bin version:v4.2.0.0-dev(ESP32C6-4MB)

OK
```

Figura 16: Resposta de l'ESP32-C6 a les comandes AT i AT+GMR mitjançant el firmware ESP-AT.

Font: Elaboració pròpia a partir del terminal sèrie.

Tal com es mostra a la Figura 16, la comanda *AT* retorna la resposta *OK*, mentre que la comanda *AT+GMR* mostra la versió del firmware instal·lat. Aquest resultat confirma que el port de comandes està configurat correctament i que el mòdul està preparat per executar les operacions de xarxa posteriors.

6.5.4 Connexió a una xarxa Wi-Fi

El mòdul s'ha configurat en mode Station mitjançant la comanda *AT+CWMODE=1*. A continuació, s'han introduït les credencials de la xarxa amb la comanda *AT+CWJAP*, i el firmware ha confirmat correctament l'associació amb el punt d'accés mitjançant els missatges *WIFI CONNECTED* i *WIFI GOT IP*.

Posteriorment, s'ha utilitzat la comanda *AT+CIFSR* per consultar la configuració de xarxa del dispositiu. El terminal ha mostrat l'adreça IP assignada al mòdul, fet que confirma que la connexió a la xarxa s'ha completat correctament.

Aquest resultat mostra que l'ESP32-C6 pot gestionar una connexió Wi-Fi mitjançant les instruccions rebudes pel port sèrie, sense necessitat d'executar una aplicació desenvolupada específicament per a aquesta funció.

```
AT+CWMODE=1
OK
AT+CWJAP="ORBI28", "*****"
WIFI CONNECTED
WIFI GOT IP
AT+CIFSR
+CIFSR:STAIP,"192.168.10.36"
OK
```

Figura 17: Connexió de l'ESP32-C6 a una xarxa Wi-Fi i consulta de l'adreça IP mitjançant comandes AT.

Font: Elaboració pròpia a partir del terminal sèrie. La contrasenya de la xarxa s'ha ocultat per evitar mostrar informació privada.

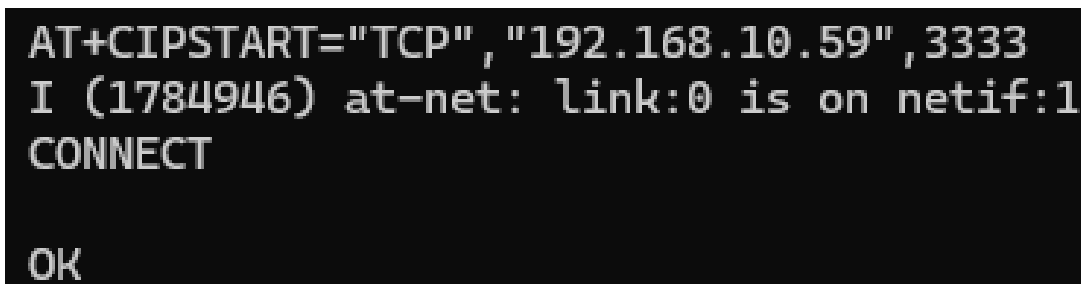
Tal com es mostra a la Figura 17, el mòdul s'ha configurat en mode Station, s'ha connectat correctament al punt d'accés i ha obtingut l'adreça IP 192.168.10.36. La resposta final *OK* confirma que les comandes s'han executat sense errors.

6.5.5 Obertura de la connexió TCP

Amb el dispositiu connectat a la xarxa Wi-Fi, s'ha iniciat una connexió TCP amb un servidor executat en un ordinador de la mateixa xarxa local. Per fer-ho, s'ha utilitzat la comanda ***AT+CIPSTART***, indicant l'adreça IP del servidor, 192.168.10.59, i el port 3333.

El firmware ha retornat els missatges *CONNECT* i *OK*, fet que confirma que la connexió s'ha establert correctament. Paral·lelament, el terminal del servidor ha registrat la connexió del client amb l'adreça IP 192.168.10.36.

Aquesta prova ha permès comprovar que l'ESP32-C6 pot accedir a un servidor TCP de la xarxa local i mantenir la connexió oberta abans de realitzar l'enviament de dades.

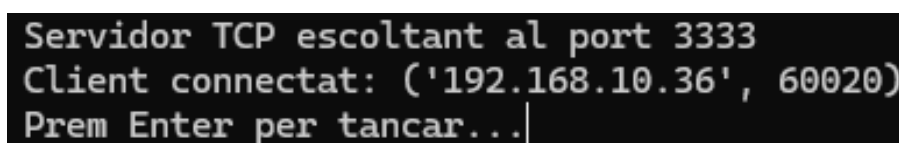


```
AT+CIPSTART="TCP","192.168.10.59",3333
I (1784946) at-net: link:0 is on netif:1
CONNECT

OK
```

Figura 18: Obertura correcta d'una connexió TCP des de l'ESP32-C6 mitjançant la comanda AT+CIPSTART.

Font: Elaboració pròpia a partir del terminal sèrie.



```
Servidor TCP escoltant al port 3333
Client connectat: ('192.168.10.36', 60020)
Prem Enter per tancar...|
```

Figura 19: Connexió de l'ESP32-C6 registrada pel servidor TCP executat a l'ordinador.

Font: Elaboració pròpia a partir del terminal de l'ordinador.

Tal com es mostra a les Figures 18 i 19, la connexió TCP s'ha establert correctament entre l'ESP32-C6 i el servidor executat a l'ordinador. La confirmació apareix tant al terminal del mòdul com al del servidor.

6.5.6 Enviament de dades al servidor

Un cop establerta la connexió TCP, s'ha utilitzat la comanda **AT+CIPSEND=18** per indicar al firmware la longitud del missatge que es volia transmetre. Després de rebre l'indicador d'enviament, s'ha introduït el text *Prova TCP correcta*.

El terminal sèrie no ha mostrat els caràcters introduïts durant l'enviament, ja que l'eco local estava desactivat. Tot i això, el servidor TCP executat a l'ordinador ha rebut correctament el missatge i l'ha mostrat a la consola.

D'aquesta manera, s'ha validat la cadena completa de comunicació entre el terminal, l'ESP32-C6 i el servidor. El resultat confirma que el mòdul pot actuar com a intermediari i transmetre mitjançant Wi-Fi les dades rebudes des d'un sistema extern.

```
AT+CIPSTART="TCP","192.168.10.59",3333
I (2213651) at-net: link:0 is on netif:1
CONNECT

OK
AT+CIPSEND=18
```

Figura 20: Comandes utilitzades per establir la connexió TCP i iniciar l'enviament de dades des de l'ESP32-C6.

Font: Elaboració pròpia a partir del terminal sèrie.

```
Servidor TCP escoltant al port 3333
Client connectat: ('192.168.10.36', 60021)
Missatge rebut: Prova TCP correcta
Prem Enter per tancar...
```

Figura 21: Recepció correcta del missatge enviat per l'ESP32-C6 al servidor TCP.

Font: Elaboració pròpia a partir del terminal del servidor TCP.

Tal com es mostra a les Figures 20 i 21, l'ESP32-C6 ha iniciat l'enviament de 18 bytes mitjançant la comanda *AT+CIPSEND=18*, i el servidor ha rebut correctament el missatge *Prova TCP correcta*. Aquest resultat confirma el funcionament complet de la comunicació TCP.

6.6 Validació amb les plataformes de desenvolupament

A més de les proves realitzades directament amb l'ESP32-C6, s'han dut a terme comprovacions bàsiques amb la Giant Gecko Starter Kit i s'ha estudiat la compatibilitat de comunicació amb la Teensy 3.6.

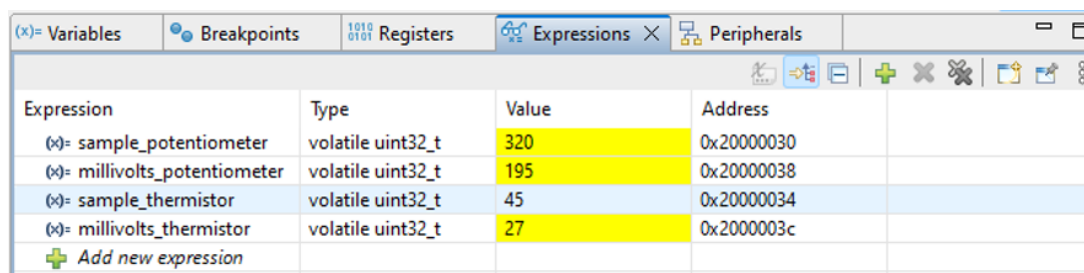
6.6.1 Proves amb la Giant Gecko

La Giant Gecko Starter Kit s'ha utilitzat per comprovar el funcionament de Simplicity Studio i familiaritzar-se amb la configuració dels perifèrics integrats a la plataforma.

En una primera prova s'han configurat els GPIO dels LED i dels polsadors. El LED0 s'ha utilitzat com a sortida digital i s'ha fet parpellejar periòdicament. Al mateix temps, el polsador

PB0 s'ha configurat com a entrada digital amb una resistència interna de *pull-up*. Quan el programa detecta una pulsació del botó, modifica l'estat del LED1. Aquesta prova ha permès comprovar el funcionament correcte de les entrades i sortides digitals de la placa.

Posteriorment, s'ha configurat el convertidor analògic-digital per llegir dos canals diferents. El canal ADC0_CH7 s'ha associat al potenciòmetre, mentre que el canal ADC0_CH6 s'ha utilitzat per obtenir la lectura del termistor. Els valors adquirits s'han emmagatzemat en les variables `sample_potentiometer` i `sample_thermistor`, i també s'han convertit a mil·livolts.



Expression	Type	Value	Address
(x)= sample_potentiometer	volatile uint32_t	320	0x20000030
(x)= millivolts_potentiometer	volatile uint32_t	195	0x20000038
(x)= sample_thermistor	volatile uint32_t	45	0x20000034
(x)= millivolts_thermistor	volatile uint32_t	27	0x2000003c
+ Add new expression			

Figura 22: Lectura dels valors del potenciòmetre i del termistor mitjançant l'ADC de la Giant Gecko Starter Kit.

Font: Elaboració pròpia a partir de l'entorn de depuració de Simplicity Studio.

Tal com es mostra a la Figura 22, l'entorn de depuració permet visualitzar tant els valors digitals obtinguts pel convertidor com la tensió corresponent en mil·livolts. Aquest resultat confirma que els dos canals analògics s'han configurat i llegit correctament.

Les aplicacions s'han compilat i programat correctament, i el comportament observat ha coincidit amb el definit al codi. D'aquesta manera, s'ha validat el funcionament dels GPIO, dels pulsadors, dels LED i del convertidor analògic-digital, així com la correcta configuració de l'entorn Simplicity Studio.

6.6.2 Compatibilitat amb la Teensy 3.6

La compatibilitat amb la Teensy 3.6 s'ha analitzat a partir de les interfícies disponibles i de les connexions definides dins del disseny.

La comunicació prevista es basa en UART i requereix connectar de manera creuada les línies TX i RX, a més de compartir una referència comuna de terra. Aquesta configuració permet que la Teensy envii ordres o dades a l'ESP32-C6 i rebi les respostes generades pel mòdul.

En el moment de redactar la memòria no s'ha executat una aplicació completa amb les dues plataformes connectades físicament. Per tant, la compatibilitat queda definida a nivell de disseny, mentre que la validació funcional conjunta es durà a terme sobre la placa docent

definitiva abans de la defensa del Treball de Fi de Grau.

6.7 Resum dels resultats

Les proves realitzades han permès validar el funcionament bàsic de l'ESP32-C6, incloent-hi la compilació i programació del firmware i el control dels GPIO. També s'ha comprovat la connectivitat Wi-Fi en els modes Station i SoftAP, tant mitjançant la connexió a una xarxa existent com mitjançant la creació d'un punt d'accés propi.

Pel que fa a Bluetooth Low Energy, el dispositiu s'ha detectat i connectat correctament des de l'aplicació nRF Connect. A més, s'ha validat l'intercanvi de dades mitjançant una característica GATT, enviant un missatge des del telèfon mòbil i visualitzant-lo al monitor sèrie de l'ESP32-C6.

També s'ha comprovat el funcionament del firmware ESP-AT. El mòdul ha respost correctament a les comandes AT, s'ha connectat a una xarxa Wi-Fi, ha obtingut una adreça IP, ha establert una connexió TCP i ha transmès correctament un missatge a un servidor executat en un ordinador de la mateixa xarxa.

Les proves amb la Giant Gecko han confirmat el funcionament dels GPIO, dels polsadors, dels LED i del convertidor analògic-digital, així com la correcta configuració de l'entorn Simplicity Studio. En canvi, la comunicació conjunta de l'ESP32-C6 amb la Teensy 3.6 i la Giant Gecko queda pendent de validació experimental sobre la placa docent definitiva.

La Taula 5 resumeix quines proves i exemples s'han executat sobre cadascuna de les plaques validades durant el projecte: el mòdul ESP32-C6-WROOM-1 i la Giant Gecko Starter Kit. El símbol ✓ indica que la prova s'ha realitzat i validat sobre la placa corresponent, i el símbol ✗ que la prova no s'ha realitzat sobre aquella placa. La comunicació conjunta amb les plataformes principals resta pendent fins a disposar de la PCB definitiva.

Taula 5: Proves i exemples realitzats sobre cada placa.

Prova / exemple	ESP32-C6	Giant Gecko
Compilació i programació	✓	✓
Control de GPIO i LED	✓	✓
Lectura de pulsadors	✗	✓
Lectura ADC (potenciòmetre i termistor)	✗	✓
Wi-Fi en mode Station	✓	✗
Wi-Fi en mode SoftAP	✓	✗
Comunicació BLE i intercanvi GATT	✓	✗
Funcionament com a mòdem (ESP-AT)	✓	✗
Enviament de dades per TCP	✓	✗
Comunicació UART conjunta amb l'ESP32-C6	Pendent fins a la PCB definitiva	

Font: Elaboració pròpia a partir de les proves realitzades.

En el moment de tancar aquesta memòria, la placa final encara no havia estat rebuda de l'empresa encarregada de la seva fabricació. Per aquest motiu, no ha estat possible executar les proves que requereixen la PCB completament fabricada i muntada, especialment la validació de la comunicació UART entre l'ESP32-C6 i les dues plataformes de desenvolupament.

Els resultats principals de les proves es recullen a la Taula 4, mentre que la Taula 5 mostra la distribució de les proves i exemples per placa. En conjunt, s'han validat experimentalment les funcionalitats individuals de l'ESP32-C6 i de la Giant Gecko mitjançant les plaques de desenvolupament disponibles. Les proves pendents corresponen exclusivament a la integració física del sistema complet sobre la PCB definitiva.

Abans de la defensa del Treball de Fi de Grau es duran a terme les comprovacions pendents sobre la placa docent definitiva, i els resultats obtinguts es presentaran i comentaran davant del tribunal.

Aquest apartat recull únicament una síntesi dels resultats experimentals obtinguts. La valoració global d'aquests resultats, el grau d'assoliment dels objectius i les possibles línies de continuació es desenvolupen al capítol següent.

Capítol 7

Conclusions

Aquest capítol presenta la valoració global del treball realitzat, el grau d'assoliment dels objectius plantejats i les possibles línies de continuació del projecte.

7.1 Conclusions generals

El desenvolupament d'aquest Treball de Fi de Grau ha permès definir la integració i validar les funcionalitats necessàries per incorporar comunicació sense fils a la nova placa docent de l'Escola d'Enginyeria de la Universitat Autònoma de Barcelona.

L'estudi de les diferents alternatives disponibles ha conduït a la selecció de l'ESP32-C6-WROOM-1-N8 com a mòdul més adequat per als requisits del projecte. La seva combinació de connectivitat, interfícies de comunicació, facilitat d'integració, disponibilitat de documentació i cost permet utilitzar-lo tant com a dispositiu autònom com, principalment, com a mòdul auxiliar controlat per una altra plataforma.

Les proves realitzades han confirmat que el mòdul pot executar les funcionalitats de comunicació previstes i que els entorns de desenvolupament seleccionats són adequats per treballar-hi. També s'ha demostrat que el firmware ESP-AT es pot adaptar a les connexions disponibles i utilitzar-se per controlar el dispositiu mitjançant una interfície sèrie.

La principal limitació del treball ha estat la impossibilitat de validar experimentalment el sistema complet sobre la PCB definitiva, ja que la placa no s'ha rebut de l'empresa encarregada de la seva fabricació abans del tancament de la memòria. En conseqüència, la integració de l'ESP32-C6 amb la Teensy 3.6 i la Giant Gecko ha quedat definida i revisada a nivell de disseny, però la comunicació conjunta mitjançant UART resta pendent de comprovació física.

Tot i aquesta limitació, les funcionalitats individuals del mòdul s'han validat mitjançant

plaques de desenvolupament. S'ha comprovat la connectivitat Wi-Fi en els modes Station i SoftAP, la comunicació Bluetooth Low Energy mitjançant GATT, el funcionament del firmware ESP-AT, l'establiment d'una connexió TCP i l'enviament correcte de dades. També s'han validat diferents perifèrics de la Giant Gecko mitjançant Simplicity Studio.

La placa estarà disponible abans de la defensa del Treball de Fi de Grau, de manera que es duran a terme les comprovacions pendents i els resultats obtinguts es presentaran i comentaran davant del tribunal.

Un dels aspectes més rellevants del treball és que no s'ha limitat a comprovar el funcionament del mòdul, sinó que també s'ha preparat una base reutilitzable per al professorat i l'alumnat. Els projectes desenvolupats, la carpeta compartida de materials i el manual docent de l'Annex A facilitaran la reproducció de les proves i l'adaptació dels codis a noves pràctiques.

En conjunt, el treball proporciona una base tècnica i documental sòlida per continuar el redisseny de la placa docent i ampliar-ne les possibilitats d'ús en activitats relacionades amb microcontroladors, comunicacions sense fils i sistemes encastats.

7.2 Grau d'assoliment dels objectius

L'objectiu general del projecte era integrar i validar un mòdul de comunicació sense fils dins d'una placa docent compatible amb la Teensy 3.6 i la Giant Gecko Starter Kit. Aquest objectiu s'ha assolit pel que fa a la selecció del mòdul, la definició de la seva integració, el desenvolupament del firmware i la validació individual de les funcionalitats de comunicació. La validació experimental del sistema complet sobre la PCB definitiva resta pendent en el moment de tancar aquesta memòria.

S'ha completat l'estudi de les alternatives disponibles, la selecció del mòdul i la definició de les connexions necessàries per integrar-lo dins de la nova arquitectura. També s'han preparat els entorns de desenvolupament i s'han validat les funcionalitats principals previstes per al dispositiu.

La utilització de l'ESP32-C6 com a mòdul autònom s'ha comprovat mitjançant els projectes desenvolupats amb ESP-IDF. Paral·lelament, l'adaptació d'ESP-AT ha permès validar el funcionament del dispositiu com a mòdul de comunicació controlat externament, que és el mode d'ús previst principalment per a la placa docent.

També s'han realitzat proves bàsiques amb la Giant Gecko Starter Kit i s'ha definit la comunicació UART de l'ESP32-C6 amb les plataformes principals. Aquesta comunicació no s'ha pogut validar físicament sobre la PCB definitiva abans del tancament de la memòria, ja que la placa encara no havia estat rebuda de l'empresa encarregada de la seva fabricació.

Per tant, es considera que s'han assolit els objectius relacionats amb l'estudi i la selecció del mòdul, la definició de la seva integració, el desenvolupament del firmware, la connectivitat Wi-Fi, la comunicació Bluetooth Low Energy i el funcionament mitjançant ESP-AT. Els aspectes pendents corresponen únicament a la integració física completa del sistema i a la seva validació conjunta sobre la nova placa docent.

7.3 Línies futures

La continuació més immediata del projecte consisteix a rebre i verificar la nova placa docent, i executar-hi les proves realitzades prèviament sobre les plaques de desenvolupament.

Aquesta validació haurà d'incloure la comprovació de les alimentacions, els senyals de control, les interfícies USB i UART, la programació del mòdul i el comportament de les comunicacions sense fils.

Una de les tasques prioritàries serà validar la comunicació directa entre l'ESP32-C6 i la Teensy 3.6 o la Giant Gecko. Aquesta integració permetrà que el microcontrolador principal adquireixi dades dels sensors i utilitzi l'ESP32-C6 per transmetre-les a altres dispositius.

A partir d'aquesta arquitectura es podran desenvolupar pràctiques docents més completes, com ara la transmissió de dades a un servidor, el control remot d'actuadors o la comunicació amb un telèfon mòbil.

També es podria desenvolupar una aplicació mòbil pròpia que substituís nRF Connect i oferís una interfície adaptada a les activitats docents. D'aquesta manera, els estudiants podrien visualitzar dades i enviar ordres sense necessitat d'utilitzar una eina genèrica de desenvolupament.

Altres línies de continuació possibles són:

- Desenvolupar i validar un protocol de comunicació UART entre el microcontrolador principal i l'ESP32-C6 sobre la placa docent definitiva.
- Implementar un servidor web accessible des d'un navegador.
- Incorporar actualitzacions remotes del firmware.
- Analitzar el consum energètic i l'abast de les comunicacions.
- Automatitzar part de les proves de validació.
- Desenvolupar noves pràctiques basades en Thread o Zigbee.
- Ampliar la carpeta compartida de materials amb nous exemples, projectes i documentació docent.

- Actualitzar el manual docent a mesura que s'incorporin noves funcionalitats.

Aquestes ampliacions permetrien convertir la placa en una plataforma docent més completa, flexible i adaptable a diferents assignatures i nivells de dificultat.

Annex A

Manual docent de configuració i proves

A.1 Introducció

Aquest annex presenta el manual docent associat als projectes de programació i validació desenvolupats durant el Treball de Fi de Grau. El seu objectiu és facilitar la reproducció de les proves realitzades i permetre que futurs usuaris puguin preparar els entorns de desenvolupament, configurar els projectes, programar les plaques i comprovar-ne el funcionament.

La descripció tècnica de les funcionalitats implementades, del funcionament general dels programes i dels resultats obtinguts es presenta als capítols corresponents de la memòria. Per evitar repetir aquesta informació, el present annex se centra exclusivament en els procediments pràctics necessaris per utilitzar els materials desenvolupats.

Els codis font complets no s'inclouen directament dins del document. Aquests es lliuren en una carpeta externa organitzada segons la plataforma i el tipus de prova. Aquesta carpeta conté els projectes de l'ESP32-C6, els projectes de la Giant Gecko Starter Kit i els scripts auxiliars.

La carpeta es pot consultar i descarregar mitjançant l'enllaç següent:

Accés a la carpeta de materials del projecte

Al llarg del manual s'indica la ruta concreta de cada projecte dins d'aquesta carpeta, els programes necessaris per obrir-lo, els paràmetres que cal adaptar, les comandes que s'han d'executar i el resultat que s'ha d'obtenir per considerar correcta cada prova.

A.2 Organització dels materials lliurats

La carpeta externa indicada a l'apartat anterior conté els projectes i els recursos auxiliars desenvolupats durant el treball. Els materials s'organitzen en els directoris següents:

- **README.md**: Document inicial de la carpeta, amb una descripció general dels continguts i les indicacions bàsiques per localitzar cada projecte.
- **ESP32-C6**: Projectes complets desenvolupats amb ESP-IDF per a les proves de funcionament bàsic, GPIO, Wi-Fi, Bluetooth Low Energy i ESP-AT.
- **Giant_Gecko**: Projectes desenvolupats amb Simplicity Studio per a les proves de GPIO, polsadors, LED i altres perifèrics utilitzats.
- **Scripts**: Programes auxiliars necessaris per a algunes proves, com el servidor TCP executat a l'ordinador.

Dins dels directoris **ESP32-C6** i **Giant_Gecko**, cada prova es troba guardada com un projecte independent. No s'inclouen les carpetes de compilació generades automàticament, com ara **build**, ja que es poden tornar a crear en compilar els projectes i augmenten innecessàriament la mida dels materials lliurats.

A.3 Preparació dels entorns de desenvolupament

Abans d'obrir i executar els projectes inclosos en la carpeta de materials, cal instal·lar i configurar els diferents entorns i eines utilitzats durant el desenvolupament del projecte. Les eines necessàries depenen de la plataforma i del tipus de prova que es vulgui reproduir.

Per als projectes de l'ESP32-C6 s'utilitza ESP-IDF com a entorn principal de desenvolupament. Per a les proves de Bluetooth Low Energy també és necessària l'aplicació nRF Connect en un telèfon mòbil.

Els projectes de la Giant Gecko Starter Kit s'obren, compilen, programen i depuren mitjançant Simplicity Studio. Finalment, per a les proves del firmware ESP-AT i de comunicació TCP es necessita un terminal sèrie i un entorn de Python per executar el servidor auxiliar.

Les eines utilitzades en aquest projecte són les següents:

- **ESP-IDF**: Compilació, programació i monitorització dels projectes de l'ESP32-C6.
- **nRF Connect**: Detecció del dispositiu BLE, connexió als serveis GATT i intercanvi de dades.

- **Simplicity Studio:** Desenvolupament, compilació, programació i depuració dels projectes de la Giant Gecko.
- **Terminal sèrie:** Enviament de comandes AT i visualització de les respostes de l'ESP32-C6.
- **Python:** Execució del servidor TCP utilitzat durant la prova d'enviament de dades.

En els apartats següents es descriu la preparació de cadascun d'aquests entorns. Les versions concretes utilitzades durant el desenvolupament també s'inclouran a la carpeta de materials per facilitar la reproducció de les proves.

A.3.1 Instal·lació i configuració d'ESP-IDF

ESP-IDF és l'entorn oficial utilitzat per desenvolupar, compilar, programar i monitoritzar els projectes de l'ESP32-C6. Durant aquest treball s'ha utilitzat la versió 6.0.1 en un ordinador amb Windows 11.

Per preparar l'entorn cal descarregar l'ESP-IDF Installation Manager des de la pàgina oficial d'Espressif [28, 29]:

`https://dl.espressif.com/dl/eim/`

La documentació oficial d'instal·lació per a Windows es pot consultar a:

`https://docs.espressif.com/projects/esp-idf/en/stable/esp32/get-started/
windows-setup.html`

L'instal·lador s'encarrega de descarregar i configurar els components necessaris per treballar amb ESP-IDF, entre els quals s'inclouen l'entorn de Python, Git, CMake, Ninja, les eines de compilació i les utilitats de programació del dispositiu. Per tant, no és necessari instal·lar aquests components manualment si el procés d'instal·lació finalitza correctament.

Durant la instal·lació cal seleccionar la versió 6.0.1 d'ESP-IDF, ja que és la versió utilitzada durant el desenvolupament i la validació dels projectes d'aquest treball.

En la configuració utilitzada durant el projecte, ESP-IDF es troba instal·lat a:

`C:\esp32\v6.0.1\esp-idf`

Les eines complementàries gestionades per Espressif poden instal·lar-se en una ruta diferent, definida automàticament durant el procés. Aquesta ubicació no s'ha de modificar manualment, sempre que l'instal·lador finalitzi sense errors.

Un cop completada la instal·lació, cal obrir la terminal creada per l'instal·lador. Per comprovar que la instal·lació és correcta, s'ha d'executar:

```
idf.py --version
```

La terminal ha de mostrar una resposta similar a:

```
ESP-IDF v6.0.1
```

Amb aquesta comprovació es considera que l'entorn ha quedat instal·lat i preparat per utilitzar els projectes de l'ESP32-C6.

A.3.2 Instal·lació de nRF Connect

nRF Connect for Mobile és l'aplicació utilitzada per detectar l'ESP32-C6, establir una connexió Bluetooth Low Energy i accedir als serveis i característiques GATT implementats en el projecte [21].

L'aplicació es pot instal·lar en un telèfon mòbil des de la botiga d'aplicacions del dispositiu, cercant el nom *nRF Connect for Mobile*. Un cop instal·lada, cal activar el Bluetooth del telèfon i concedir els permisos necessaris per detectar dispositius propers. Segons el sistema operatiu i la seva versió, també pot ser necessari habilitar permisos relacionats amb la ubicació.

L'aplicació permet escanejar els dispositius BLE disponibles, identificar l'ESP32-C6, establir-hi una connexió i consultar els serveis i característiques GATT. També permet executar operacions de lectura i escriptura per comprovar l'intercanvi de dades amb el mòdul.

No s'ha utilitzat la versió de nRF Connect per a ordinador perquè, per realitzar funcions de comunicació Bluetooth Low Energy, pot requerir hardware compatible de Nordic Semiconductor, com un kit de desenvolupament o un dongle USB. En canvi, la versió mòbil permet realitzar directament totes les comprovacions necessàries mitjançant el Bluetooth integrat del telèfon, sense afegir dispositius externs.

No cal realitzar cap configuració addicional en aquesta fase. El procediment d'escaneig, connexió i intercanvi de dades es descriurà posteriorment a l'apartat corresponent al projecte Bluetooth Low Energy.

A.3.3 Instal·lació i configuració de Simplicity Studio

Simplicity Studio és l'entorn de desenvolupament utilitzat per obrir, compilar, programar i depurar els projectes de la Giant Gecko Starter Kit. Durant aquest treball s'ha utilitzat Simplicity Studio 5, per la qual cosa es recomana utilitzar aquesta mateixa versió per evitar possibles problemes de compatibilitat amb els projectes lliurats.

L'instal·lador de Simplicity Studio 5 es pot descarregar des de la pàgina oficial de Silicon Labs [30, 31]:

```
https://www.silabs.com/software-and-tools/simplicity-studio/  
simplicity-studio-version-5
```

La guia oficial d'instal·lació es pot consultar a:

```
https://docs.silabs.com/simplicity-studio-5-users-guide/latest/  
ss-5-users-guide-getting-started/install-ss-5-and-software
```

Abans d'iniciar la instal·lació es recomana connectar la Giant Gecko Starter Kit a l'ordinador mitjançant el connector USB de depuració. D'aquesta manera, Simplicity Studio pot detectar la placa i seleccionar amb més facilitat les eines i els paquets compatibles.

Durant el procés d'instal·lació cal acceptar els termes de la llicència, seleccionar la carpeta de destinació i completar la instal·lació de l'entorn. Un cop obert Simplicity Studio, cal accedir a l'Installation Manager i instal·lar el suport corresponent als microcontroladors EFM32 i a la Giant Gecko Starter Kit.

També cal instal·lar el compilador i els paquets de desenvolupament proposats per l'entorn. La selecció concreta pot variar segons la revisió de la placa, però Simplicity Studio mostra automàticament els recursos compatibles quan detecta el kit connectat.

Un cop finalitzada la instal·lació, la placa ha d'aparèixer a la finestra principal de Simplicity Studio. Des d'aquesta pantalla es pot comprovar que el dispositiu ha estat reconegut i que les opcions de creació, importació, programació i depuració de projectes estan disponibles.

No és necessari crear cap projecte nou en aquesta fase. El procediment per importar i executar els projectes inclosos a la carpeta de materials es descriurà posteriorment a l'apartat dedicat a la Giant Gecko.

A.3.4 Preparació de Python i del terminal sèrie

La instal·lació d'ESP-IDF incorpora l'entorn de Python necessari per executar les eines associades als projectes de l'ESP32-C6. Per tant, no cal instal·lar Python de manera independent.

Python s'utilitza en aquest projecte per executar el servidor TCP auxiliar i per obrir un terminal sèrie mitjançant l'eina `miniterm`. Aquesta eina permet enviar comandes AT a l'ESP32-C6 i visualitzar les respostes generades pel firmware ESP-AT.

Per comprovar que Python està disponible dins de la terminal d'ESP-IDF, cal executar:

```
python --version
```

La terminal ha de mostrar la versió de Python associada a l'entorn d'ESP-IDF.

El terminal sèrie es pot obrir mitjançant la comanda següent:

```
python -m serial.tools.miniterm COM9 115200
```

El valor `COM9` s'ha de substituir pel port assignat a la placa en cada ordinador. El valor `115200` correspon a la velocitat de comunicació utilitzada durant les proves.

Abans d'obrir `miniterm`, cal comprovar que el port sèrie no estigui sent utilitzat per una altra aplicació, com el monitor d'ESP-IDF o Simplicity Studio, ja que un mateix port COM no pot ser obert simultàniament per dos programes.

La configuració concreta del terminal, l'enviament de comandes AT i la manera de tancar `miniterm` es descriuran posteriorment a l'apartat dedicat al firmware ESP-AT.

El servidor TCP utilitzat durant les proves també s'executarà mitjançant l'entorn de Python inclòs amb ESP-IDF. El fitxer corresponent estarà disponible al directori `Scripts` de la carpeta de materials.

A.4 Connexió i identificació de les plaques

Abans de compilar i programar els projectes, cal connectar correctament les plaques a l'ordinador i identificar el port de comunicació assignat pel sistema operatiu.

A.4.1 Connexió de la placa ESP32-C6

La placa de desenvolupament utilitzada per realitzar les proves de l'ESP32-C6 disposa de dos connectors USB tipus C, identificats com a `COM` i `USB`.

El connector **COM** està associat al convertidor USB-UART de la placa i permet programar el dispositiu, obrir el monitor sèrie i enviar comandes AT. Aquest és el connector que s'ha utilitzat principalment durant les proves descrites en aquest manual.

El connector **USB** està connectat directament a la interfície USB Serial/JTAG de l'ESP32-C6. Aquesta connexió també permet realitzar funcions de programació, comunicació i depuració, però el seu comportament depèn del firmware carregat i de la configuració del projecte.

Per reproduir els procediments del manual es recomana connectar inicialment el cable USB al connector **COM**.

El cable utilitzat ha de permetre la transmissió de dades. Alguns cables USB només serveixen per alimentar dispositius i no permeten que la placa sigui detectada per l'ordinador.

A.4.2 Identificació del port COM a Windows

Quan es connecta la placa a l'ordinador, Windows assigna un port de comunicació [32]. Aquest port pot variar entre ordinadors i també pot canviar si la placa es connecta a un connector USB diferent.

Per identificar-lo, cal seguir els passos següents:

1. Connectar la placa ESP32-C6 a l'ordinador mitjançant el connector **COM**.
2. Obrir l'Administrador de dispositius de Windows.
3. Desplegar l'apartat *Ports (COM i LPT)*.
4. Localitzar el dispositiu que apareix en connectar la placa.
5. Anotar el número de port assignat, per exemple **COM9**.

Al llarg del manual s'utilitza **COM9** com a exemple. Aquest valor s'ha de substituir pel port que aparegui en cada ordinador.

A.4.3 Connexió de la Giant Gecko Starter Kit

La Giant Gecko Starter Kit incorpora un depurador integrat que permet programar i depurar el microcontrolador mitjançant una connexió USB.

Per utilitzar-la, cal connectar el cable USB al connector associat al depurador de la placa. Un cop connectada, Simplicity Studio ha de detectar automàticament el kit i mostrar-lo a la pantalla principal.

En les proves realitzades amb la Giant Gecko no és necessari identificar manualment un port COM per compilar i programar els projectes, ja que aquestes funcions es gestionen directament des de Simplicity Studio.

A.4.4 Comprovacions prèvies

Abans de programar qualsevol projecte, es recomana comprovar que:

- La placa està connectada mitjançant un cable USB de dades.
- L'ordinador ha detectat correctament el dispositiu.
- El port COM de l'ESP32-C6 ha estat identificat.
- Cap altra aplicació està utilitzant el mateix port COM.
- Simplicity Studio reconeix la Giant Gecko Starter Kit.

Un cop realitzades aquestes comprovacions, es poden començar a obrir, compilar i executar els projectes inclosos a la carpeta de materials.

A.5 Execució dels projectes de l'ESP32-C6

Els projectes de l'ESP32-C6 es troben dins del directori `ESP32-C6` de la carpeta de materials. Cada prova es lliurarà com un projecte independent d'ESP-IDF.

Les comandes s'han d'executar des d'una terminal d'ESP-IDF. El port `COM9` utilitzat als exemples s'ha de substituir pel port assignat a la placa en cada ordinador.

A.5.1 Projecte Hello World

Aquest projecte es basa en l'exemple oficial *hello_world* d'ESP-IDF [24, 20].

Ruta del projecte:

`ESP32-C6\01_Hello_World`

Aquest projecte permet comprovar que l'entorn d'ESP-IDF està instal·lat correctament, que la placa es pot programar i que el monitor sèrie funciona.

Cal accedir al directori del projecte:

```
cd C:\ruta\Codis_TFG\ESP32-C6\01_Hello_World
```

A continuació, cal executar:

```
idf.py set-target esp32c6
idf.py build
idf.py -p COM9 flash monitor
```

La prova es considera correcta quan el monitor sèrie mostra:

```
Hello world!
```

Per tancar el monitor sèrie cal prémer:

```
Ctrl + ]
```

No cal modificar cap fitxer del projecte.

A.5.2 Projecte GPIO Blink

Aquest projecte es basa en l'exemple oficial *blink* d'ESP-IDF [33].

Ruta del projecte:

```
ESP32-C6\02_GPIO_Blink
```

Aquest projecte permet comprovar el control d'una sortida digital mitjançant el parpelleig d'un LED.

Cal accedir al directori del projecte i executar:

```
idf.py set-target esp32c6
idf.py build
idf.py -p COM9 flash monitor
```

El número del GPIO i el temps de parpelleig es poden modificar al fitxer principal del projecte. Abans de realitzar qualsevol canvi, cal comprovar quin GPIO està connectat al LED de la placa utilitzada.

La prova es considera correcta quan el LED canvia periòdicament d'estat i el monitor sèrie mostra els missatges associats a l'activació i desactivació de la sortida.

A.5.3 Projecte Wi-Fi Station

Aquest projecte es basa en l'exemple oficial de connexió Wi-Fi en mode Station d'ESP-IDF [34].

Ruta del projecte:

```
ESP32-C6\03_WiFi_Station
```

Aquest projecte connecta l'ESP32-C6 a una xarxa Wi-Fi existent.

Abans de compilar-lo, cal configurar el nom i la contrasenya de la xarxa. Per obrir el menú de configuració s'ha d'executar:

```
idf.py menuconfig
```

Dins del menú cal accedir a la configuració específica de l'exemple i introduir:

- El nom de la xarxa Wi-Fi.
- La contrasenya de la xarxa Wi-Fi.

Després de guardar els canvis, cal executar:

```
idf.py set-target esp32c6  
idf.py build  
idf.py -p COM9 flash monitor
```

La prova es considera correcta quan el monitor indica que l'ESP32-C6 s'ha connectat a la xarxa i mostra l'adreça IP assignada.

A.5.4 Projecte Wi-Fi SoftAP

Aquest projecte es basa en l'exemple oficial de creació d'un punt d'accés (SoftAP) d'ESP-IDF [35].

Ruta del projecte:

```
ESP32-C6\04_WiFi_SoftAP
```

Aquest projecte configura l'ESP32-C6 com a punt d'accés Wi-Fi.

Els paràmetres principals del punt d'accés es poden modificar mitjançant:

`idf.py menuconfig`

Els valors que es poden adaptar són:

- El nom de la xarxa creada.
- La contrasenya.
- El canal Wi-Fi.
- El nombre màxim de dispositius connectats.

Per compilar i programar el projecte cal executar:

```
idf.py set-target esp32c6
idf.py build
idf.py -p COM9 flash monitor
```

Un cop iniciat el programa, la xarxa creada per l'ESP32-C6 ha d'aparèixer entre les xarxes Wi-Fi disponibles del telèfon mòbil.

La prova es considera correcta quan el telèfon es connecta al punt d'accés i el monitor sèrie indica la connexió del dispositiu client i l'adreça IP assignada.

A.5.5 Projecte Bluetooth Low Energy amb GATT

Ruta del projecte:

ESP32-C6\05_BLE_GATT

Aquest projecte configura l'ESP32-C6 com a perifèric Bluetooth Low Energy i implementa un servei GATT per intercanviar dades, prenent com a base l'exemple oficial *bleprph* de la pila NimBLE d'ESP-IDF [36, 37].

Cal accedir al directori del projecte i executar:

```
idf.py set-target esp32c6
idf.py build
idf.py -p COM9 flash monitor
```

Un cop iniciat el programa, cal obrir *nRF Connect for Mobile* i seguir els passos següents:

1. Iniciar l'escaneig de dispositius Bluetooth Low Energy.
2. Localitzar el dispositiu corresponent a l'ESP32-C6.
3. Establir la connexió.
4. Accedir al servei GATT disponible.
5. Seleccionar la característica que permet operacions d'escriptura.
6. Enviar un missatge de prova.

El missatge enviat es pot introduir en format de text o en format hexadecimal, segons les opcions disponibles a l'aplicació.

La prova es considera correcta quan el monitor sèrie mostra el contingut enviat des del telèfon mòbil.

Els paràmetres que es poden modificar al projecte són el nom anunciat pel dispositiu, els identificadors del servei i de la característica i el tractament de les dades rebudes.

A.5.6 Projecte ESP-AT

La compilació del firmware ESP-AT segueix el procediment oficial d'Espressif [38, 23].

Ruta del projecte:

ESP32-C6\06_ESP_AT

Aquest directori conté el projecte ESP-AT utilitzat per configurar l'ESP32-C6 com a mòdul de comunicació controlat mitjançant comandes AT.

El firmware inclou la configuració utilitzada durant el projecte, especialment els paràmetres del port UART de comandes.

Abans de compilar-lo, cal accedir al directori del projecte ESP-AT i inicialitzar l'entorn d'ESP-IDF corresponent.

La compilació es realitza mitjançant les eines pròpies d'ESP-AT. Un cop generat el firmware, el binari de fàbrica es pot programar mitjançant una comanda amb l'estructura següent:

```
esptool.py --chip esp32c6 -p COM9 -b 460800  
write_flash 0x0 build\factory\factory_ESP32C6-4MB.bin
```

El nom i la ubicació exacta del fitxer poden variar segons la configuració de memòria seleccionada. Cal utilitzar el binari generat dins del directori `build\factory`.

Després de programar el firmware, es pot obrir el terminal sèrie mitjançant:

```
python -m serial.tools.miniterm COM9 115200
```

El port `COM9` s'ha de substituir pel port assignat a la placa.

Per comprovar la comunicació cal escriure:

```
AT
```

La resposta esperada és:

```
OK
```

Per consultar la versió del firmware cal executar:

```
AT+GMR
```

La prova es considera correcta quan el dispositiu respon a les dues comandes.

A.5.7 Connexió Wi-Fi mitjançant comandes AT

Amb el firmware ESP-AT carregat, cal configurar l'ESP32-C6 en mode Station:

```
AT+CWMODE=1
```

A continuació, cal connectar-lo a la xarxa Wi-Fi:

```
AT+CWJAP="NOM_XARXA","CONTRASENYA"
```

Els valors `NOM_XARXA` i `CONTRASENYA` s'han de substituir pels valors corresponents.

Per consultar l'adreça IP assignada cal executar:

```
AT+CIFSR
```

La connexió es considera correcta quan el dispositiu respon amb `OK` i mostra una adreça IP vàlida.

A.5.8 Connexió TCP i enviament de dades

Abans d'obrir la connexió TCP, cal executar el servidor auxiliar inclòs al directori:

```
Scripts\Servidor_TCP
```

El servidor s'executa des d'una terminal d'ESP-IDF mitjançant:

```
python servidor_tcp.py
```

L'ordinador i l'ESP32-C6 han d'estar connectats a la mateixa xarxa Wi-Fi.

Per obrir la connexió TCP des de l'ESP32-C6 cal executar:

```
AT+CIPSTART="TCP","ADRECA_IP",PORT
```

Els valors `ADRECA_IP` i `PORT` s'han de substituir per l'adreça IP de l'ordinador i el port configurat al servidor.

Per indicar la quantitat de dades que es volen enviar cal utilitzar:

```
AT+CIPSEND=LONGITUD
```

Quan el dispositiu mostra el símbol:

```
>
```

cal escriure el missatge amb la longitud indicada.

La prova es considera correcta quan el terminal del servidor mostra la connexió de l'ESP32-C6 i el missatge rebut.

Per tancar la connexió es pot executar:

```
AT+CIPCLOSE
```

A.6 Execució dels projectes de la Giant Gecko

Els projectes de la Giant Gecko es troben dins del directori `Giant_Gecko`. Aquests projectes s'han d'obrir, compilar i programar mitjançant Simplicity Studio.

A.6.1 Projecte GPIO, LED i polsadors

Ruta del projecte:

Giant_Gecko\01_GPIO

Per executar el projecte cal:

1. Connectar la Giant Gecko Starter Kit a l'ordinador.
2. Obrir Simplicity Studio.
3. Importar el projecte des de la carpeta de materials.
4. Seleccionar la Giant Gecko com a dispositiu objectiu.
5. Compilar el projecte.
6. Programar la placa.
7. Iniciar l'execució o la depuració.

La prova es considera correcta quan el LED configurat canvia d'estat segons el programa i els polsadors provoquen el comportament definit.

Els paràmetres que es poden modificar són el GPIO assignat, el LED utilitzat, el polsador seleccionat i el temps de commutació.

A.6.2 Projecte ADC

Ruta del projecte:

Giant_Gecko\02_ADC

Aquest projecte permet llegir valors analògics, com els procedents del potenciòmetre i del termistor de la Giant Gecko Starter Kit.

Per executar-lo cal importar-lo a Simplicity Studio, compilar-lo, programar la placa i iniciar una sessió de depuració.

Durant la depuració es poden afegir les variables corresponents a la finestra de visualització de variables per observar els valors adquirits.

La prova es considera correcta quan els valors del potenciòmetre varien en modificar-ne la posició i la lectura del termistor canvia en funció de la temperatura.

Els canals ADC i les variables observades es poden adaptar al fitxer principal del projecte.

A.7 Comandes principals

En aquest apartat es recullen les comandes més importants utilitzades durant la preparació, compilació, programació i validació dels projectes.

A.7.1 Comandes generals d'ESP-IDF

Comanda	Funció
<code>cd RUTA_DEL_PROJECTE</code>	Permet accedir al directori del projecte que es vol compilar o programar.
<code>idf.py --version</code>	Mostra la versió d'ESP-IDF disponible a l'entorn.
<code>idf.py set-target esp32c6</code>	Configura l'ESP32-C6 com a dispositiu objectiu del projecte. Normalment només cal executar aquesta comanda la primera vegada que es prepara el projecte o quan es canvia de dispositiu.
<code>idf.py menuconfig</code>	Obre el menú de configuració del projecte. Permet modificar paràmetres com el nom i la contrasenya de la xarxa Wi-Fi o altres opcions específiques del projecte.
<code>idf.py build</code>	Compila el projecte i genera els fitxers necessaris per programar el dispositiu.
<code>idf.py -p COM9 flash</code>	Programa el projecte a la memòria flash de l'ESP32-C6. El port COM9 s'ha de substituir pel port assignat a la placa.
<code>idf.py -p COM9 monitor</code>	Obre el monitor sèrie del dispositiu sense tornar a programar la placa.
<code>idf.py -p COM9 flash monitor</code>	Programa el projecte i obre automàticament el monitor sèrie quan finalitza la programació.

Comanda	Funció
<code>idf.py fullclean</code>	Elimina els fitxers generats durant compilacions anteriors. Es pot utilitzar quan apareixen errors relacionats amb una configuració o compilació prèvia.
<code>Ctrl +]</code>	Tanca el monitor sèrie d'ESP-IDF i retorna el control a la terminal.

A.7.2 Comandes del terminal sèrie i de Python

Comanda	Funció
<code>python --version</code>	Mostra la versió de Python disponible dins de l'entorn d'ESP-IDF.
<code>python -m serial.tools.miniterm COM9 115200</code>	Obre el port sèrie indicat mitjançant <code>miniterm</code> . El primer paràmetre correspon al port COM i el segon a la velocitat de comunicació.
<code>Ctrl +]</code>	Tanca el terminal <code>miniterm</code> .
<code>python servidor_tcp.py</code>	Executa el servidor TCP auxiliar inclòs a la carpeta de materials. Aquesta comanda s'ha d'executar des del directori on es troba el fitxer.

A.7.3 Comandes principals d'ESP-AT

Les comandes AT s'han d'introduir des del terminal sèrie quan l'ESP32-C6 té carregat el firmware ESP-AT.

Comanda	Funció
<code>AT</code>	Comprova que la comunicació amb el firmware ESP-AT funciona. La resposta esperada és <code>OK</code> .
<code>AT+GMR</code>	Mostra la versió del firmware ESP-AT instal·lat i informació relacionada amb el dispositiu.
<code>AT+CWMODE=1</code>	Configura l'ESP32-C6 en mode Wi-Fi Station.

Comanda	Funció
AT+CWJAP="NOM.XARXA", CONTRASENYA"	Connecta el dispositiu a la xarxa Wi-Fi indicada. Cal substituir els valors pel nom i la contrasenya corresponents.
AT+CIFSR	Mostra les adreces IP associades a les interfícies de xarxa del dispositiu.
AT+CIPSTART="TCP", "ADRECA_IP",PORT	Obre una connexió TCP amb l'adreça IP i el port indicats.
AT+CIPSEND=LONGITUD	Indica el nombre de bytes que s'enviaran a través de la connexió TCP. Després d'obtenir el símbol >, cal escriure el missatge.
AT+CIPCLOSE	Tanca la connexió TCP activa.

Les comandes anteriors corresponen a les operacions utilitzades durant les proves del projecte. El firmware ESP-AT incorpora altres comandes que es poden consultar a la documentació oficial d'Espressif [39, 25, 26, 27].

A.8 Paràmetres que cal adaptar

Alguns dels valors utilitzats als exemples depenen de l'ordinador, de la xarxa disponible o de la placa utilitzada. Abans d'executar cada projecte, cal revisar els paràmetres indicats en aquest apartat.

A.8.1 Port de comunicació

Als exemples del manual s'utilitza habitualment el port:

COM9

Aquest valor correspon al port assignat a la placa durant les proves i pot ser diferent en altres ordinadors. Cal substituir-lo pel port identificat a l'Administrador de dispositius de Windows.

Per exemple, si la placa apareix connectada al port COM7, la comanda de programació serà:

```
idf.py -p COM7 flash monitor
```

A.8.2 Rutes dels projectes

Les rutes mostrades al manual són relatives a la carpeta principal de materials. La ubicació completa dependrà del directori on cada usuari hagi guardat la carpeta.

Per exemple, la ruta relativa:

```
ESP32-C6\03_WiFi_Station
```

pot correspondre a una ruta completa com:

```
C:\TFG\Codis_TFG\ESP32-C6\03_WiFi_Station
```

No cal utilitzar exactament aquesta ubicació. Es recomana evitar rutes excessivament llargues, espais i caràcters especials, ja que poden provocar problemes en algunes eines de compilació.

A.8.3 Configuració de la xarxa Wi-Fi

Els projectes Wi-Fi Station i ESP-AT requereixen indicar:

- El nom de la xarxa Wi-Fi.
- La contrasenya de la xarxa.

En el projecte Wi-Fi Station, aquests valors es configuren mitjançant:

```
idf.py menuconfig
```

En el firmware ESP-AT, les credencials s'introdueixen directament mitjançant la comanda:

```
AT+CWJAP="NOM_XARXA","CONTRASENYA"
```

Els valors `NOM_XARXA` i `CONTRASENYA` s'han de substituir per les dades de la xarxa utilitzada.

Per seguretat, no s'han d'incloure credencials personals dins dels projectes lliurats.

A.8.4 Configuració del punt d'accés Wi-Fi

En el projecte Wi-Fi SoftAP es poden adaptar els paràmetres següents:

- El nom de la xarxa creada.
- La contrasenya.
- El canal Wi-Fi.
- El nombre màxim de clients connectats.

Aquests valors es configuren mitjançant el menú del projecte:

```
idf.py menuconfig
```

La contrasenya ha de complir els requisits mínims establerts pel protocol de seguretat utilitzat.

A.8.5 Adreça IP i port del servidor TCP

Per obrir una connexió TCP cal adaptar els valors de la comanda següent:

```
AT+CIPSTART="TCP", "ADRECA_IP", PORT
```

El valor `ADRECA_IP` correspon a l'adreça IPv4 de l'ordinador on s'executa el servidor TCP.

El valor `PORT` correspon al port configurat dins del fitxer del servidor:

```
servidor_tcp.py
```

L'ordinador i l'ESP32-C6 han d'estar connectats a la mateixa xarxa local. També cal comprovar que el tallafoc de Windows permet les connexions entrants de Python al port seleccionat.

A.8.6 Longitud del missatge TCP

La comanda:

```
AT+CIPSEND=LONGITUD
```

indica el nombre de bytes que s'enviaran a través de la connexió TCP.

El valor `LONGITUD` ha de coincidir amb el nombre de bytes del missatge que s'introduirà posteriorment. Si la longitud indicada és superior a les dades escrites, el dispositiu continuarà esperant més caràcters. Si és inferior, el missatge quedarà incomplet.

A.8.7 Pins GPIO

Els números de GPIO utilitzats als projectes de parpelleig, control de LED o lectura de pulsadors poden variar segons la placa o el circuit utilitzat.

Abans de modificar-los, cal consultar:

- L'esquemàtic de la placa.
- La distribució de pins del mòdul.
- La configuració del projecte.

No s'ha d'assignar un GPIO sense comprovar prèviament si està connectat a un altre perifèric o si té una funció especial durant l'arrencada del dispositiu.

A.8.8 Paràmetres Bluetooth Low Energy

Al projecte Bluetooth Low Energy es poden modificar els paràmetres següents:

- El nom anunciat pel dispositiu.
- Els identificadors UUID del servei i de la característica.
- Els permisos de lectura i escriptura.
- El tractament de les dades rebudes.

Qualsevol canvi en els identificadors GATT s'haurà de tenir en compte quan s'accedeixi al dispositiu mitjançant nRF Connect.

A.8.9 Configuració UART d'ESP-AT

La configuració del port de comandes d'ESP-AT inclou:

- El número de la UART.
- El pin de transmissió TX.
- El pin de recepció RX.
- La velocitat de comunicació.

- Els pins de control de flux, si s'utilitzen.

Els projectes lliurats incorporen la configuració utilitzada durant el treball. Aquests valors no s'han de modificar sense comprovar abans les connexions de la placa i la disponibilitat dels GPIO.

La velocitat configurada al terminal ha de coincidir amb la velocitat definida al firmware. En els exemples del manual s'utilitza:

115200

Si les dues velocitats no coincideixen, els caràcters rebuts poden aparèixer incorrectament o no es pot establir la comunicació.

A.8.10 Paràmetres de la Giant Gecko

Als projectes de la Giant Gecko poden variar:

- Els pins assignats als LED i als polsadors.
- Els canals utilitzats per l'ADC.
- Les variables observades durant la depuració.
- Els pins de les interfícies de comunicació.

Abans de modificar aquests valors, cal comprovar la documentació de la Giant Gecko Starter Kit i la configuració inclosa al projecte corresponent.

A.8.11 Enllaç als materials

Els projectes, els fitxers de configuració, els scripts auxiliars i la documentació complementària es poden consultar i descarregar mitjançant la carpeta compartida següent:

Accés a la carpeta de materials del projecte

La carpeta s'ha configurat amb permisos de lectura per a qualsevol persona que disposi de l'enllaç. Els materials s'organitzen segons la plataforma i el tipus de prova, tal com s'ha descrit als apartats anteriors.

Bibliografia

- [1] A. Rafecas Rubió, “Disseny d’una placa docent basada en la Teensy 3.5,” Treball de Fi de Grau, Escola d’Enginyeria, UAB, Cerdanyola del Vallès, 2023.
- [2] Teensy 3.6. PJRC. [En línia]. Disponible a: <https://www.pjrc.com/store/teensy36.html>
- [3] Teensy technical specifications. PJRC. [En línia]. Disponible a: <https://www.pjrc.com/teensy/techspecs.html>
- [4] Teensy datasheets. PJRC. [En línia]. Disponible a: <https://www.pjrc.com/teensy/datasheets.html>
- [5] EFM32GG starter kit. Silicon Labs. [En línia]. Disponible a: <https://www.silabs.com/development-tools/mcu/32-bit/efm32gg-starter-kit>
- [6] Simplicity studio. Silicon Labs. [En línia]. Disponible a: <https://www.silabs.com/software-and-tools/simplicity-studio>
- [7] SoCs. Espressif Systems. [En línia]. Disponible a: <https://www.espressif.com/en/products/socs>
- [8] Viquipèdia. ESP32. [En línia]. Disponible a: <https://es.wikipedia.org/wiki/ESP32>
- [9] Modules. Espressif Systems. [En línia]. Disponible a: <https://www.espressif.com/en/products/modules>
- [10] ESP-IDF programming guide (ESP32-C6). Espressif Systems. [En línia]. Disponible a: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32c6/index.html>
- [11] ESP-AT user guide (ESP32-C6). Espressif Systems. [En línia]. Disponible a: <https://docs.espressif.com/projects/esp-at/en/latest/esp32c6/index.html>
- [12] ESP32 SoC options. ESP Boards. [En línia]. Disponible a: <https://www.espboards.dev/blog/esp32-soc-options/>
- [13] ESP32-C6. Espressif Systems. [En línia]. Disponible a: <https://www.espressif.com/en/products/socs/esp32-c6>
- [14] (2024) ESP32-C6-WROOM-1 & ESP32-C6-WROOM-1U datasheet. Espressif Systems. [En línia]. Disponible a: https://www.espressif.com/sites/default/files/documentation/esp32-c6-wroom-1_wroom-1u_datasheet_en.pdf
- [15] ESP32-C6-WROOM-1. Espressif Systems. [En línia]. Disponible a: <https://www.espressif.com/en/producttype/esp32-c6-wroom-1>

- [16] (2024) ESP32-C6-MINI-1 & ESP32-C6-MINI-1U datasheet. Espressif Systems. [En línia]. Disponible a: https://www.espressif.com/sites/default/files/documentation/esp32-c6-mini-1_mini-1u_datasheet_en.pdf
- [17] Universal asynchronous receiver/transmitter (UART). Espressif Systems. [En línia]. Disponible a: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32c6/api-reference/peripherals/uart.html>
- [18] USB serial/JTAG controller console. Espressif Systems. [En línia]. Disponible a: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32c6/api-guides/usb-serial-jtag-console.html>
- [19] ESP-IDF get started (ESP32-C6). Espressif Systems. [En línia]. Disponible a: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32c6/get-started/index.html>
- [20] ESP-IDF examples. Espressif Systems. [En línia]. Disponible a: <https://github.com/espressif/esp-idf/tree/master/examples>
- [21] nRF connect for mobile. Nordic Semiconductor. [En línia]. Disponible a: <https://www.nordicsemi.com/Products/Development-tools/nRF-Connect-for-mobile>
- [22] What is ESP-AT. Espressif Systems. [En línia]. Disponible a: https://docs.espressif.com/projects/esp-at/en/latest/esp32c6/Get_Started/What_is_ESP-AT.html
- [23] ESP-AT. Espressif Systems. [En línia]. Disponible a: <https://github.com/espressif/esp-at>
- [24] ESP-IDF hello world example. Espressif Systems. [En línia]. Disponible a: https://github.com/espressif/esp-idf/tree/master/examples/get-started/hello_world
- [25] Basic AT commands. Espressif Systems. [En línia]. Disponible a: https://docs.espressif.com/projects/esp-at/en/latest/esp32c6/AT_Command_Set/Basic_AT_Commands.html
- [26] Wi-Fi AT commands. Espressif Systems. [En línia]. Disponible a: https://docs.espressif.com/projects/esp-at/en/latest/esp32c6/AT_Command_Set/Wi-Fi_AT_Commands.html
- [27] TCP-IP AT commands. Espressif Systems. [En línia]. Disponible a: https://docs.espressif.com/projects/esp-at/en/latest/esp32c6/AT_Command_Set/TCP-IP_AT_Commands.html
- [28] ESP-IDF installation manager. Espressif Systems. [En línia]. Disponible a: <https://dl.espressif.com/dl/eim/>
- [29] ESP-IDF windows setup. Espressif Systems. [En línia]. Disponible a: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/get-started/windows-setup.html>
- [30] Simplicity studio 5. Silicon Labs. [En línia]. Disponible a: <https://www.silabs.com/software-and-tools/simplicity-studio/simplicity-studio-version-5>
- [31] Install SS-5 and software. Silicon Labs. [En línia]. Disponible a: <https://docs.silabs.com/simplicity-studio-5-users-guide/latest/ss-5-users-guide-getting-started/install-ss-5-and-software>
- [32] Establish serial connection with ESP32-C6. Espressif Systems. [En línia]. Disponible a: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32c6/get-started/establish-serial-connection.html>

-
- [33] ESP-IDF blink example. Espressif Systems. [En línia]. Disponible a: <https://github.com/espressif/esp-idf/tree/master/examples/get-started/blink>
 - [34] Wi-Fi station example. Espressif Systems. [En línia]. Disponible a: https://github.com/espressif/esp-idf/tree/master/examples/wifi/getting_started/station
 - [35] Wi-Fi SoftAP example. Espressif Systems. [En línia]. Disponible a: https://github.com/espressif/esp-idf/tree/master/examples/wifi/getting_started/softAP
 - [36] NimBLE peripheral (bleprph) example. Espressif Systems. [En línia]. Disponible a: <https://github.com/espressif/esp-idf/tree/master/examples/bluetooth/nimble/bleprph>
 - [37] BLE introduction. Espressif Systems. [En línia]. Disponible a: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32c6/api-guides/ble/get-started/ble-introduction.html>
 - [38] How to clone project and compile it. Espressif Systems. [En línia]. Disponible a: https://docs.espressif.com/projects/esp-at/en/latest/esp32c6/Compile_and_Develop/How_to_clone_project_and_compile_it.html
 - [39] AT command set. Espressif Systems. [En línia]. Disponible a: https://docs.espressif.com/projects/esp-at/en/latest/esp32c6/AT_Command_Set/index.html