



UNIVERSIDAD
NACIONAL
DE LA PLATA

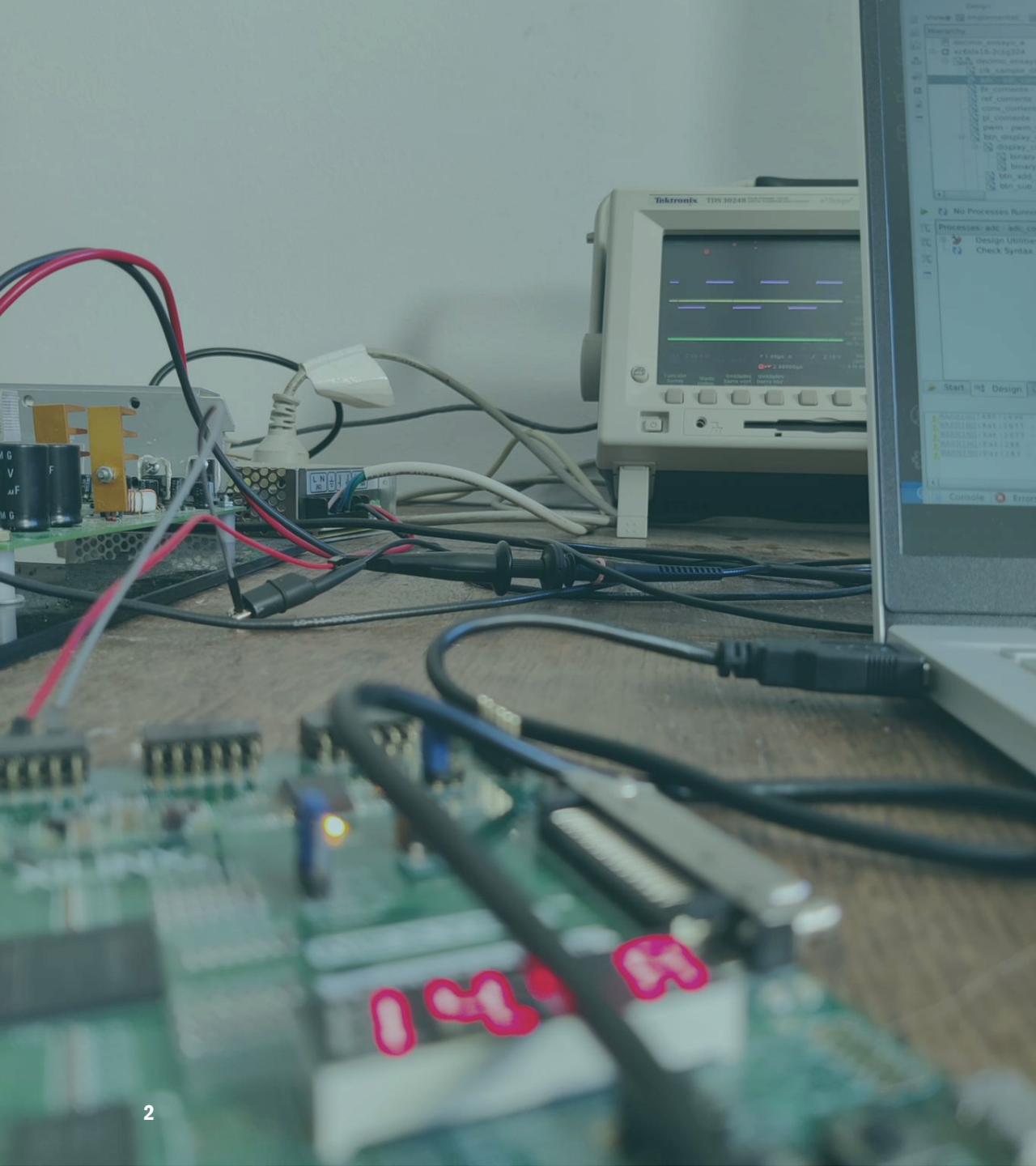
IMPLEMENTACIÓN DEL CONTROL AUTOMÁTICO DE UN SISTEMA HÍBRIDO ELÉCTRICO EN FPGA

Alex Benjamín Gubkien

Jerónimo José Moré

Paul Puleston

Instituto LEICI, Departamento de Electrotecnia



ÍNDICE

Motivación

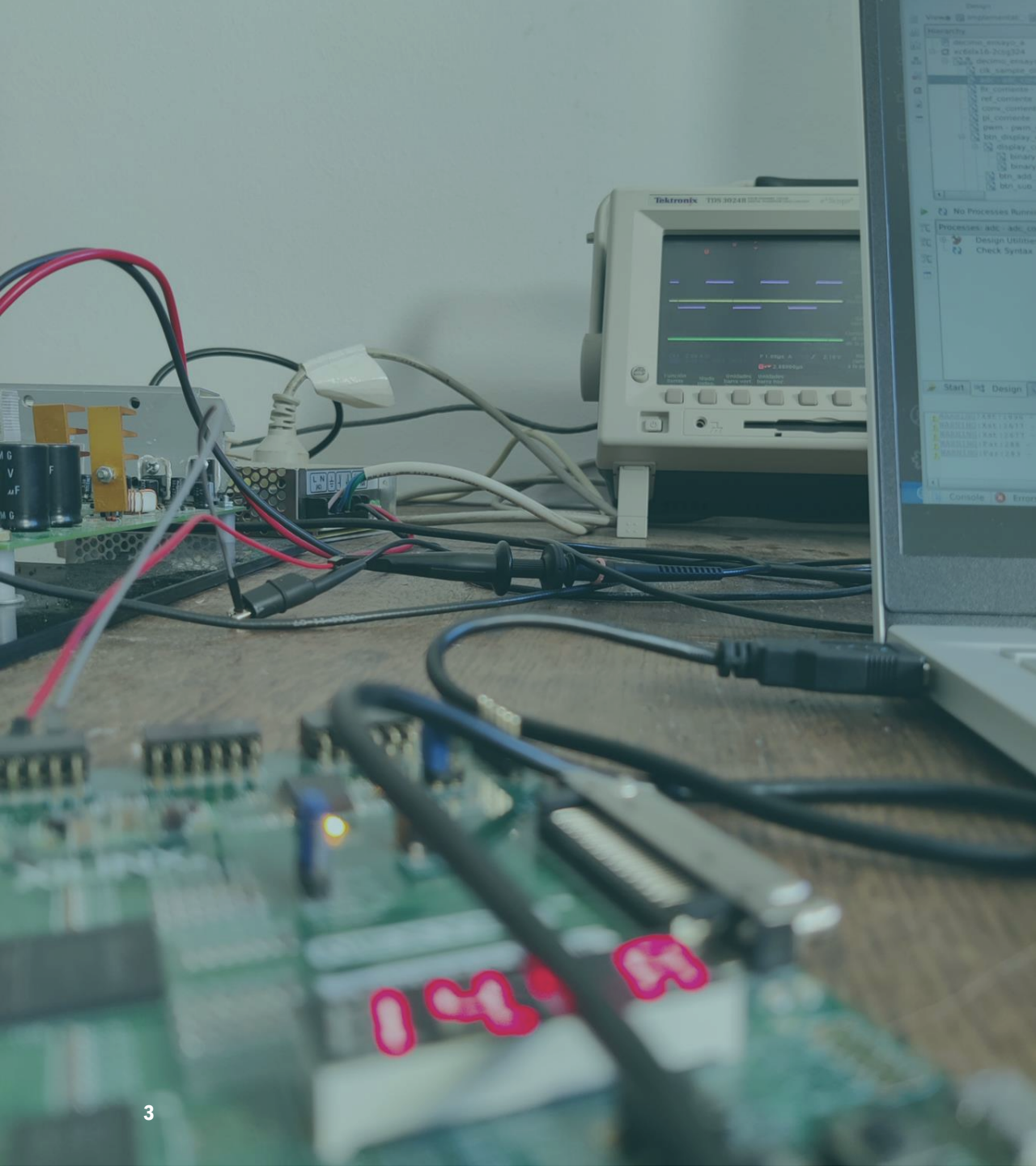
Investigación

Diseño

Implementación

Ensayos

Conclusiones



ÍNDICE

Motivación

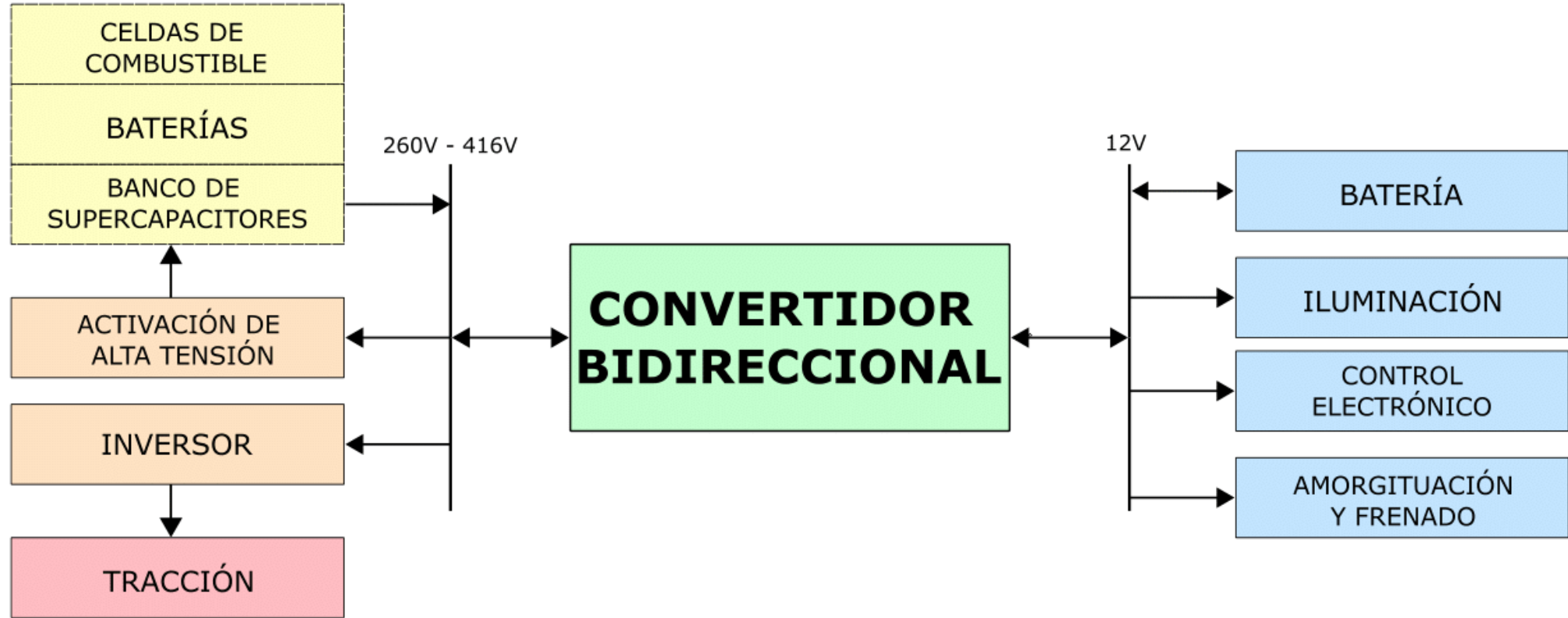
Investigación

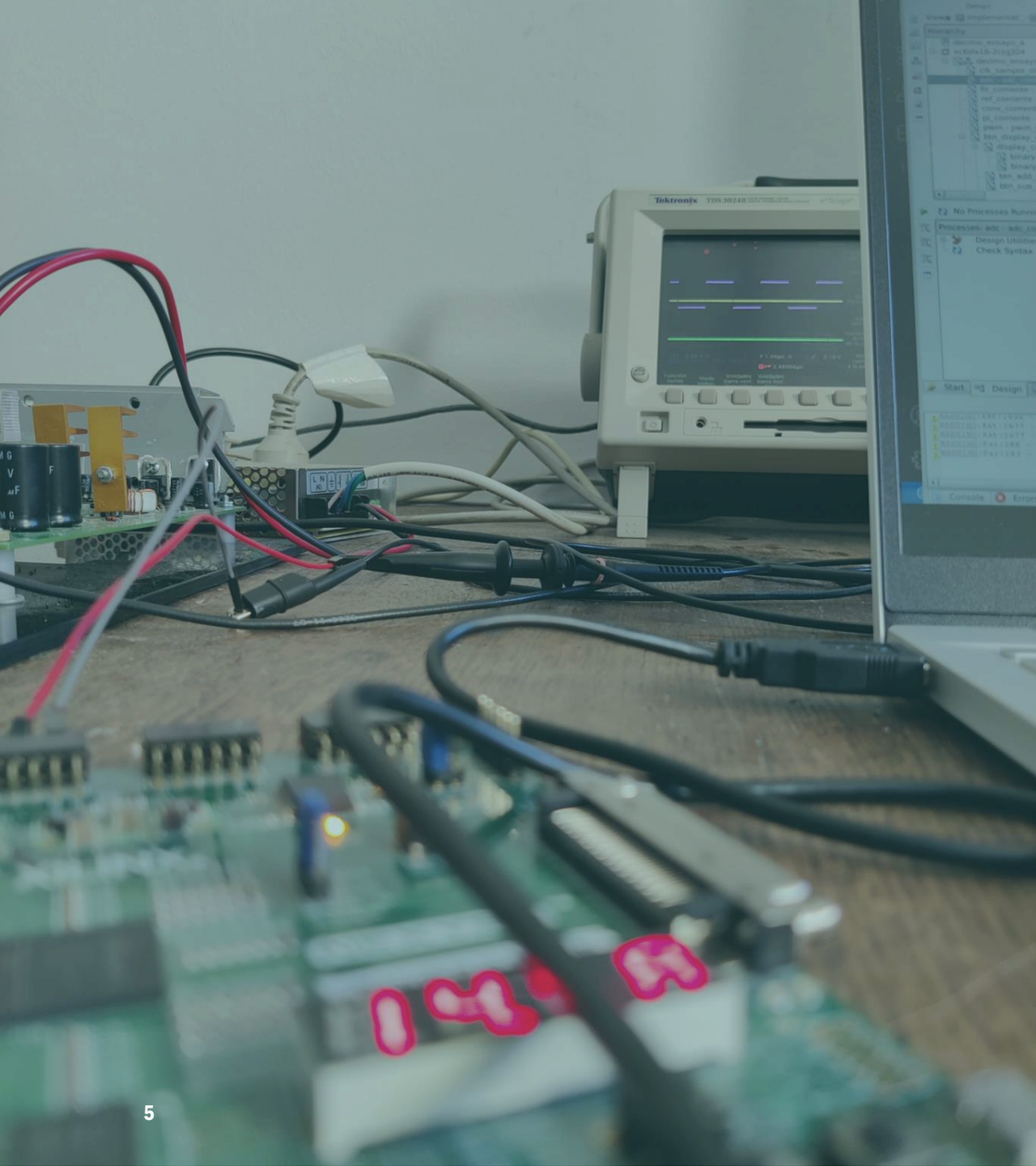
Diseño

Implementación

Ensayos

Conclusiones





ÍNDICE

Motivación

Investigación

Diseño

Implementación

Ensayos

Conclusiones

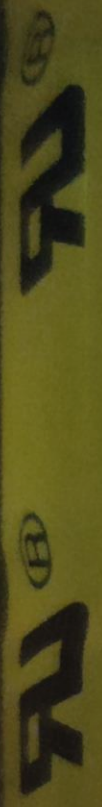
INVESTIGACIÓN

La primer etapa del proyecto consistió en el estudio en profundidad de los módulos de almacenamiento del sistema eléctrico híbrido, los convertidores electrónicos de potencia, y la arquitectura FPGA con sus lenguajes HDL.



AEROGEL CAPACITOR AEROGEL

Y11X346-01 Y11X346-01 Y14043-01 Y14043-01



PowerStor[®] Powers

AEROGEL CAPACITOR AEROGEL CAPACITOR
Y11X346-01 Y11X346-01 Y11X346-01 Y11X346-01

XV Series XV Series

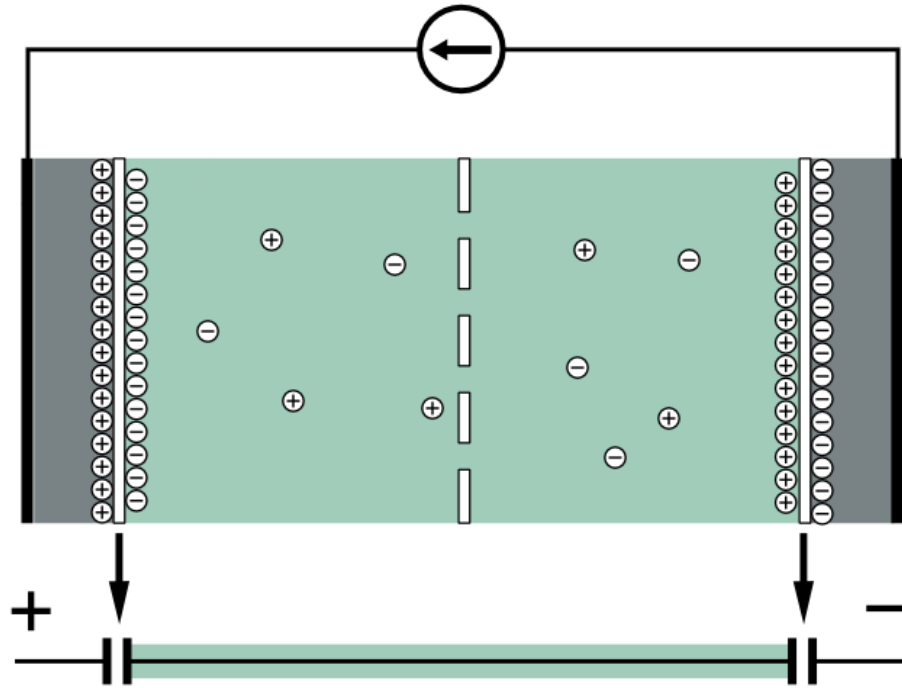


0.41Wh 2.7V 400F 0.41Wh 2.7V 400F

XV Series XV Series

SUPERCAPACITORS

INVESTIGACIÓN: SUPERCAPACITORES

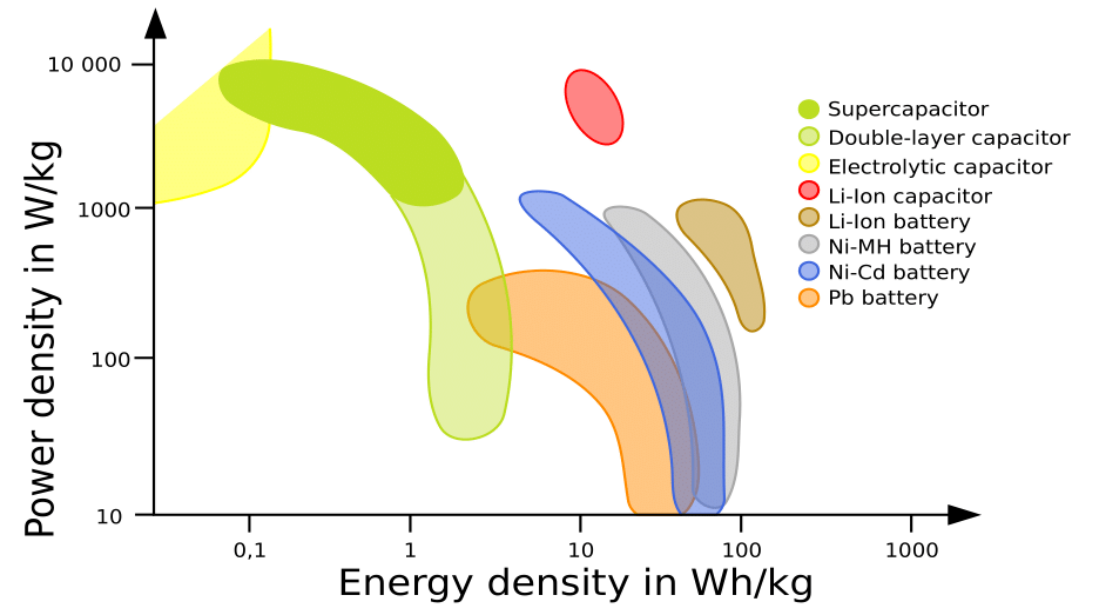
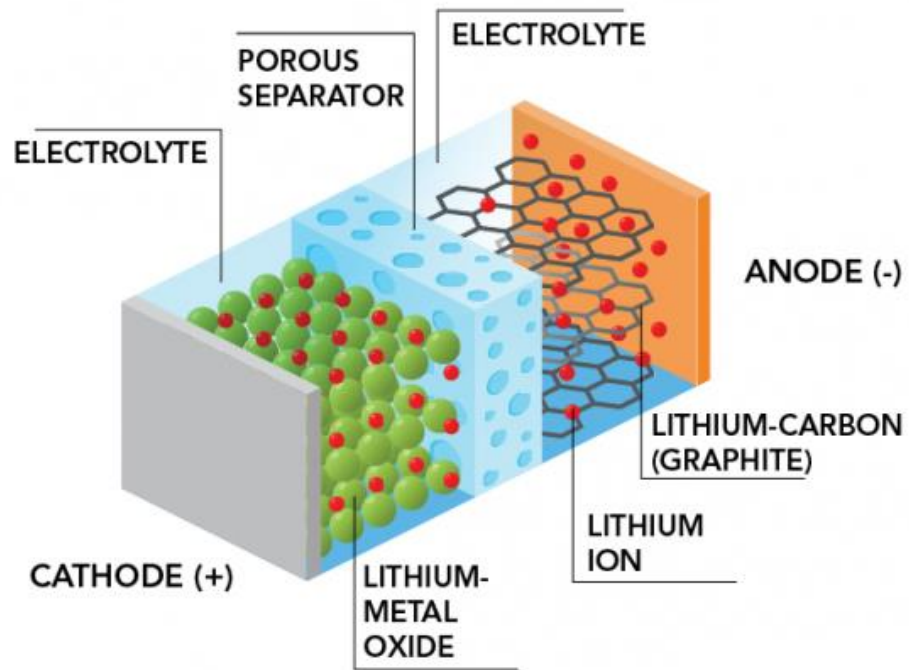


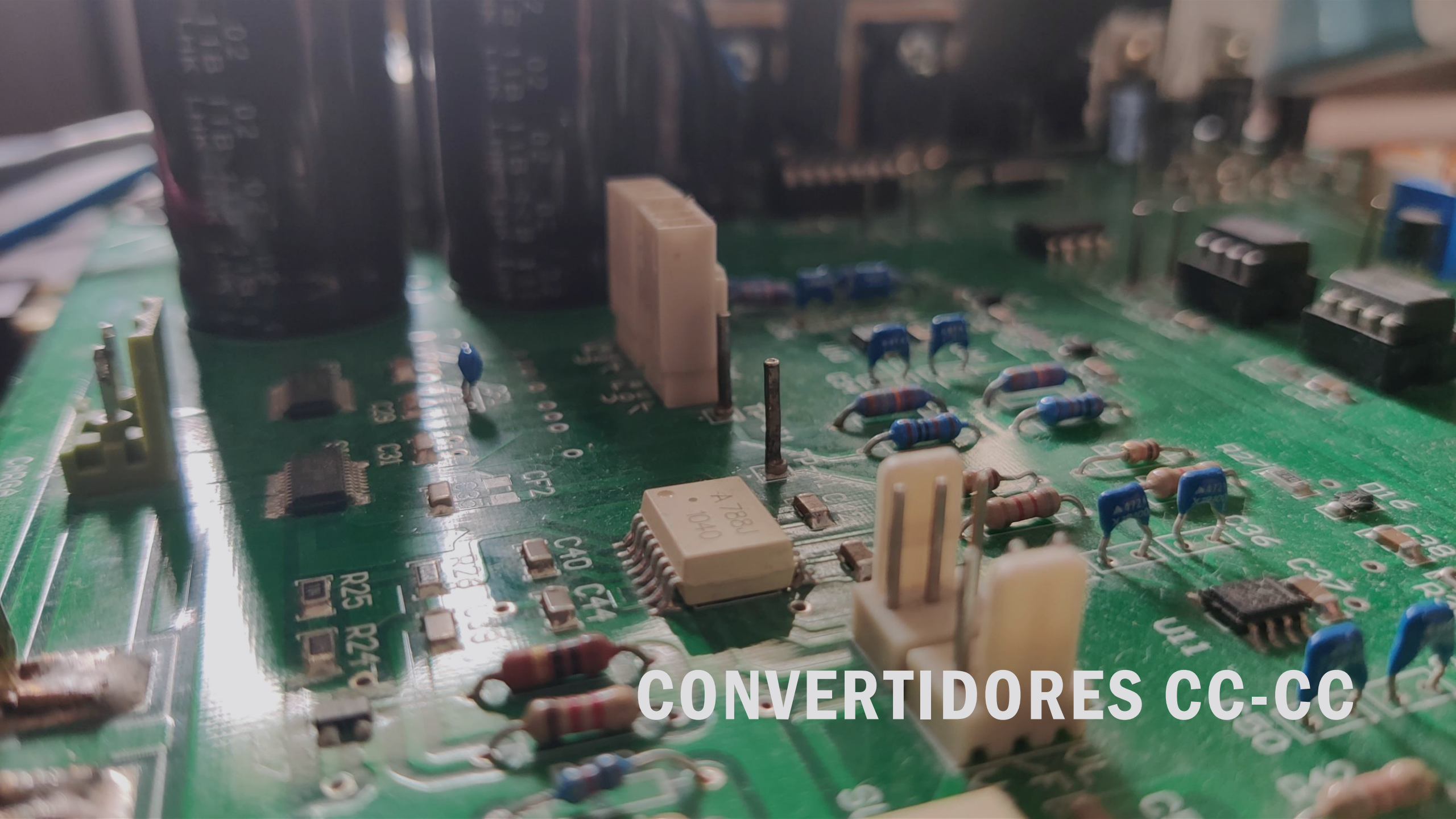
$$C = \epsilon \frac{A}{d}$$



BATERÍAS DE LITIO

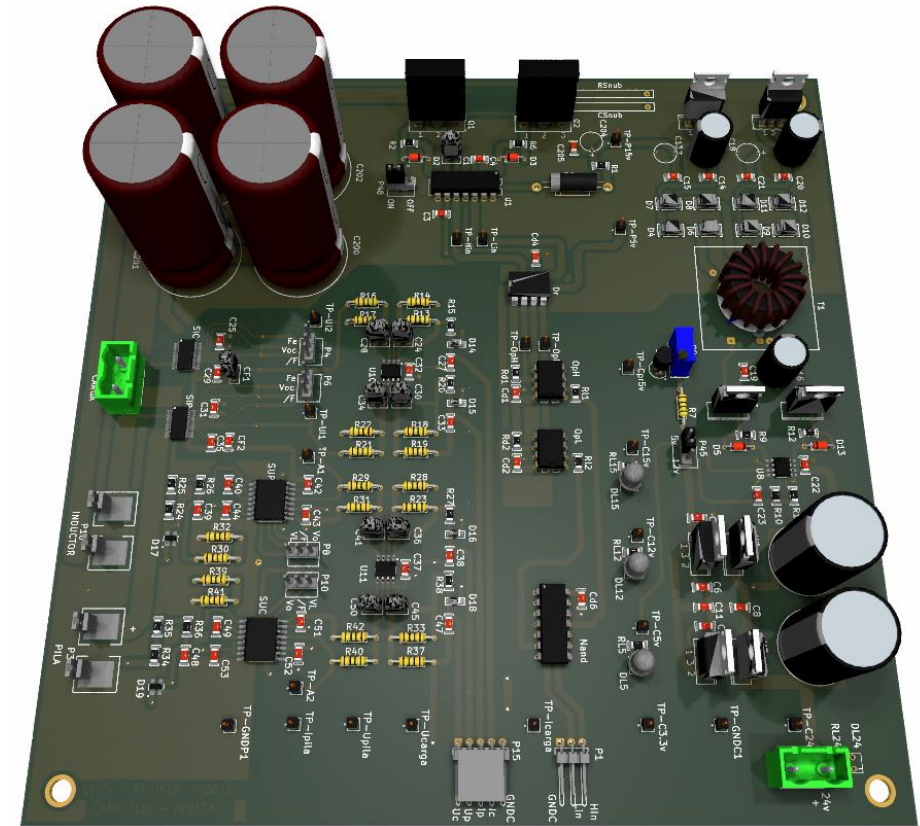
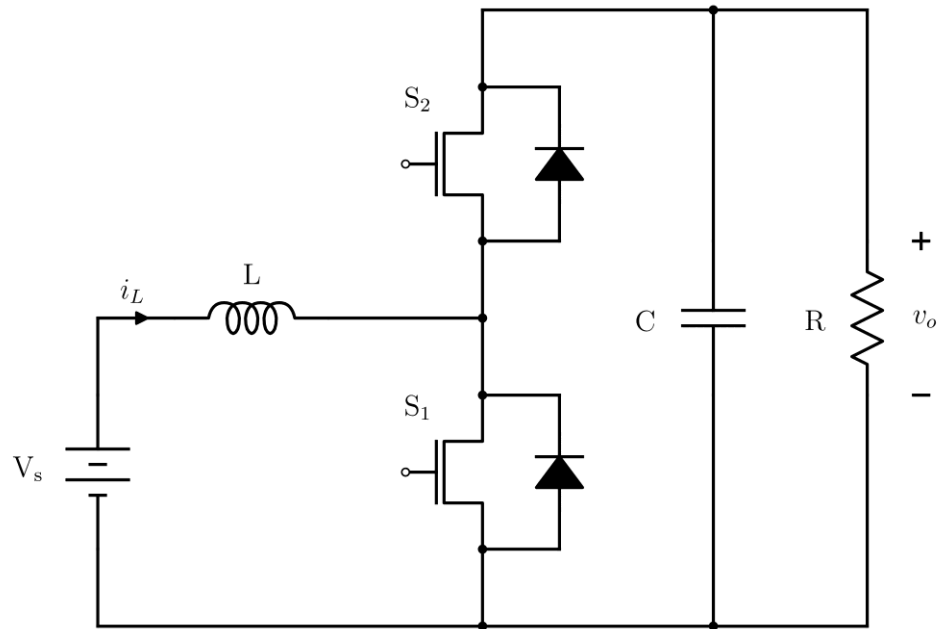
INVESTIGACIÓN: BATERÍAS DE LITIO

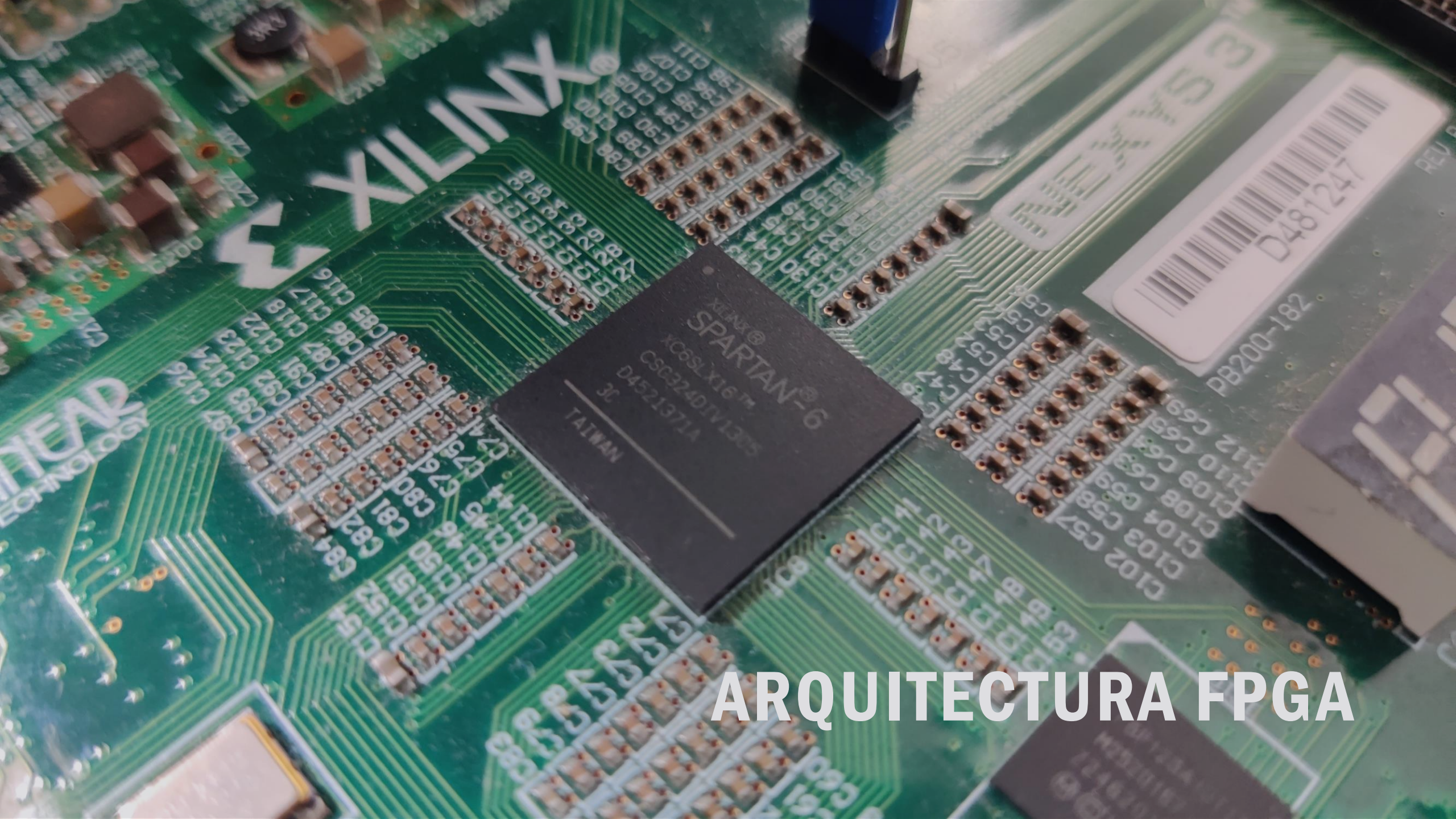




CONVERTIDORES CC-CC

INVESTIGACIÓN: CONVERTIDORES CC-CC





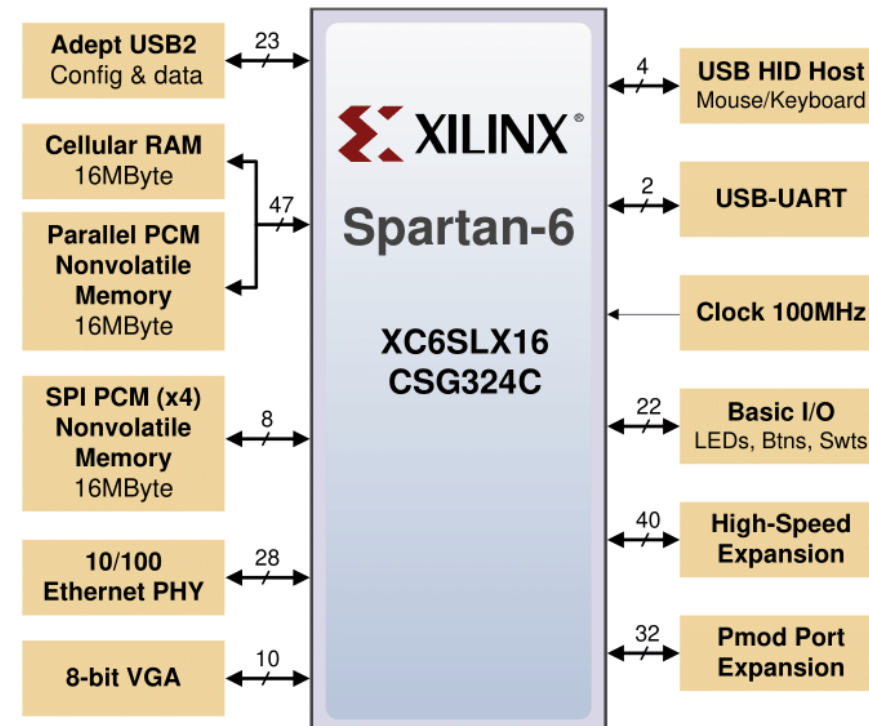
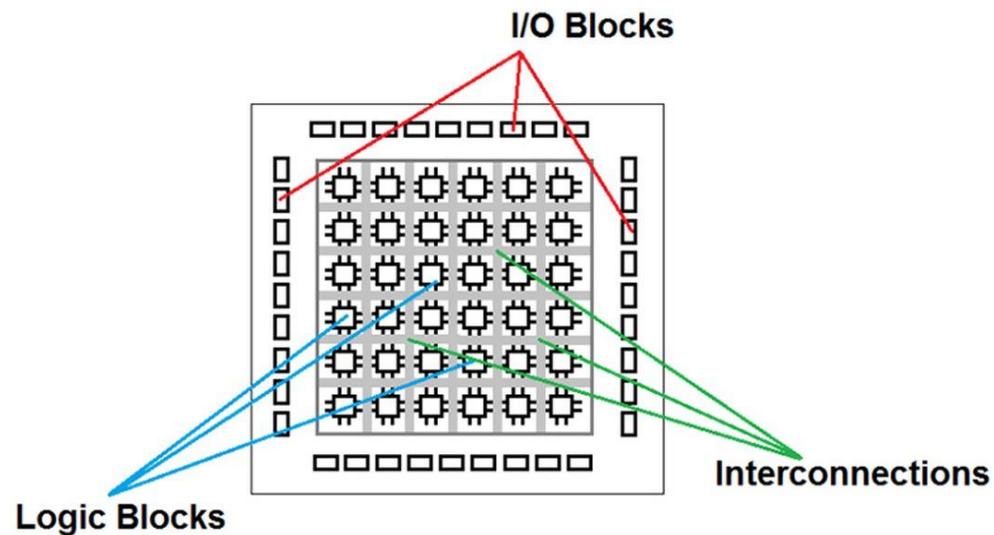
XILINX®
SPARTAN-6
XC6SLX16™
CSC3240TV1325
D481247
TAIWAN

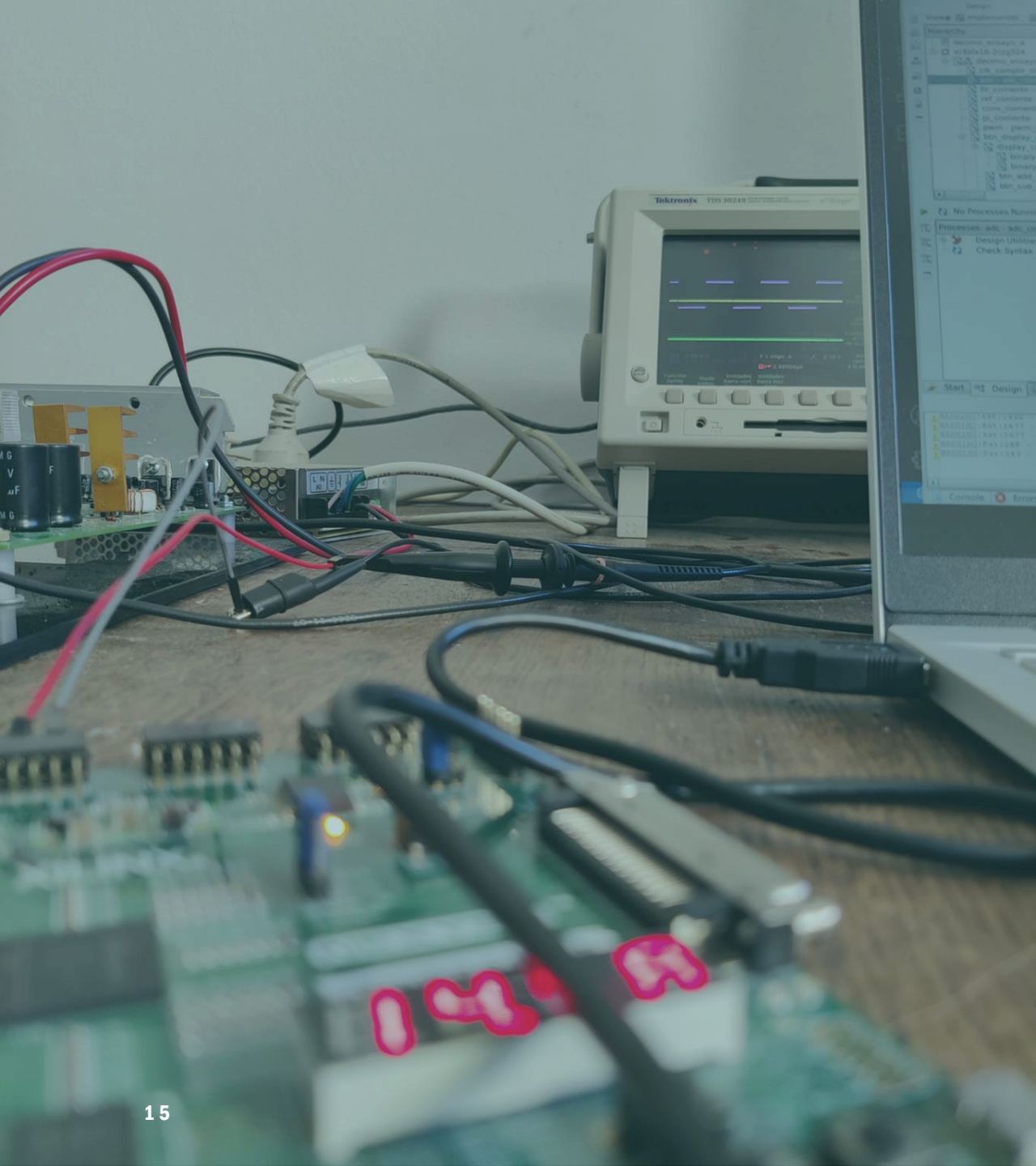
PB200-182
D481247

ARQUITECTURA FPGA

INVESTIGACIÓN: ARQUITECTURA FPGA

FIELD-PROGRAMMABLE GATE ARRAY





ÍNDICE

Motivación

Investigación

Diseño

Implementación

Ensayos

Conclusiones

DISEÑO

OBJETIVOS

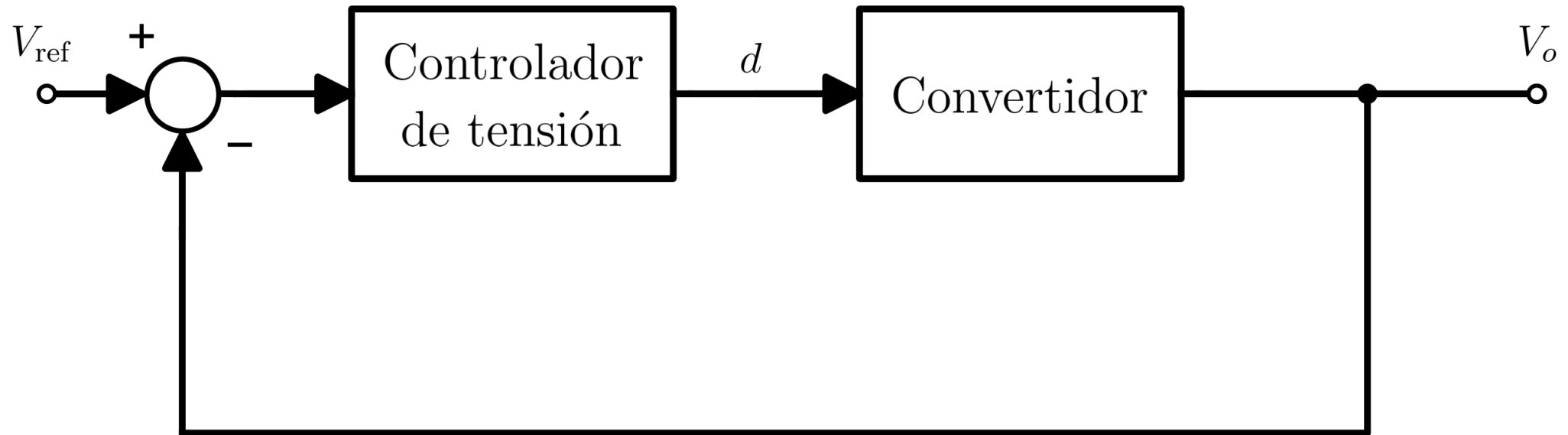
Diseñar un sistema que permita un control de la tensión de carga mediante una referencia establecida manualmente

Asegurar la estabilidad de las variables de estado del convertidor electrónico de potencia

CONTROLADORES UTILIZADOS

Controladores PI+D

DISEÑO: SISTEMA DE CONTROL



SISTEMA DE CONTROL: DINÁMICA CERO

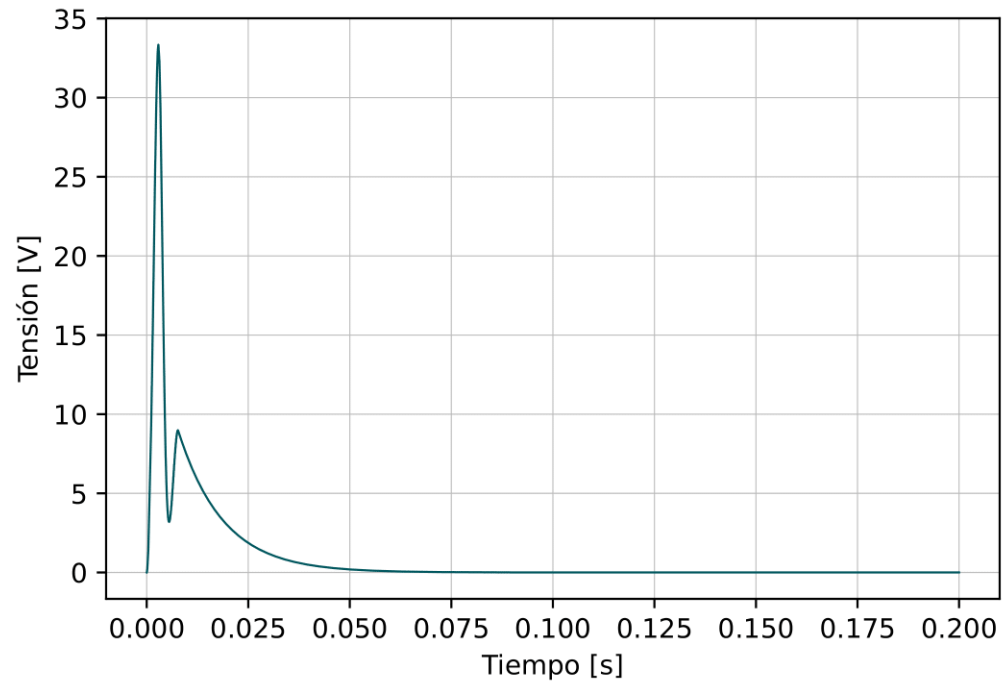
Aunque la salida esté fija en un valor constante, internamente continúa existiendo una **dinámica escondida** o **dinámica cero** asociada a los estados restantes.

Inestabilidad de la corriente de entrada al convertidor.

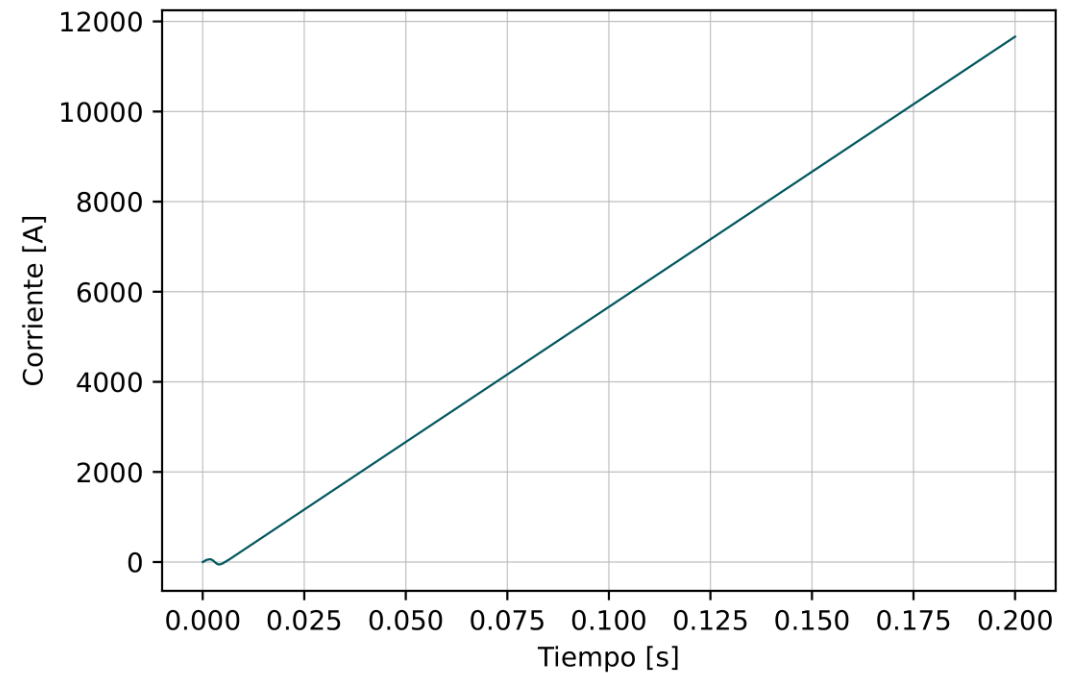
SISTEMA DE CONTROL: DINÁMICA CERO

REGULACIÓN DE TENSION DE SALIDA A 50V

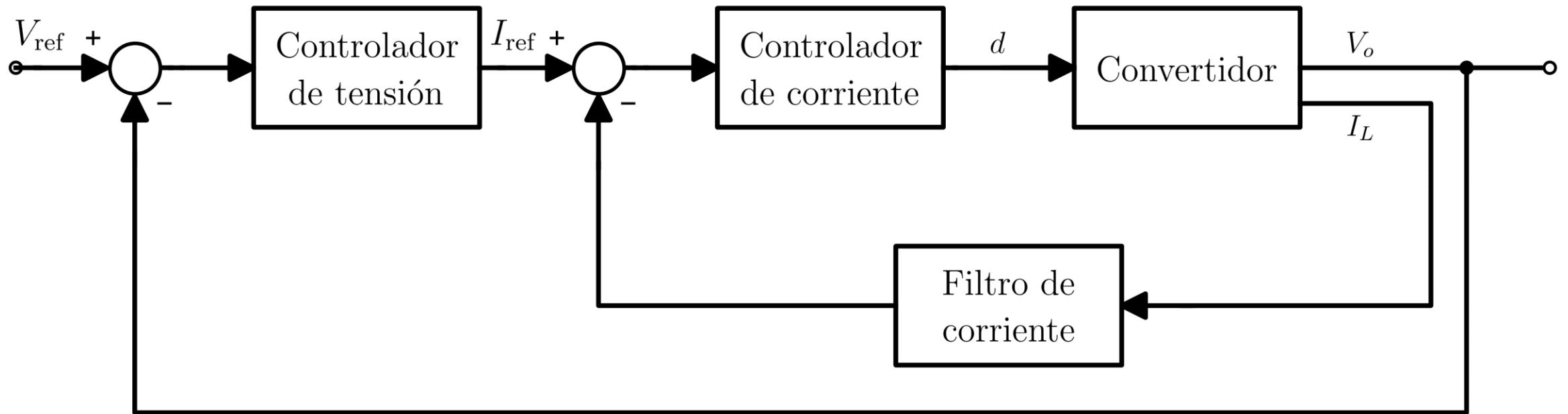
Tensión en la salida



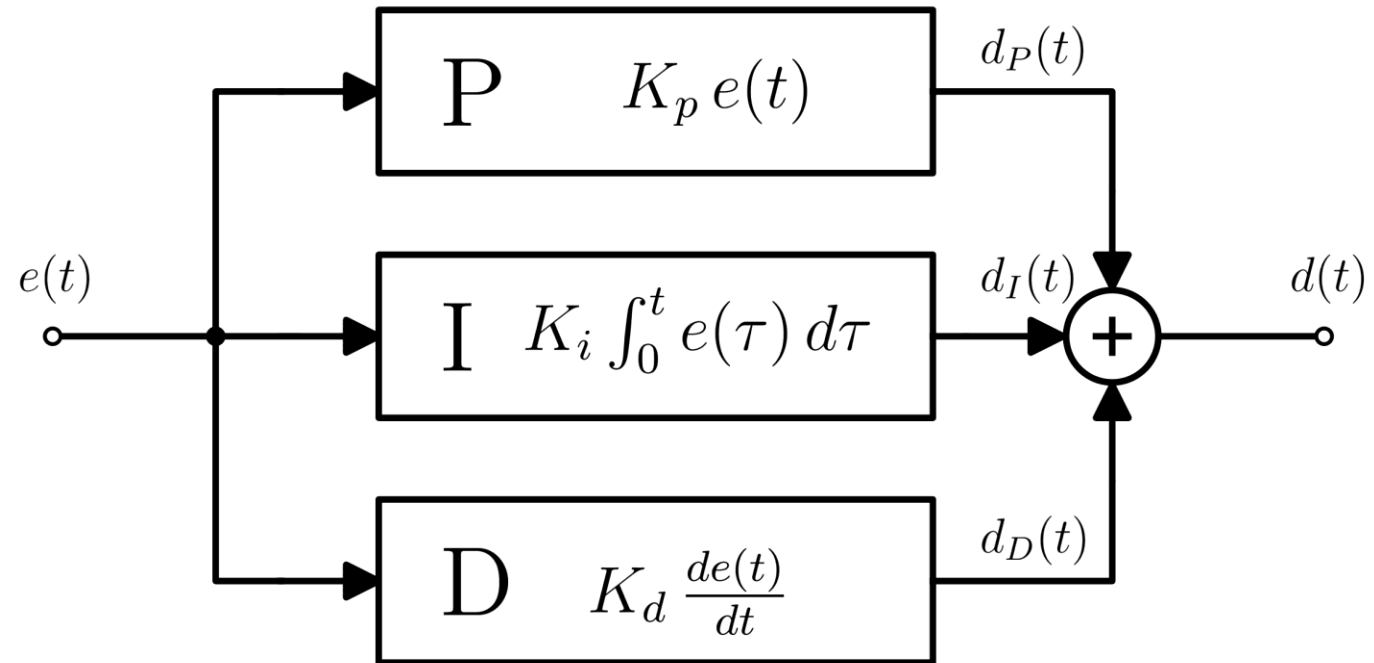
Corriente por el inductor



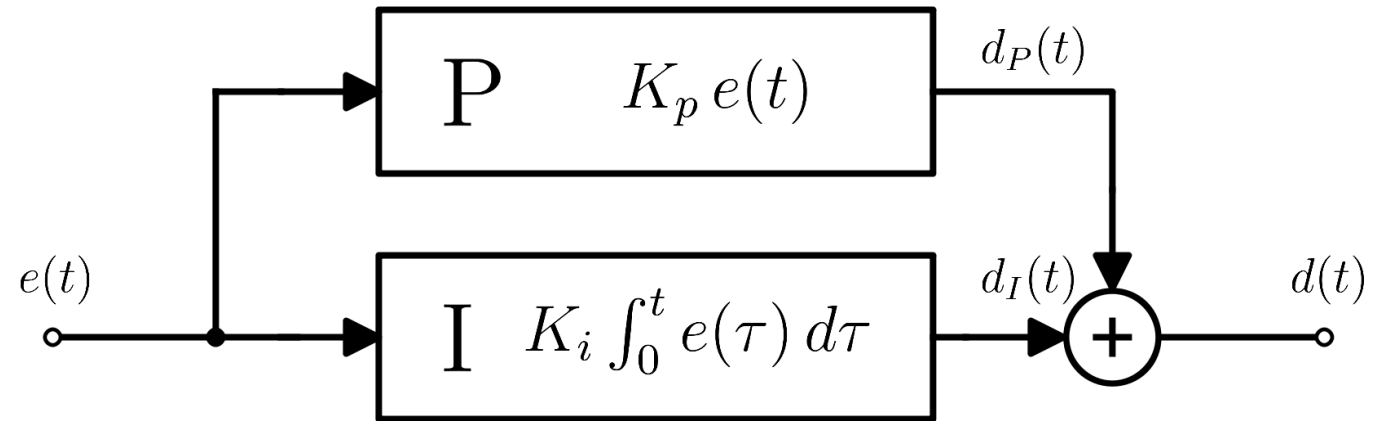
DISEÑO: SISTEMA DE CONTROL



DISEÑO: CONTROLADORES PI+D

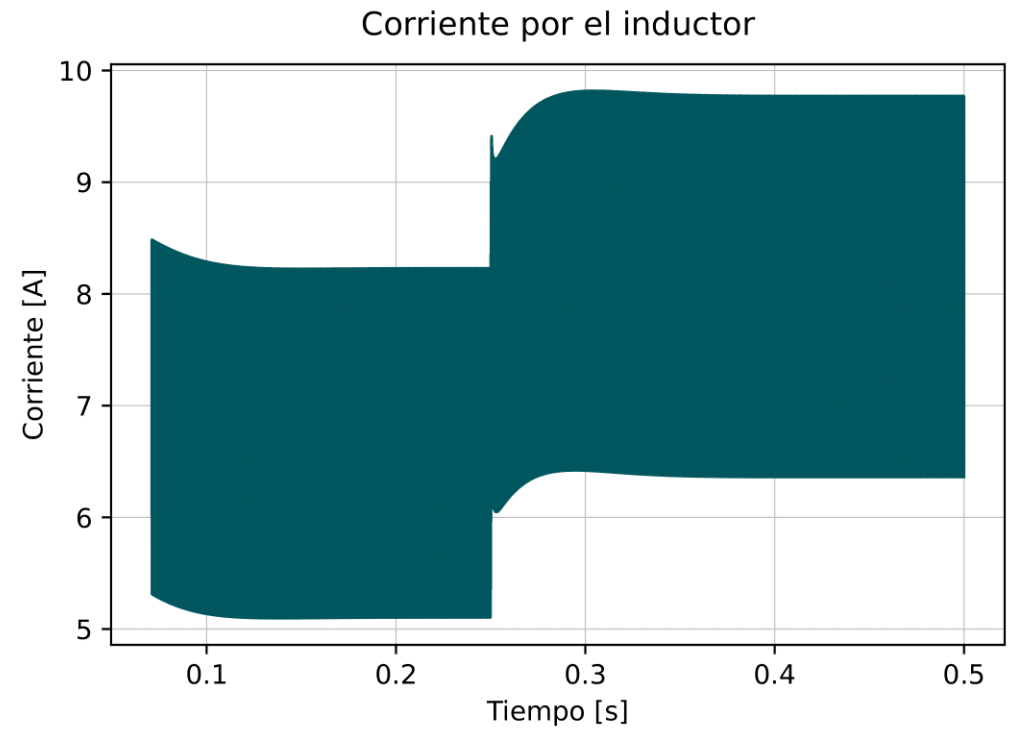
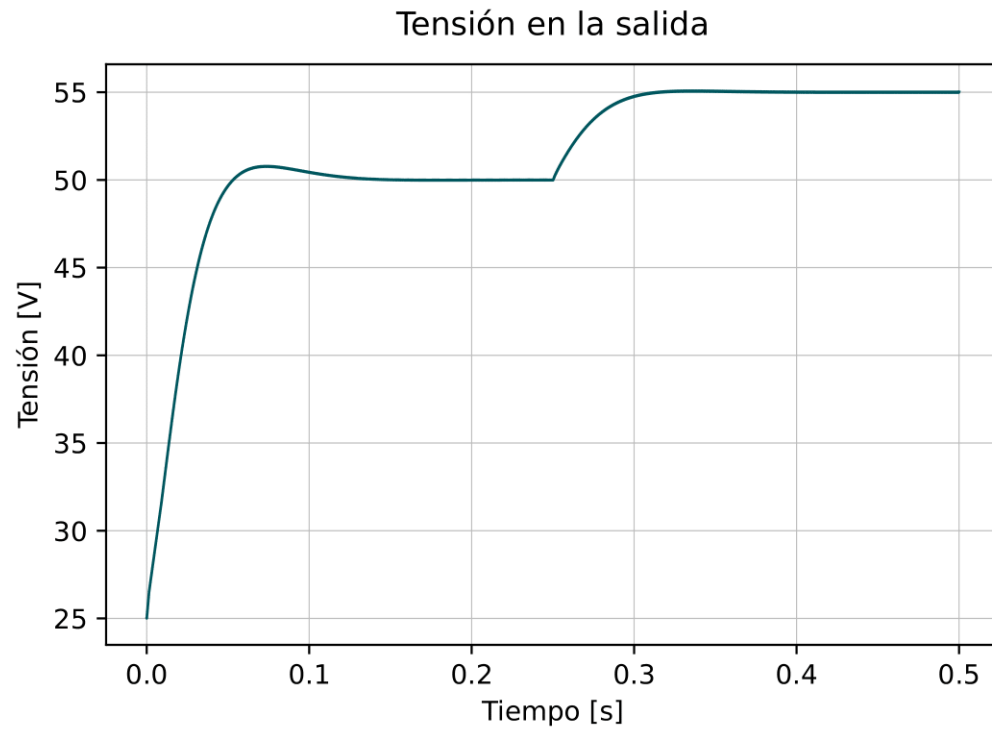


DISEÑO: CONTROLADORES PI



SISTEMA DE CONTROL: SIMULACIONES

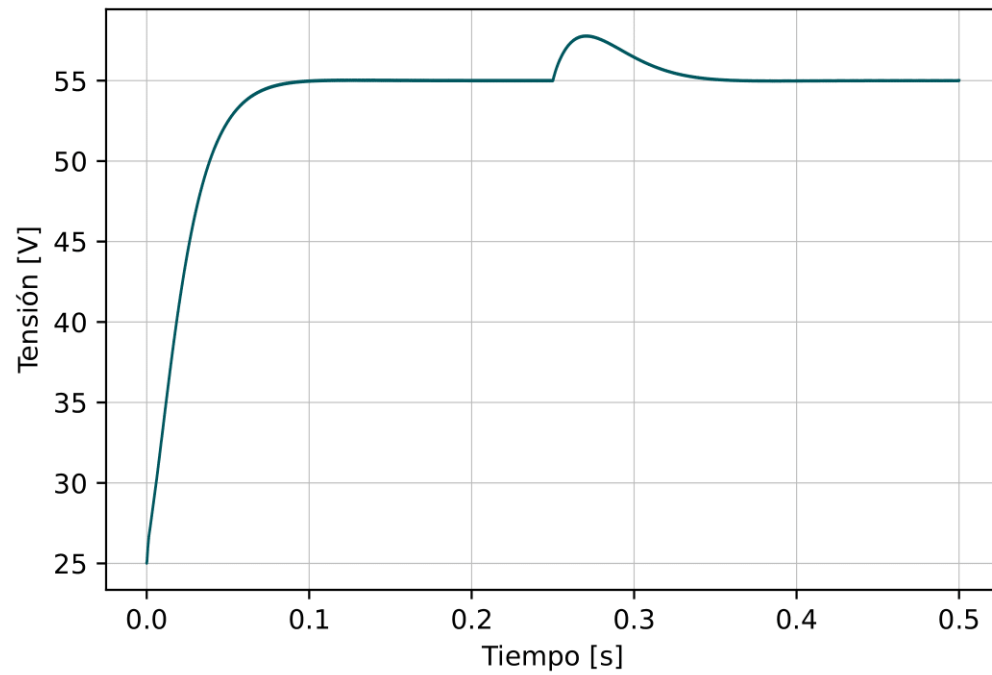
ESCALÓN DE TENSIÓN DE 50V A 55V



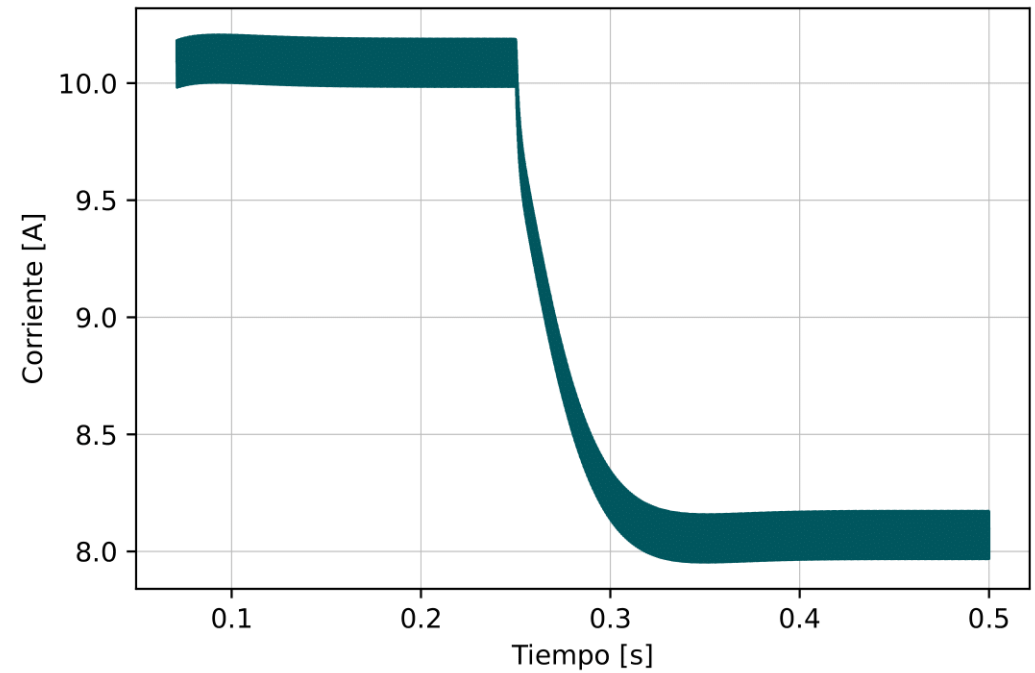
SISTEMA DE CONTROL: SIMULACIONES

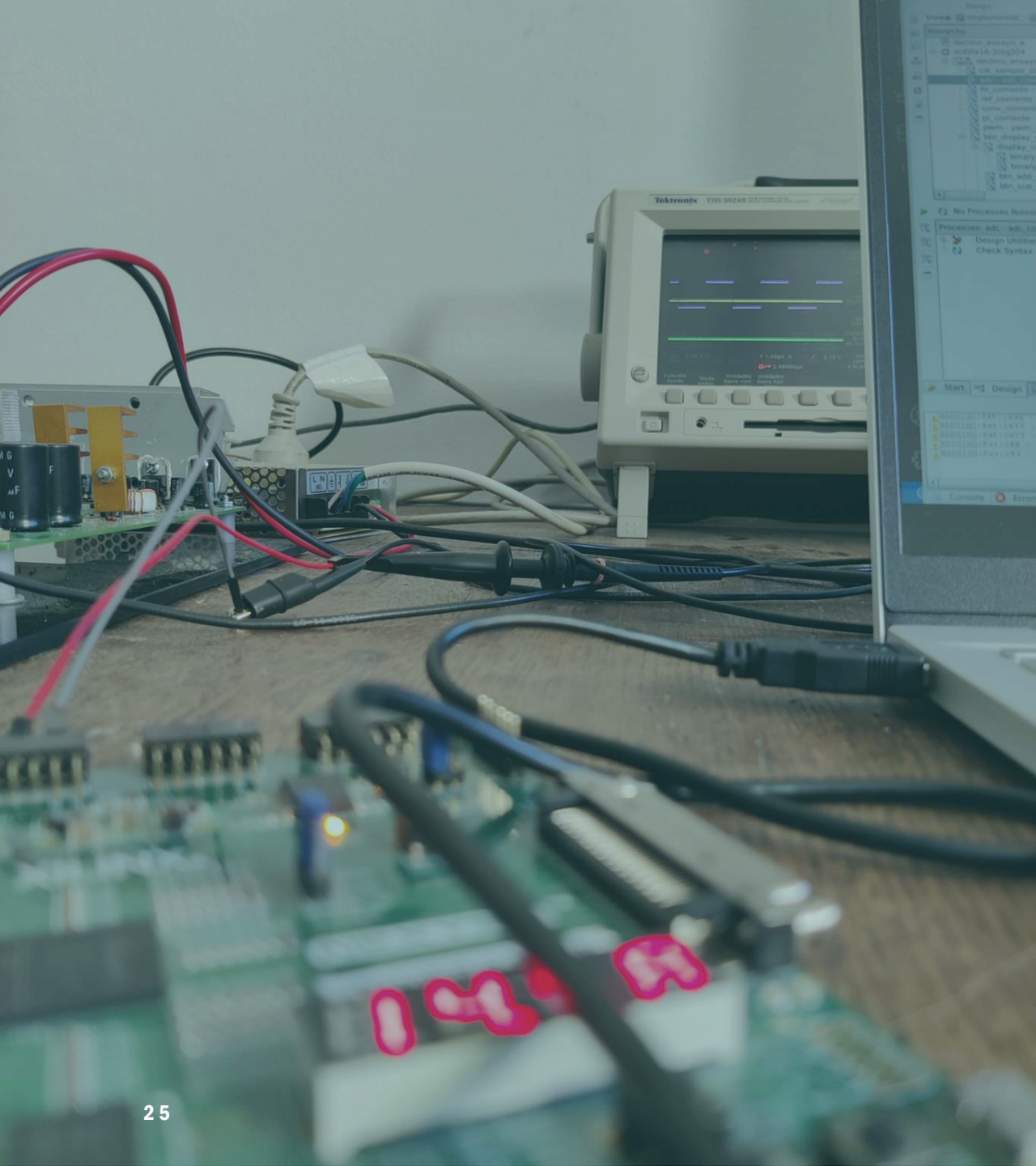
ESCALÓN POSITIVO DE RESISTENCIA DE 15Ω A 20Ω

Tensión en la salida



Corriente por el inductor filtrada





ÍNDICE

Motivación

Investigación

Diseño

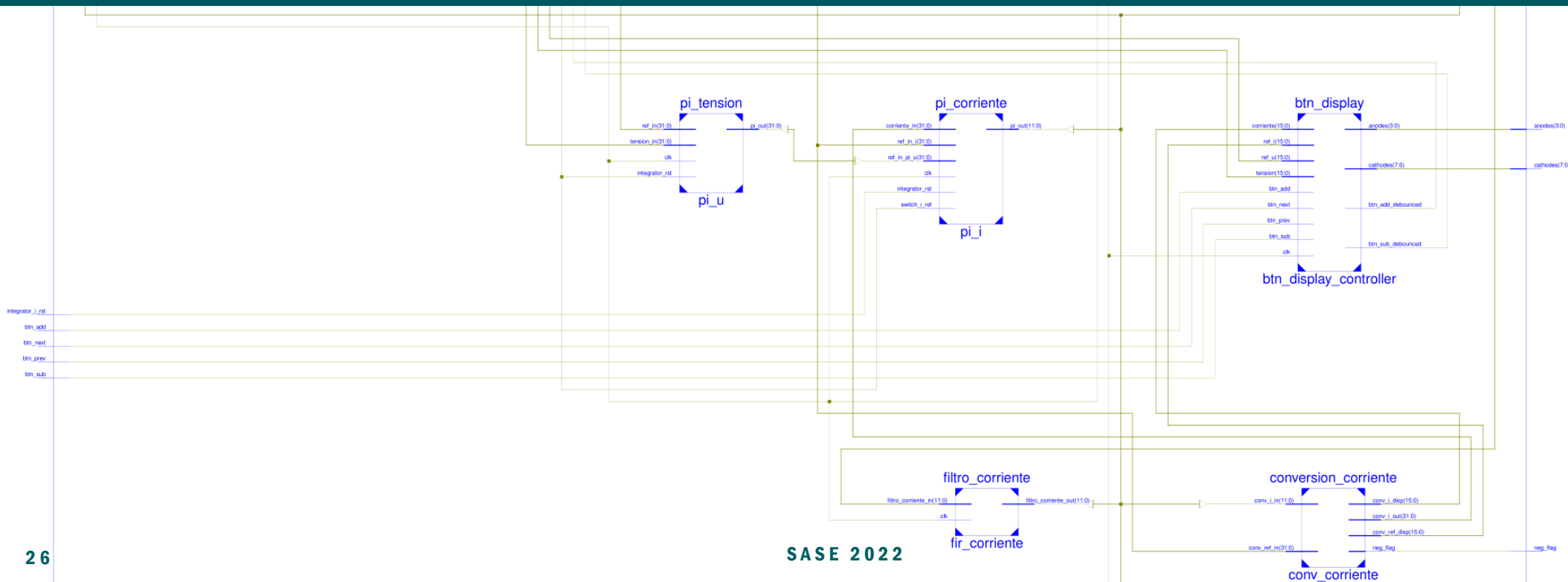
Implementación

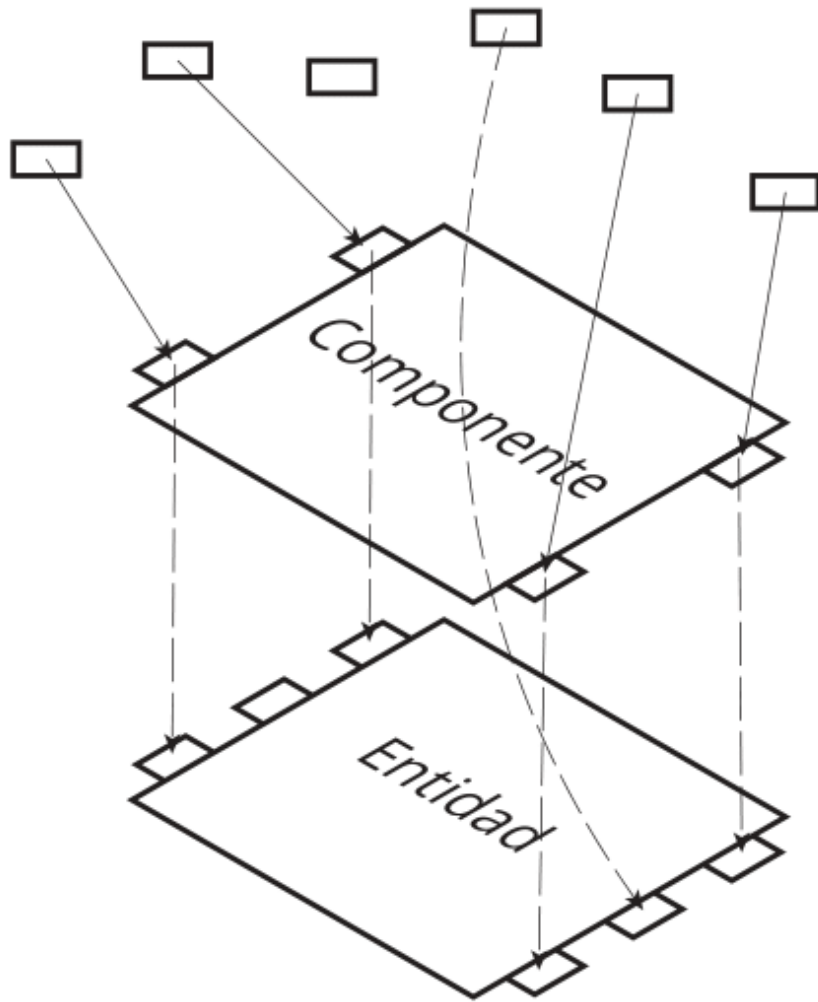
Ensayos

Conclusiones

IMPLEMENTACIÓN

La implementación fue realizada en la placa de desarrollo Nexys 3, la cual presenta una arquitectura FPGA. El algoritmo del sistema de control fue programado en VHDL.





IMPLEMENTACIÓN: VHDL

VHSIC Hardware Description Language

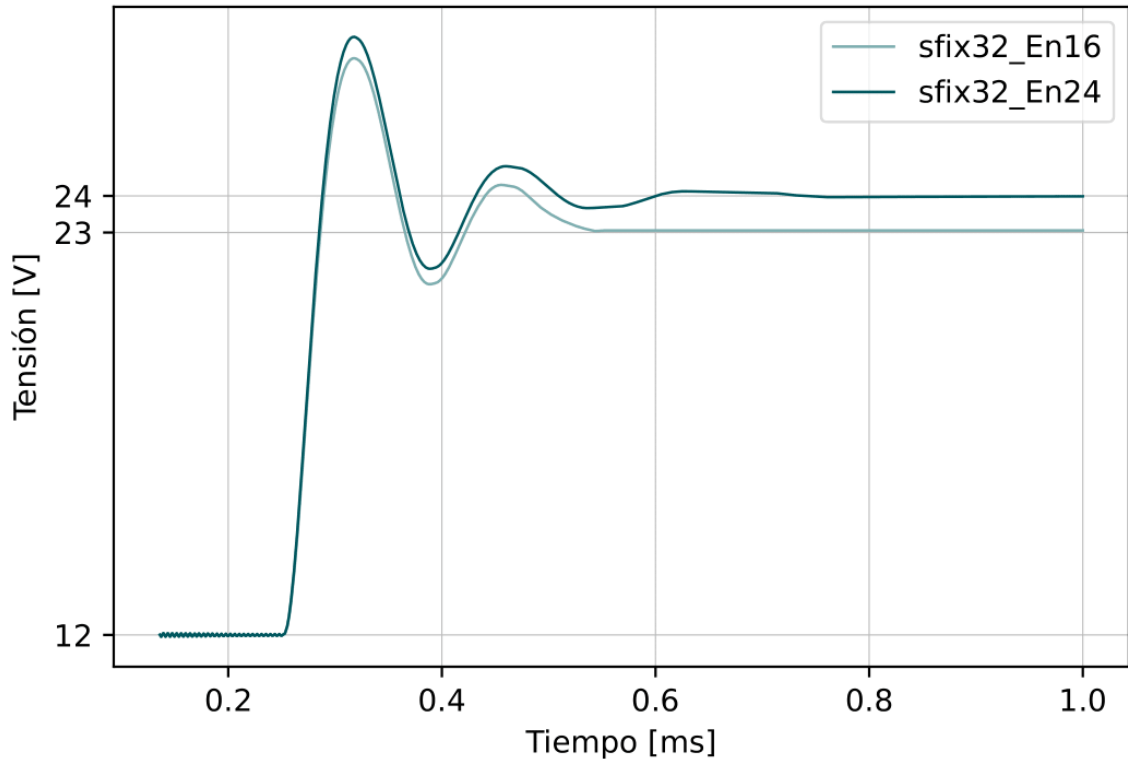
Very High-Speed Integrated Circuits Program

Modularización de cada componente del sistema de control

Flexibilidad debido a la posibilidad de instanciación de cada componente

Representación de dato con punto fijo

Tensión de carga con un control a 24V



IMPLEMENTACIÓN: REPRESENTACIÓN DE DATOS

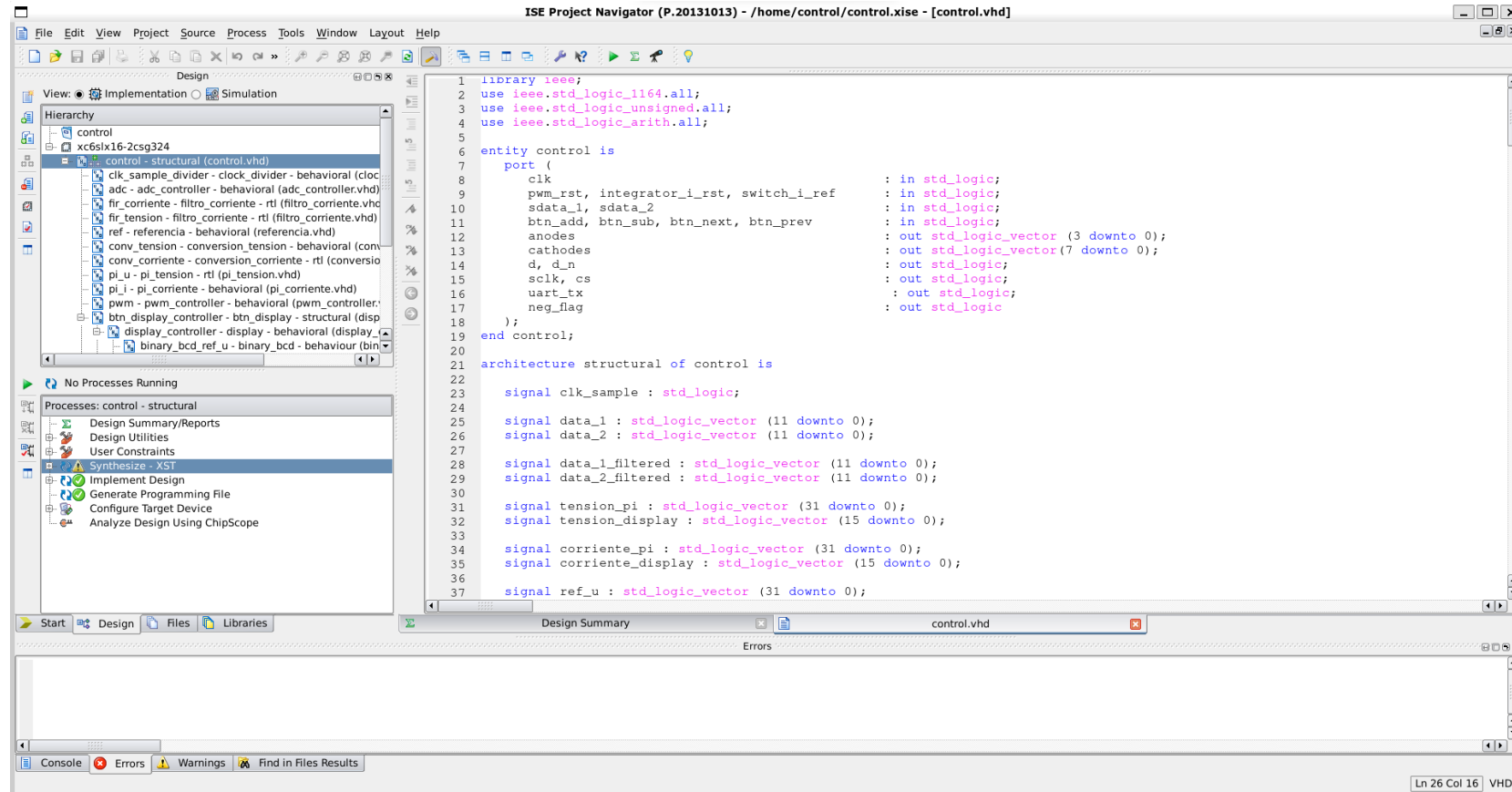
Es necesario encontrar un balance entre exactitud y cantidad de recursos utilizados en la representación digital de datos en el algoritmo

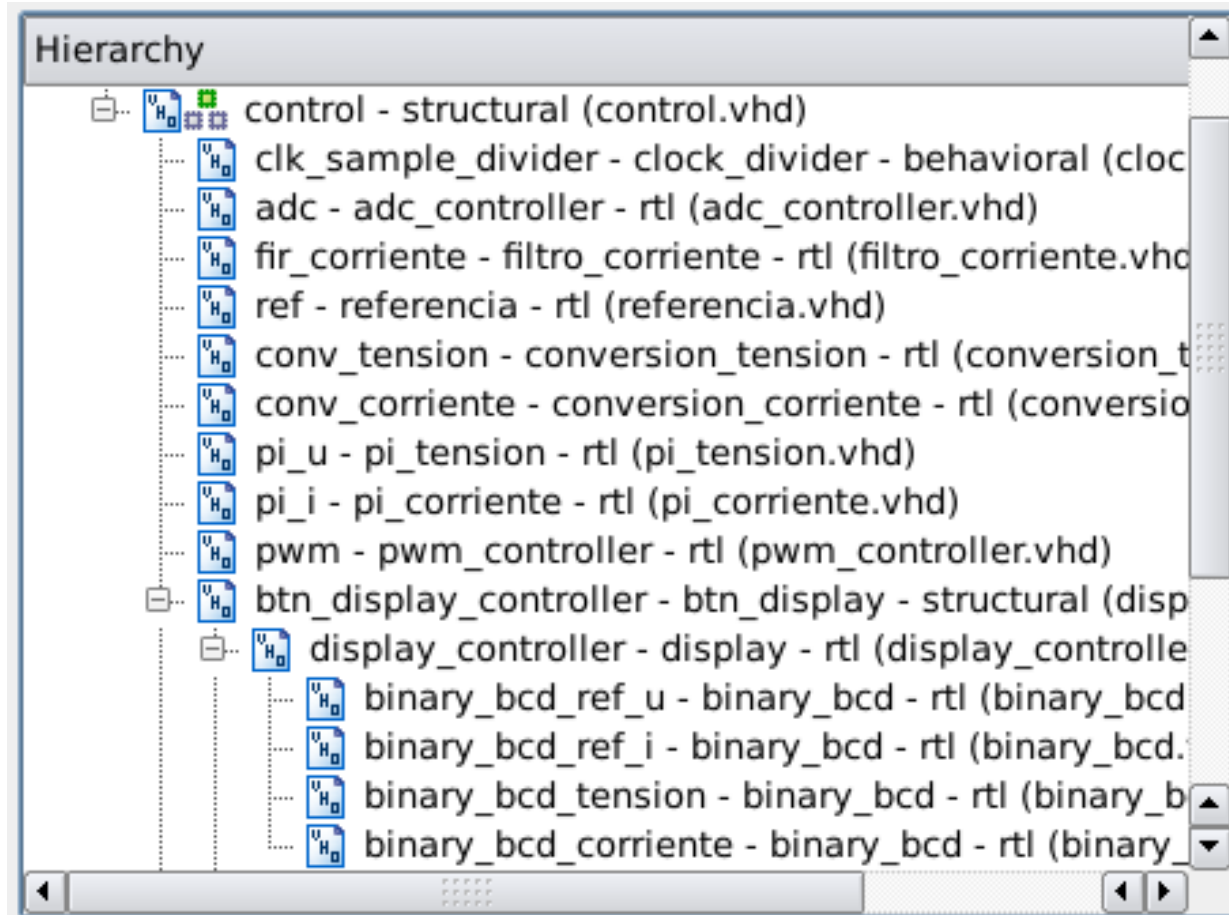
IMPLEMENTACIÓN: HERRAMIENTAS UTILIZADAS



Model*Sim*®

IMPLEMENTACIÓN: XILINX ISE





IMPLEMENTACIÓN: JERARQUÍA

RTL (*Register-Transfer-Level*): Archivos que describen las partes individuales de cada sistema y son “sintetizables”.

Estructural: Archivos que describen los sistemas de alto nivel.

IMPLEMENTACIÓN: ELEMENTOS RTL

```
architecture rtl of filtro_corriente is

    signal filtro_corriente_in_unsigned : unsigned(11 downto 0);
    signal k_0_mul_temp : unsigned(23 downto 0);
    signal k_0_out1 : signed(31 downto 0);
    signal delay_out1 : signed(31 downto 0) := to_signed(0, 32);
    signal k_1_mul_temp : signed(63 downto 0);
    signal k_1_out1 : signed(31 downto 0);
    signal add_out1 : signed(31 downto 0);
    signal conversion_out_out1 : unsigned(11 downto 0);

begin
    filtro_corriente_in_unsigned <= unsigned(filtro_corriente_in);

    k_0_mul_temp <= to_unsigned(16#F71#, 12) * filtro_corriente_in_unsigned;
    k_0_out1 <= signed(resize(k_0_mul_temp(23 downto 2), 32));

    k_1_mul_temp <= to_signed(2115099595, 32) * delay_out1;
    k_1_out1 <= k_1_mul_temp(62 downto 31);

    add_out1 <= k_0_out1 + k_1_out1;

    delay_process : process (clk)
    begin
        if rising_edge(clk) then
            delay_out1 <= add_out1;
        end if;
    end process delay_process;

    conversion_out_out1 <= unsigned(delay_out1(27 downto 16));
```

IMPLEMENTACIÓN: ELEMENTOS ESTRUCTURALES

```
component adc_controller is
  port (
    clk      : in std_logic;
    rst      : in std_logic;
    sdata_1  : in std_logic;
    sdata_2  : in std_logic;
    start    : in std_logic;

    sclk     : out std_logic;
    cs       : out std_logic;
    data_1   : out std_logic_vector(11 downto 0);
    data_2   : out std_logic_vector(11 downto 0);
    done     : out std_logic
  );
end component;

component pwm_controller is
  port (
    clk      : in std_logic;
    rst      : in std_logic;
    duty_cycle : in std_logic_vector(11 downto 0);
    pwm_out  : out std_logic;
    pwm_n_out : out std_logic
  );
end component;

component filtro_corriente is
  port (
    clk          : in std_logic;
    filtro_corriente_in : in std_logic_vector(11 downto 0);
    filtro_corriente_out : out std_logic_vector(11 downto 0)
  );
end component;

component referencia is
  port (
    clk      : in std_logic;
    btn_add  : in std_logic;
    btn_sub  : in std_logic;
  );
end component;
```

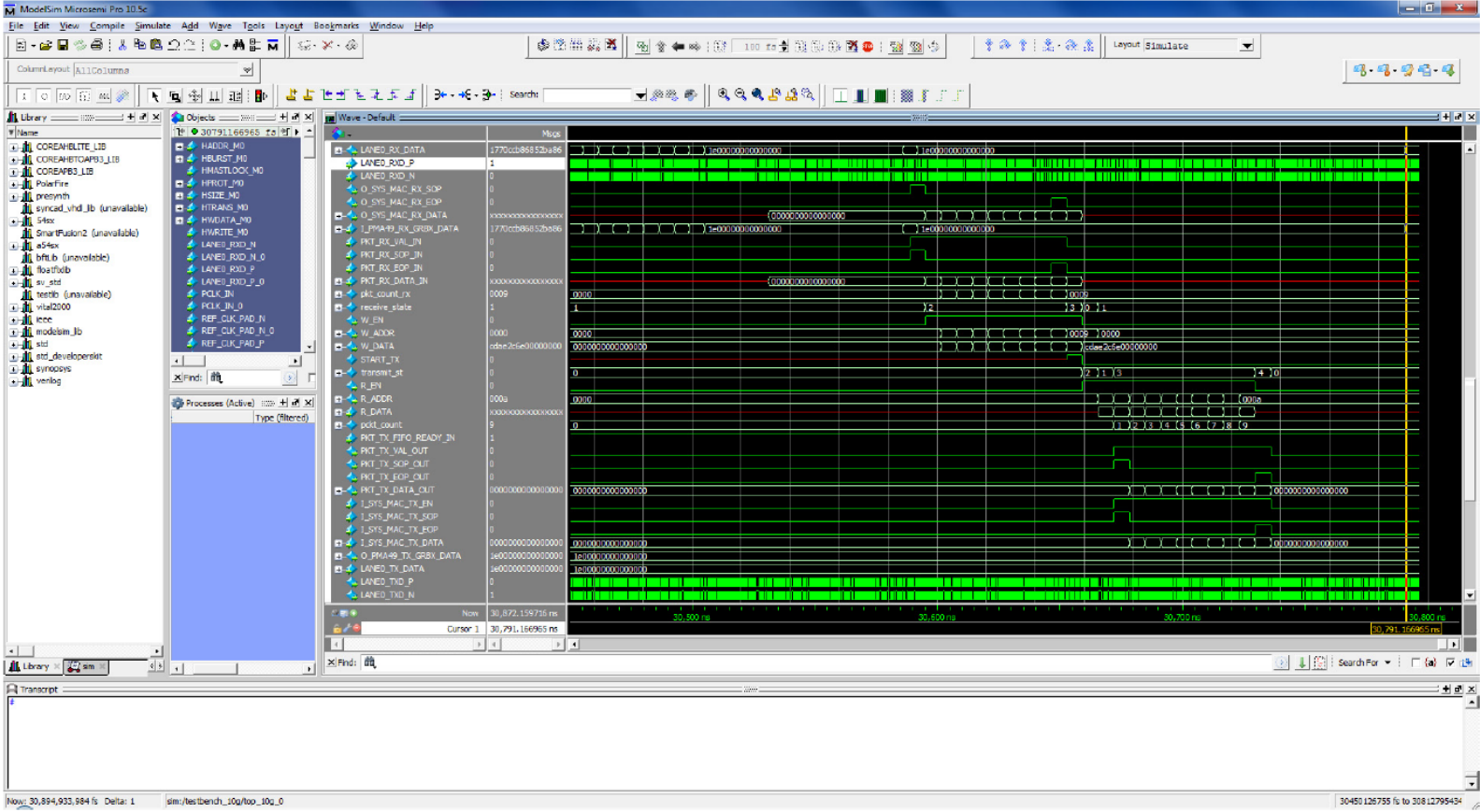
```
adc : adc_controller
  port map(
    clk      => clk,
    rst      => '0',
    sdata_1  => sdata_1,
    sdata_2  => sdata_2,
    sclk     => sclk,
    cs       => cs,
    data_1   => data_1, -- Tensión
    data_2   => data_2, -- Corriente
    start    => clk_sample,
    done     => open
  );

fir_corriente : filtro_corriente
  port map(
    clk          => clk_sample,
    filtro_corriente_in => data_2,
    filtro_corriente_out => data_2_filtered
  );

ref : referencia
  port map(
    clk      => clk,
    btn_add  => btn_add_debounced,
    ref_type => switch_i_ref,
    btn_sub  => btn_sub_debounced,
    ref_u_out => ref_u,
    ref_i_out => ref_i
  );

conv_tension : conversion_tension
  port map(
    conv_u_in      => data_1_filtered,
    conv_ref_in    => ref_u,
    conv_u_out     => tension_pi,
  );
```

IMPLEMENTACIÓN: MODELSIM



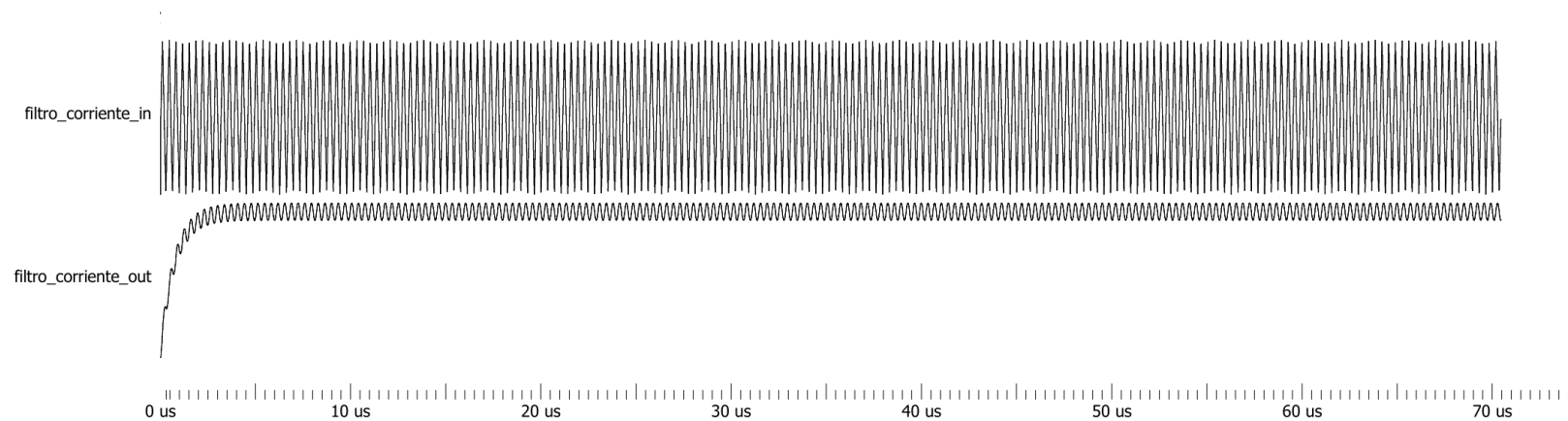
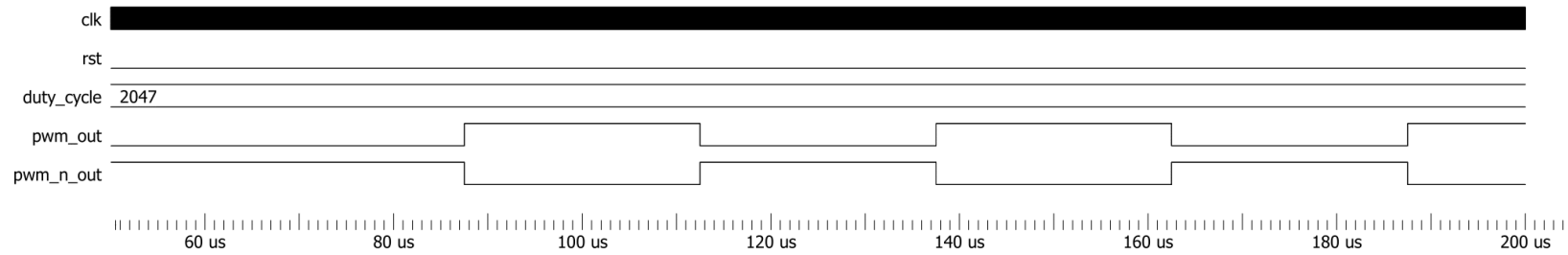

```
 uut: referencia PORT MAP (  
     clk => clk,  
     btn_add => btn_add,  
     btn_sub => btn_sub,  
     ref_out => ref_out  
 );  
  
 -- Stimulus process  
 stim_proc: process  
 begin  
     -- Hold reset state for 100 ns.  
     wait for 100 ns;  
  
     wait for clk_period*2;  
  
     btn_add <= '1';  
  
     wait for clk_period*2;  
  
     btn_add <= '0';  
  
     wait for clk_period*2;  
  
     btn_add <= '1';
```

IMPLEMENTACIÓN: TEST BENCH

Es necesario realizar una simulación de cada componente para verificar su correcto funcionamiento antes de ensamblar al sistema de alto nivel.

Nuevo tipo de componente: de comportamiento o *behavioral*.

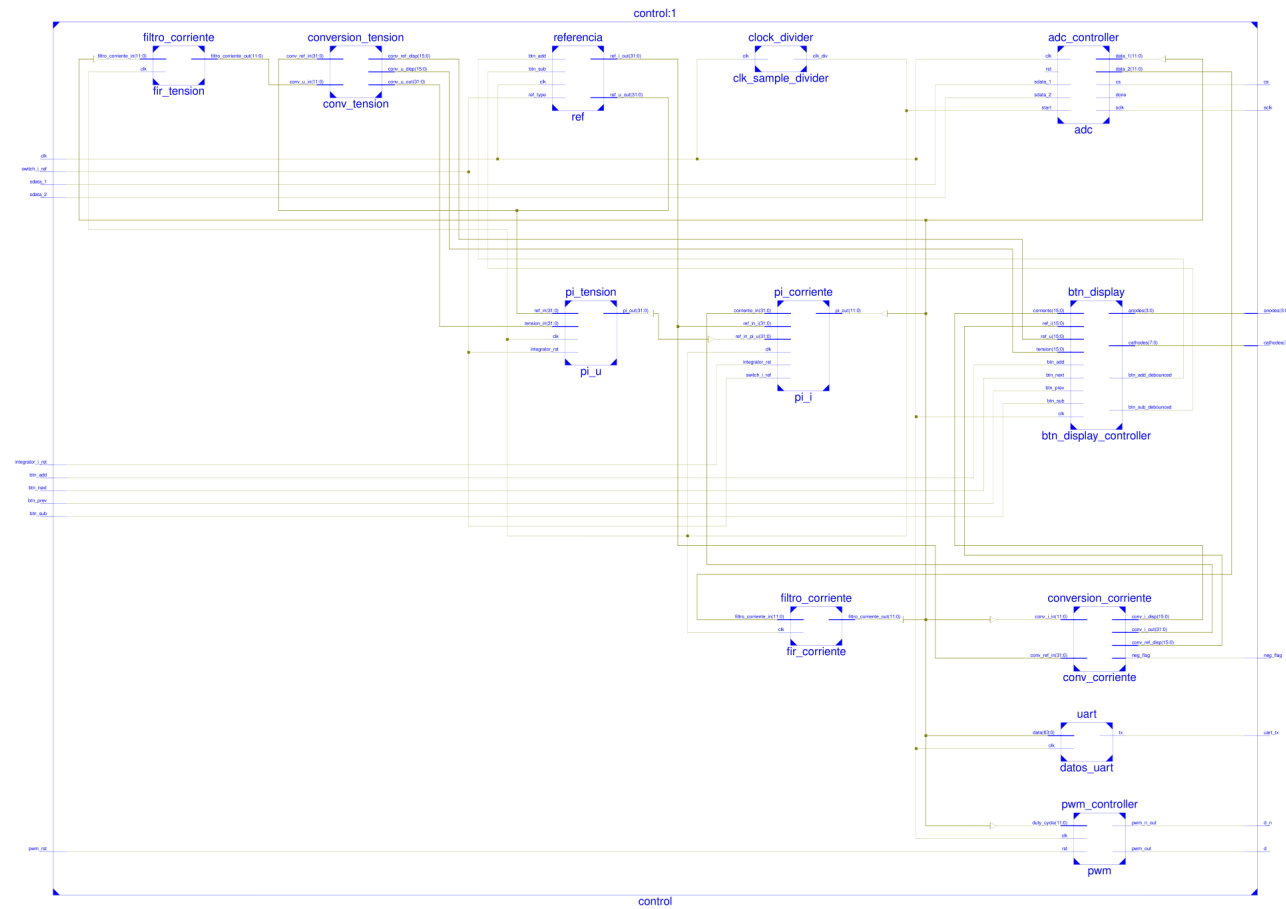
IMPLEMENTACIÓN: TEST BENCH

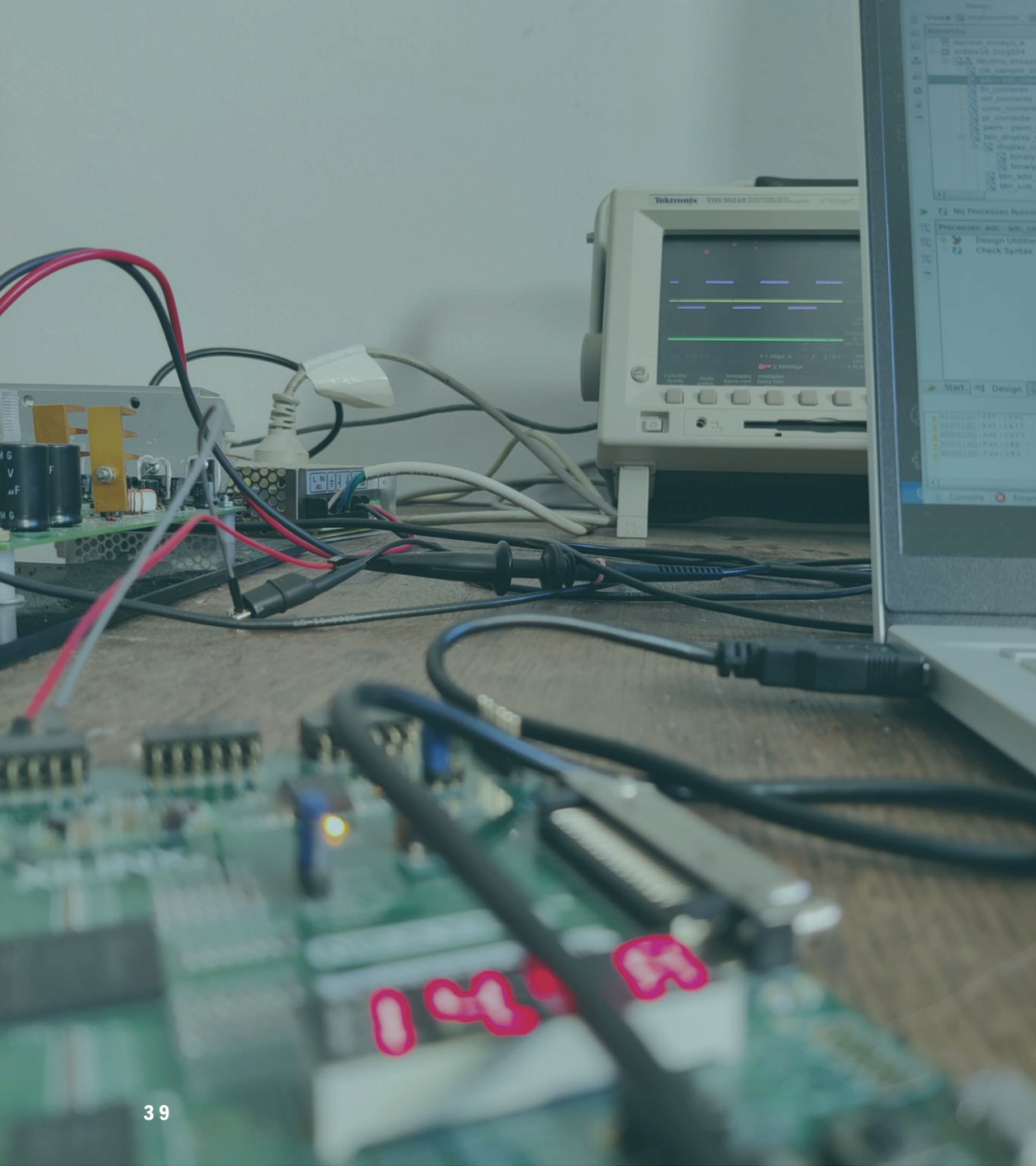


IMPLEMENTACIÓN: RESTRICCIONES DE DISEÑO

- Restricciones de síntesis** Reglas para las herramientas de síntesis que permiten mejoras de performance o tiempos de síntesis.
- Restricciones de I/O** Reglas de asignación de pines o ruteo de señales.
- Restricciones de tiempo** Reglas de performance de diseño que impactan influncian y guían la implementación de los elementos de diseño en el FPGA.
- Restricciones de área** Reglas de limitación en el área del FPGA en donde la herramienta de síntesis puede colocar los elementos de diseño.

IMPLEMENTACIÓN: ALGORITMO DE CONTROL





ÍNDICE

Motivación

Investigación

Diseño

Implementación

Ensayos

Conclusiones

ENSAYOS

OBJETIVOS

Comprobar el correcto funcionamiento del algoritmo de control junto a todos sus componentes

Ajuste de los parámetros de los controladores PID para una mejor respuesta (tuning)

Comprobar la dinámica del sistema eléctrico híbrido

ENSAYOS REALIZADOS

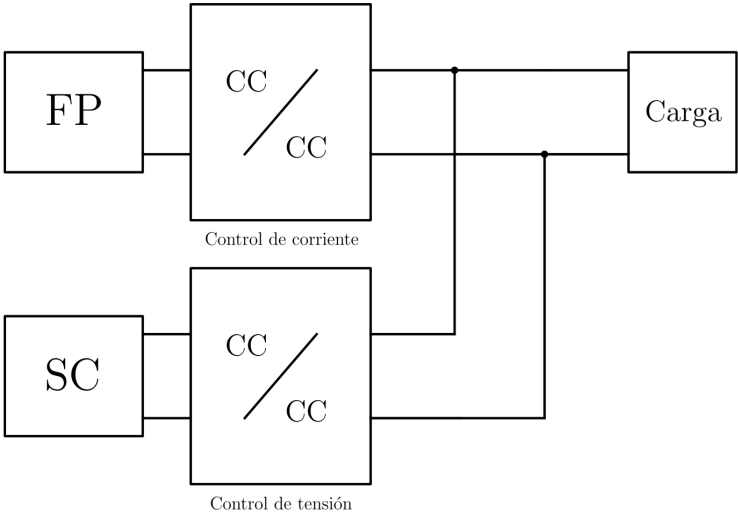
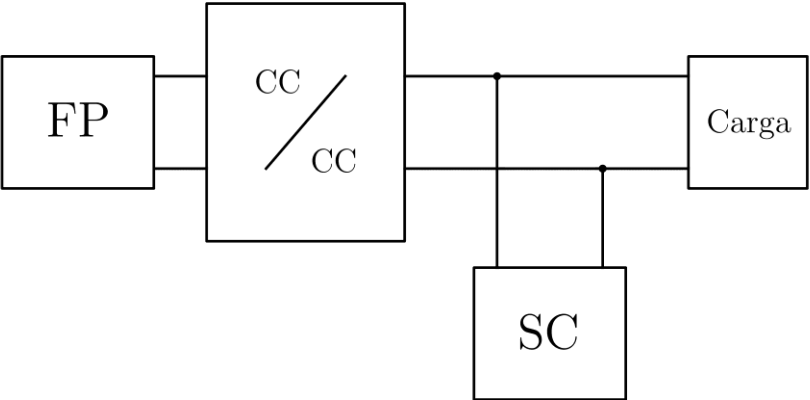
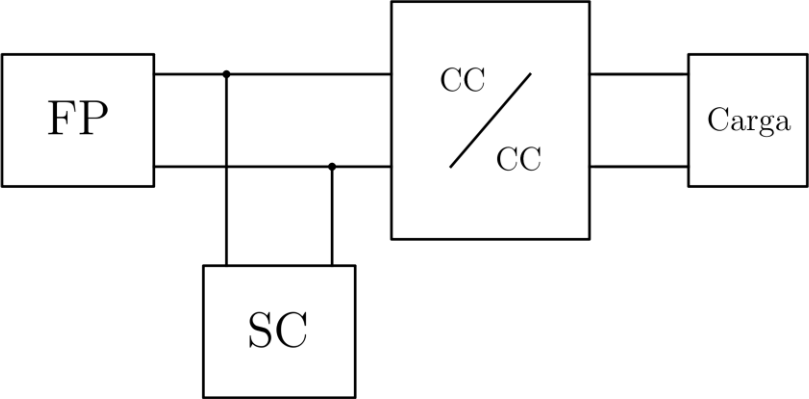
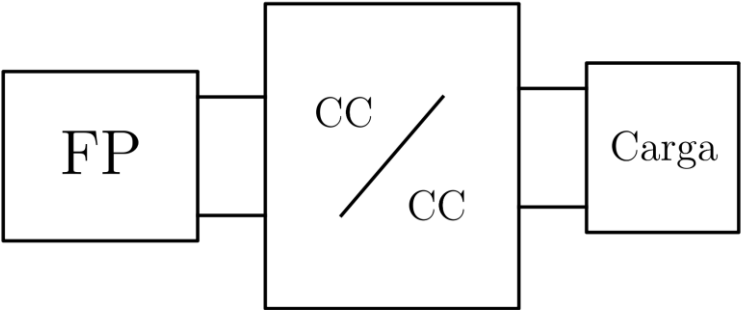
A lazo abierto

A lazo cerrado con control de corriente

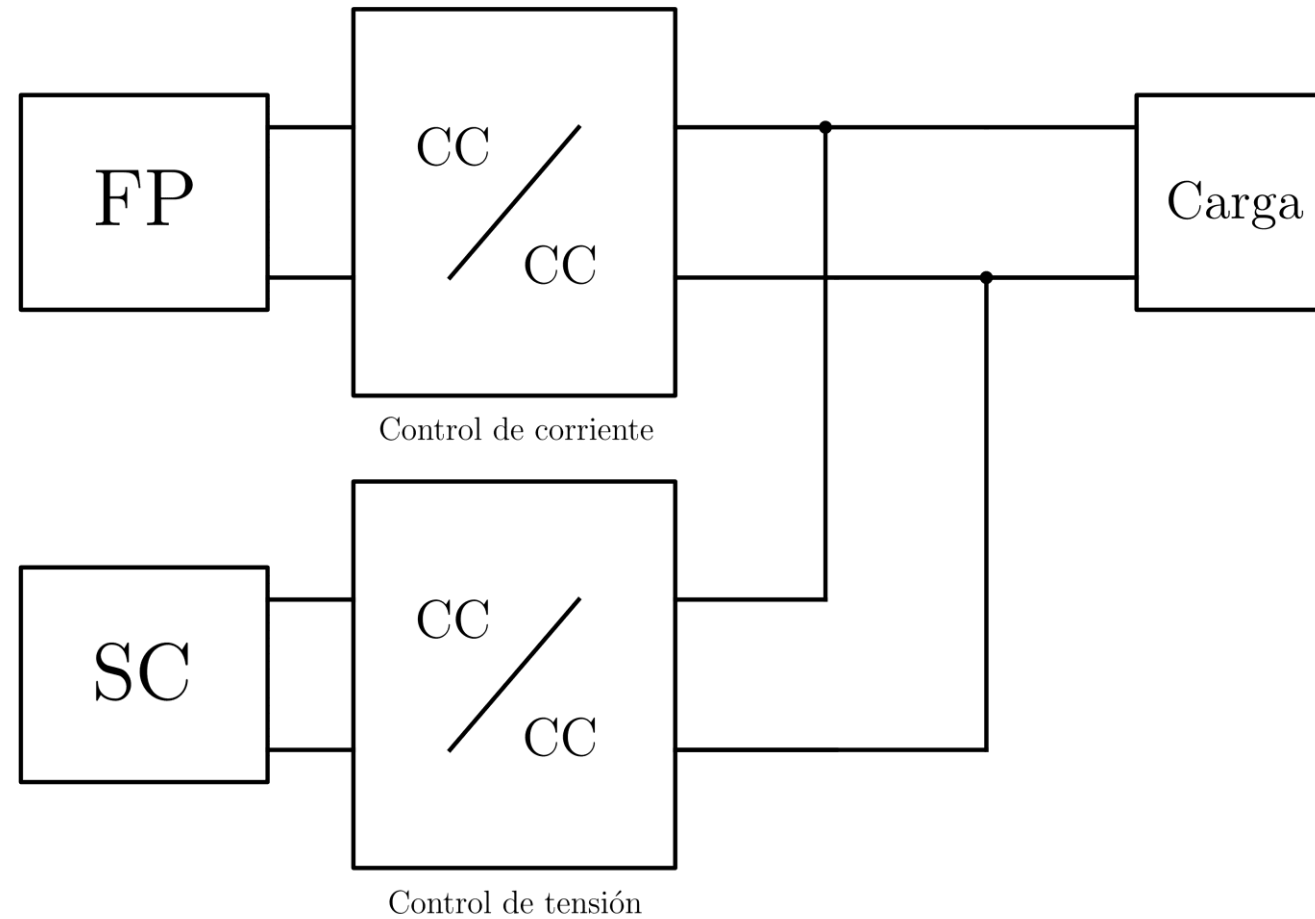
A lazo cerrado con control de corriente y tensión

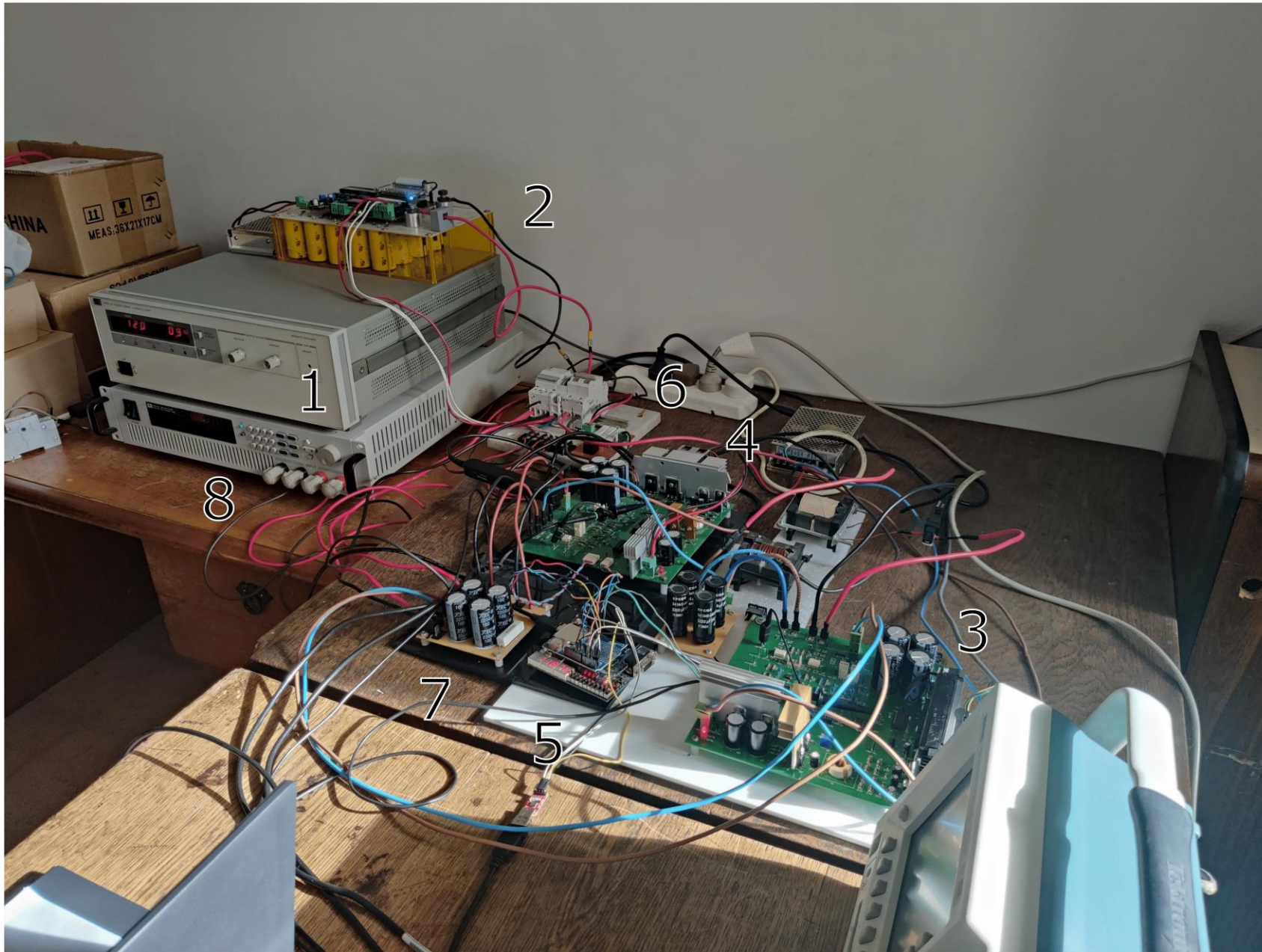
A lazo cerrado con banco de supercapacitores y fuente de potencia

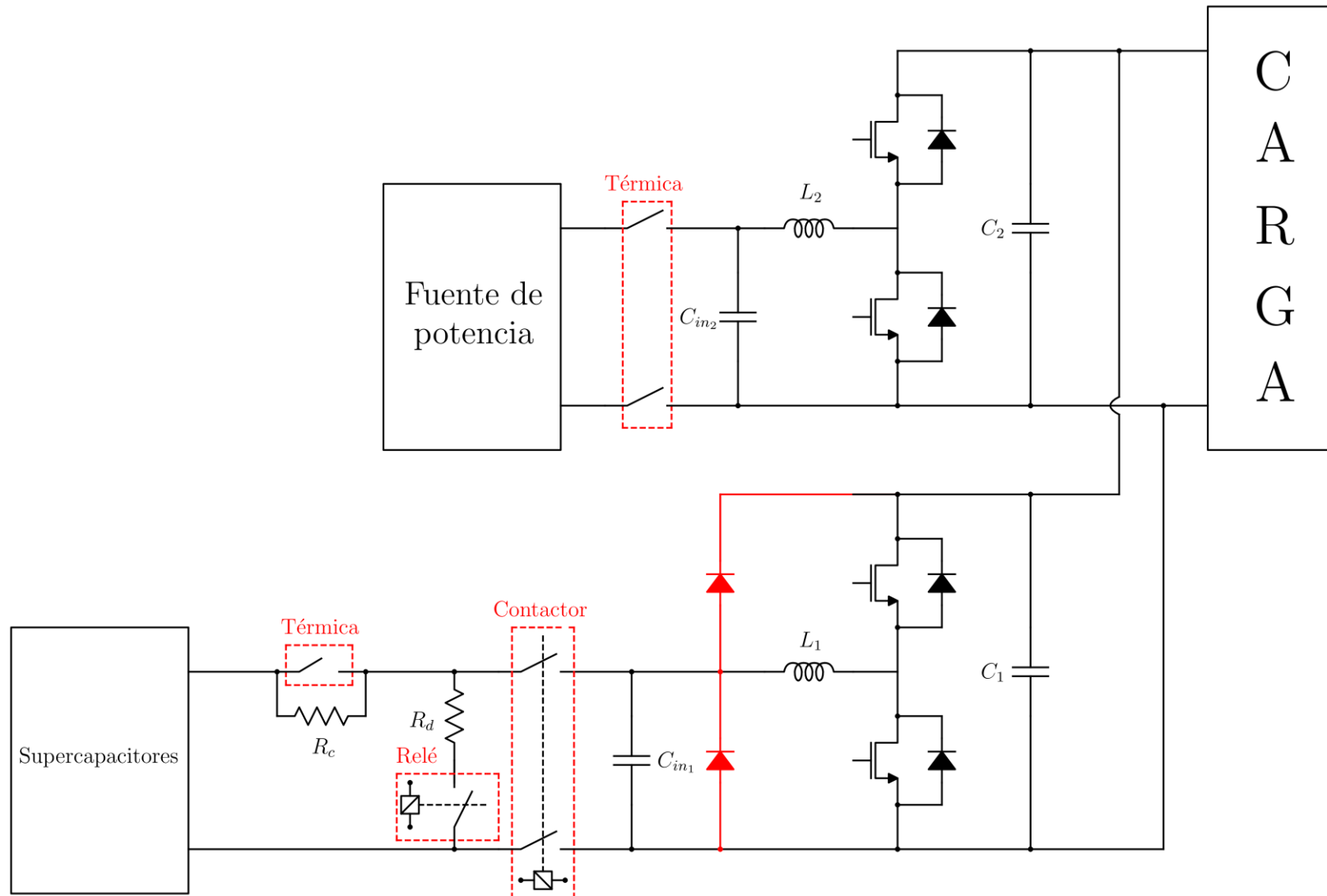
ENSAYOS: TOPOLOGÍAS ENSAYADAS



ENSAYOS: SISTEMA ELÉCTRICO HÍBRIDO



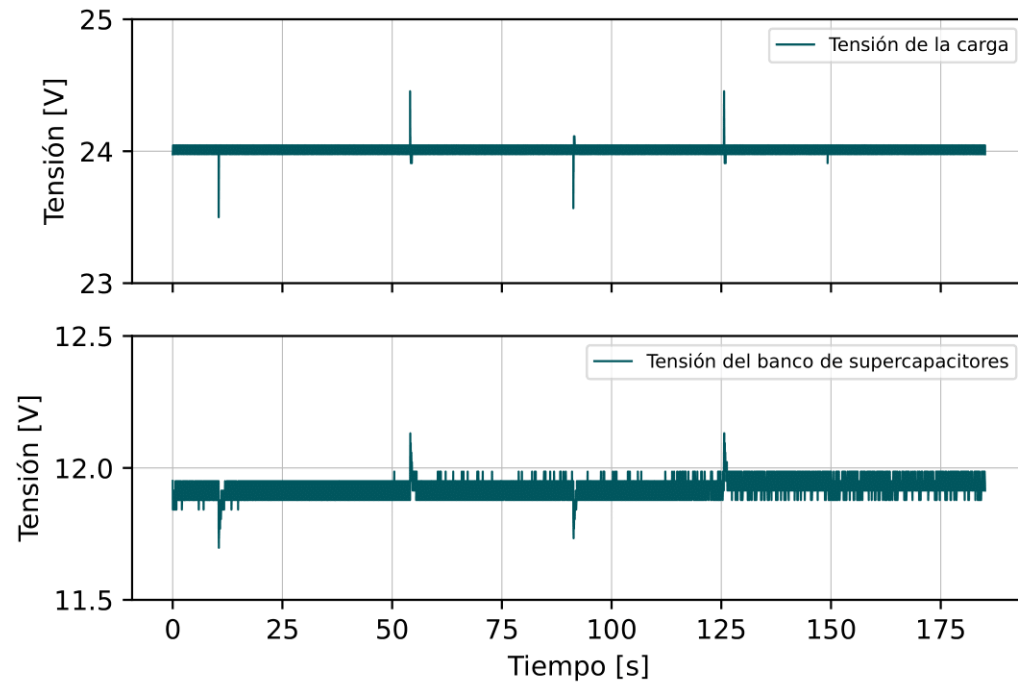




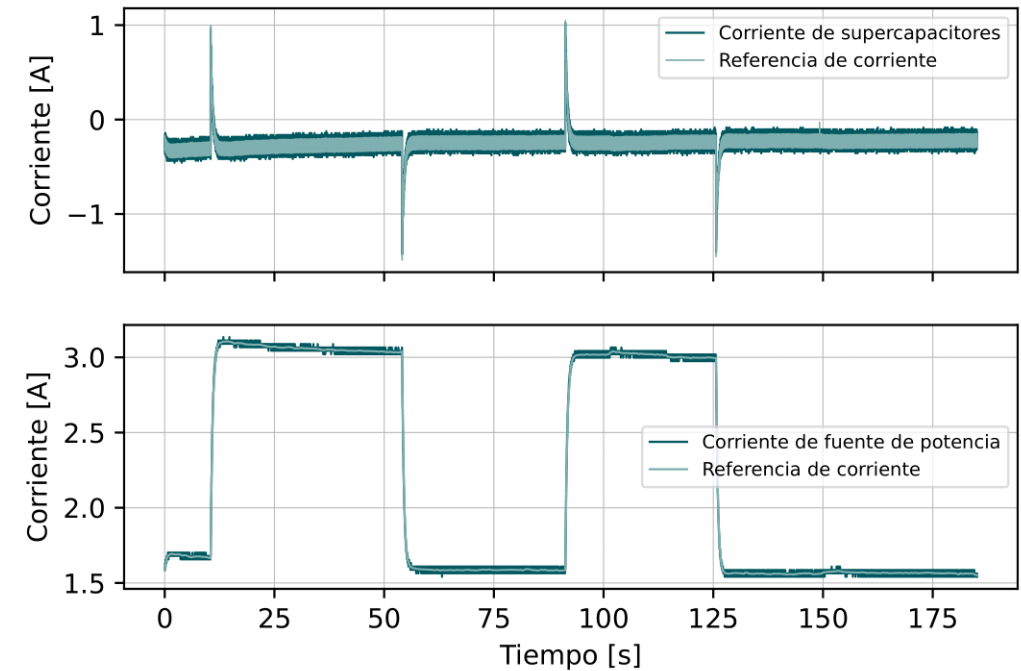
ENSAYOS: SISTEMA ELÉCTRICO HÍBRIDO

SALTOS POSITIVOS Y NEGATIVOS DE RESISTENCIA

Tensiones con saltos de resistencia



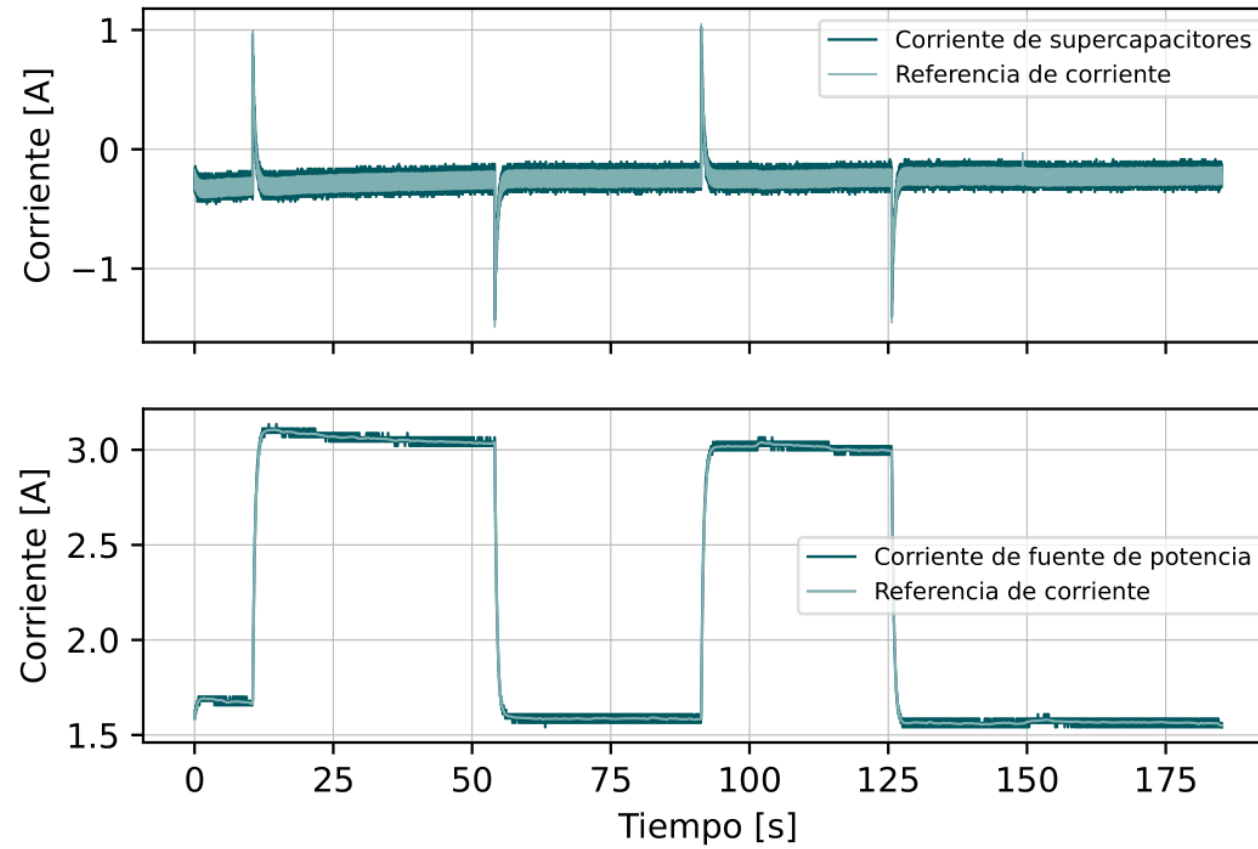
Corrientes con saltos de resistencia

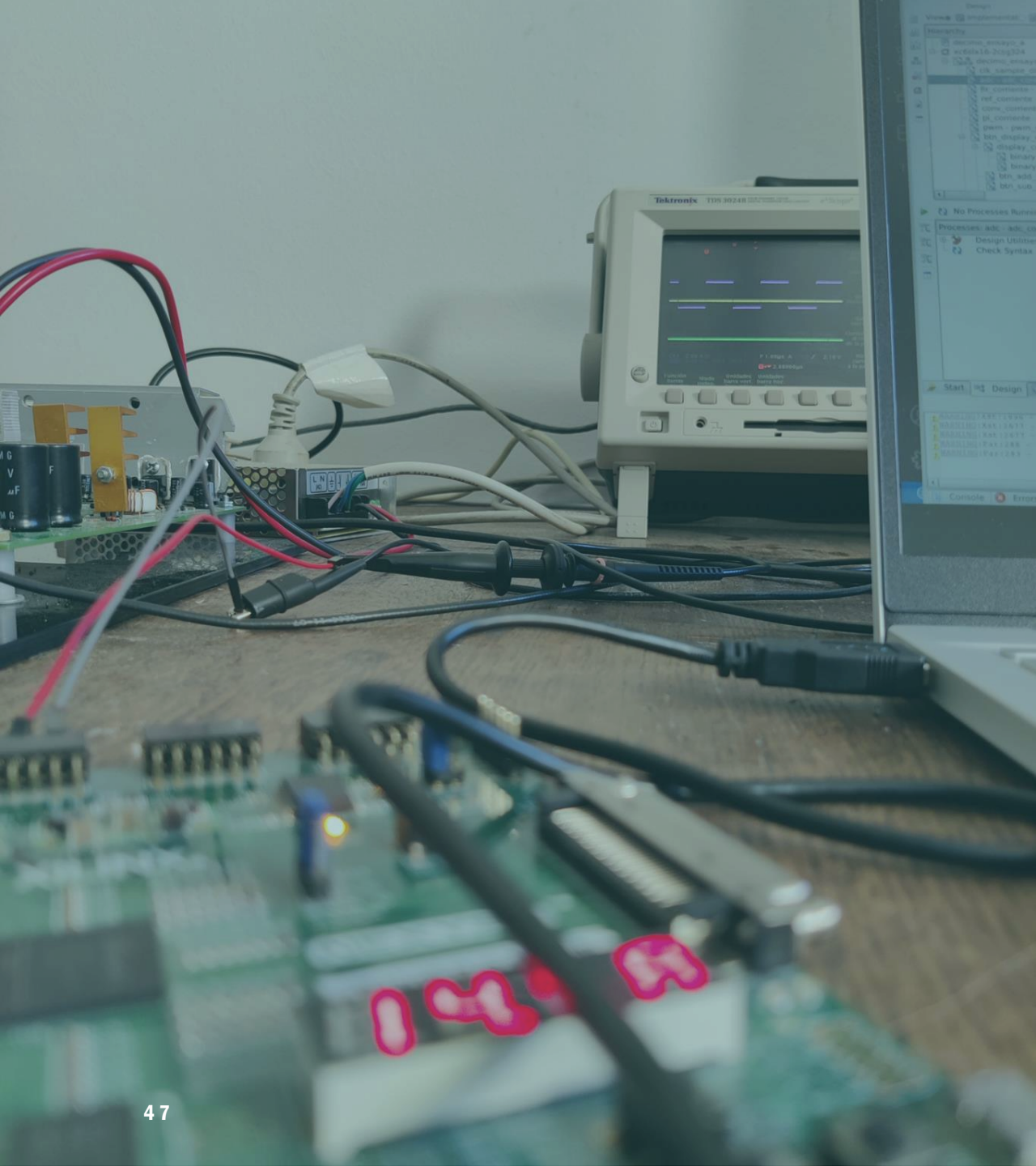


ENSAYOS: SISTEMA ELÉCTRICO HÍBRIDO

SALTOS POSITIVOS Y NEGATIVOS DE RESISTENCIA

Corrientes con saltos de resistencia





ÍNDICE

Motivación

Investigación

Diseño

Implementación

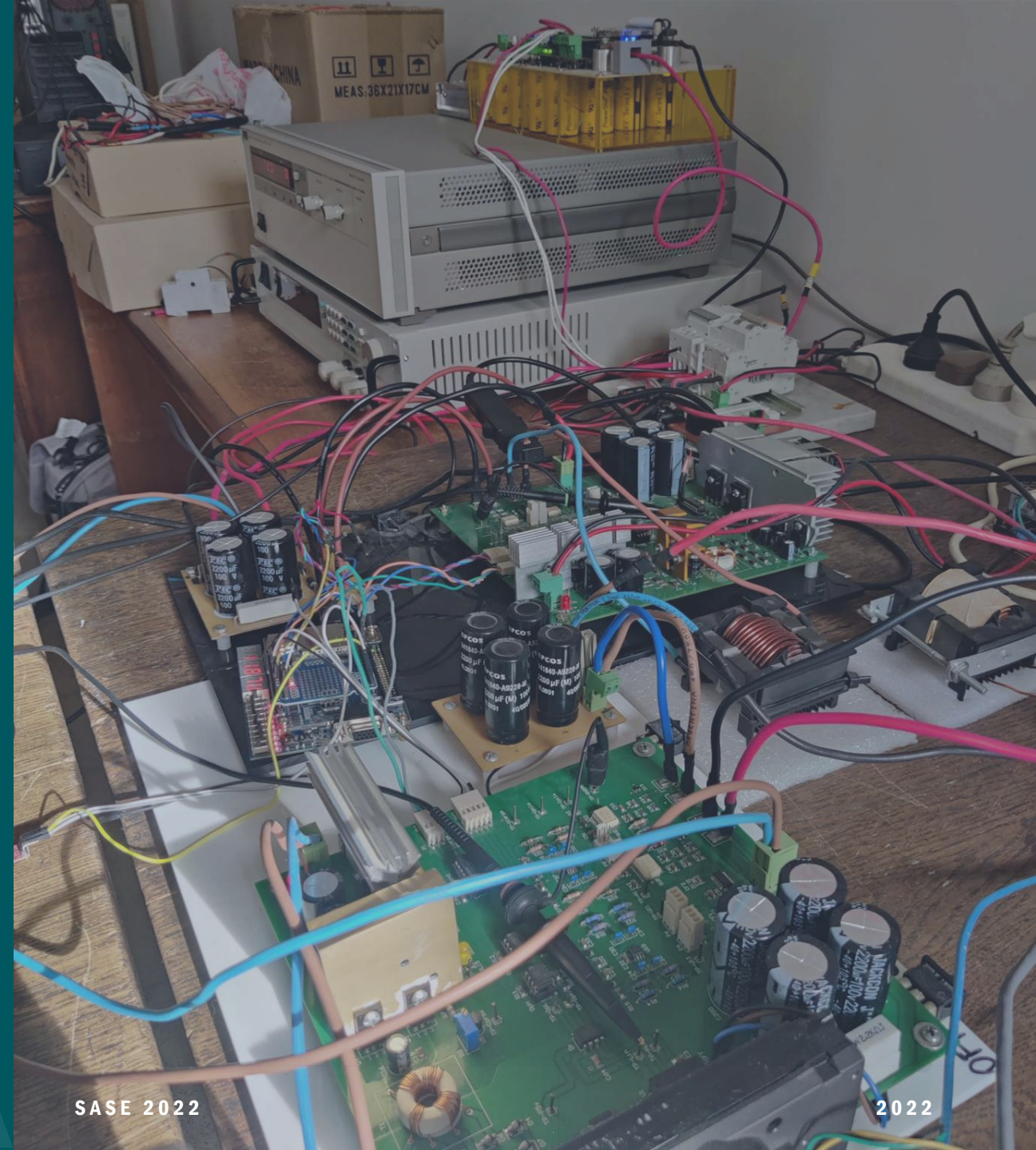
Ensayos

Conclusiones

CONCLUSIONES

Los ensayos realizados corroboraron el correcto funcionamiento del sistema de control

La dinámica del sistema eléctrico híbrido cumple con la premisa planteada en la etapa de investigación



CONCLUSIONES

OBJETIVOS CUMPLIDOS

Construcción de una plataforma híbrida con control de flujo de energía

Implementación de algoritmo de control en arquitectura FPGA mediante VHDL

TRABAJO A FUTURO

Aplicar otras estrategias de control y observar su performance

Añadir módulos de almacenamiento alternativo al sistema eléctrico híbrido

¡GRACIAS!

CAMPUS
UNIVERSITARIO



FACULTAD DE INGENIERÍA
UNIVERSIDAD NACIONAL DE LA PLATA

UNLP

<http://www.ing.unlp.edu.ar> | 0221 425-8911