



Facultad de Ingeniería  
UNLP

# CIRCUITOS DIGITALES Y MICROCONTROLADORES 2025

Presentación

Prof. Ing. José Juárez

# Circuitos Digitales y Microcontroladores

- Aula virtual en Moodle:

<https://asignaturas.unlp.edu.ar>

- Correo:

[cdymcomp@ing.unlp.edu.ar](mailto:cdymcomp@ing.unlp.edu.ar)

- Integrantes:

- **Profesor Adjunto:** Ing. José Juárez
- **Profesor Adjunto:** Dr. Juan Ignacio Fernández Michelli
- **Ayudante Diplomado:** Ing. Walter Aróztegui
- **Ayudante Diplomado:** Ing. Gabriel Vega Leañez
- **Ayudante Diplomado:** Ing. Lucas Martire
- **Ayudante Diplomado:** Ing. Eduardo Alippi
- **Ayudantes Alumnos:** Ulises Pereira, Joaquín Chanquía

# Reglamento

- 2. CLASES:

**Teoría:**

- Miércoles de 14 a 17hs (aula E26 )

**Prácticas :**

- Turno Lunes de 14 a 17hs (Lab. Barcala)
- Turno Jueves de 14 a 17hs (Lab. Barcala)

# Reglamento

- 3. TRABAJOS PRÁCTICOS

- 1-La cátedra propondrá 4 Guías de Trabajos Prácticos, con cuestionarios y ejercicios a resolver.
- 2-Cada TP posee **un ejercicio entregable** grupal que deberán ser resueltos mediante computadora, las herramientas de simulación y/o kit.
- 3-Los alumnos deberán mostrar a los docentes la solución funcionando y entregar mediante moodle el código y la documentación que se solicite.
- 4-Los ejercicios entregables tienen fecha límite de entrega y cada uno de los trabajos prácticos se calificará con nota del 1 al 10.
- 5-El trabajo entregable deberá realizarse en **Grupos de 2 alumnos preferentemente**.

# Reglamento

- **4. APROBACIÓN DEL CURSO (por Promoción)**

1. Se deberá aprobar **una evaluación teórico-práctica** que se tomará finalizando el curso con una calificación mayor o igual a 6 (seis). Esta evaluación tendrá dos instancias de recuperación.
2. Se deberán **haber aprobado los 4 trabajos prácticos** correspondientes con calificación promedio mayor o igual que 6 (seis).

# Reglamento

- 5. CALIFICACIÓN

3. Se determinará una calificación de trabajos prácticos que estará dada por el promedio simple de las calificaciones obtenidas en cada una de las guías de trabajos práctico.

4. La calificación final se obtendrá como  $0,6^* \text{ evaluación teórico-practica} + 0,4^* \text{ Trabajos Prácticos}$ , esta nota deberá ser mayor o igual que 6 (seis) para promocionar.

# Fechas Importantes

- **Miércoles 2 de julio:** Evaluación Teórico-práctica
- **Miércoles 16 de julio:** Primera Recuperación
- **Miércoles 6 de agosto:** Segunda Recuperación

# Objetivos y Contenidos

## Objetivos

- Iniciar al alumno en el estudio, diseño e implementación de **Sistemas Embebidos** con microcontroladores, orientados a la solución de problemas en tiempo real, considerando en particular su interconexión con periféricos.

## Contenidos Mínimos

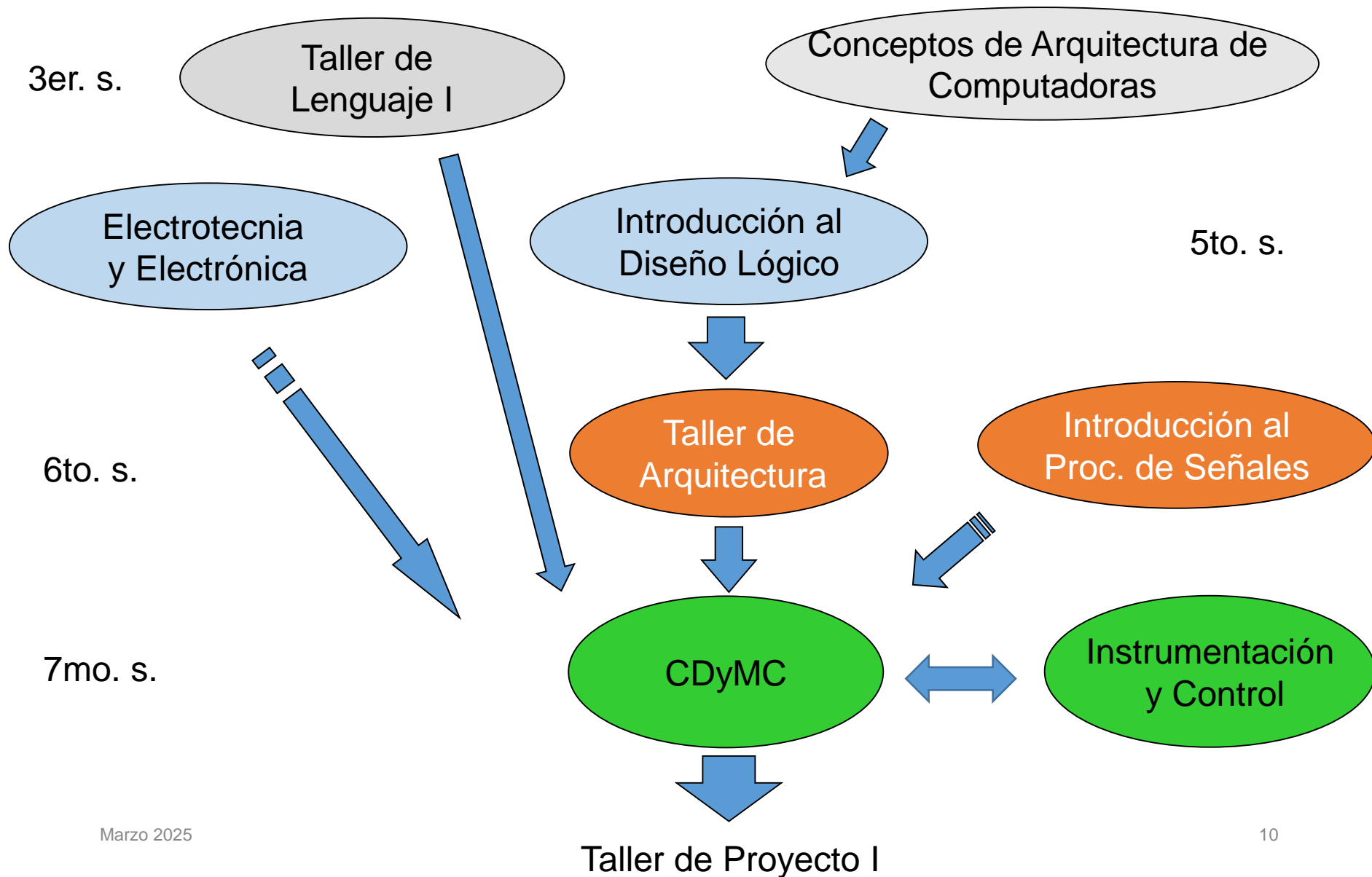
- Caracterización y componentes de un sistema embebido.
- Arquitectura de microcontroladores.
- Programación de la CPU y de los sistemas periféricos.
- Interfaces de comunicación
- Análisis y diseño de sistemas de tiempo real.



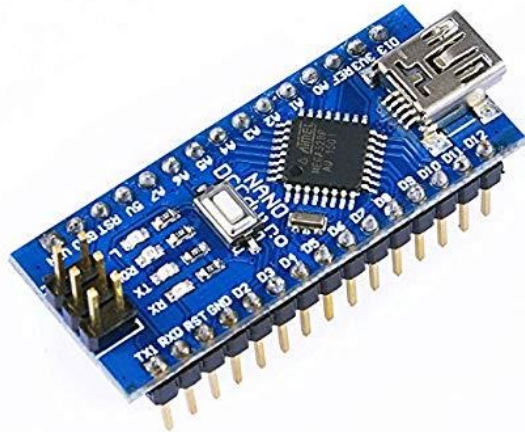
# Programa resumido

- **Caracterización de un Sistema Embebido**
  - Componentes de Hardware:
    - Microcontroladores 8bits/32Bits, Periféricos.
    - La familia de **Microcontroladores AVR**.
    - Puertos E/S, Circuitos de reloj, Temporizadores. Generadores PWM. Conversores A/D y D/A.
    - Interfaces de comunicación: UART, I2C, SPI.
  - Componentes de Software:
    - Programación en **Lenguaje C** para sistemas Embebidos.
    - Modularización, Planificadores, manejador de dispositivos.
    - Planificación de tareas de Tiempo real, tareas cooperativas, Interrupciones.
    - Modelización con Máquinas de Estado Finitos (MEF) y generalización.
    - Sistemas Operativos en tiempo real para Sistemas embebidos (RTOS).

# Integración de Conocimientos (Plan 2011)



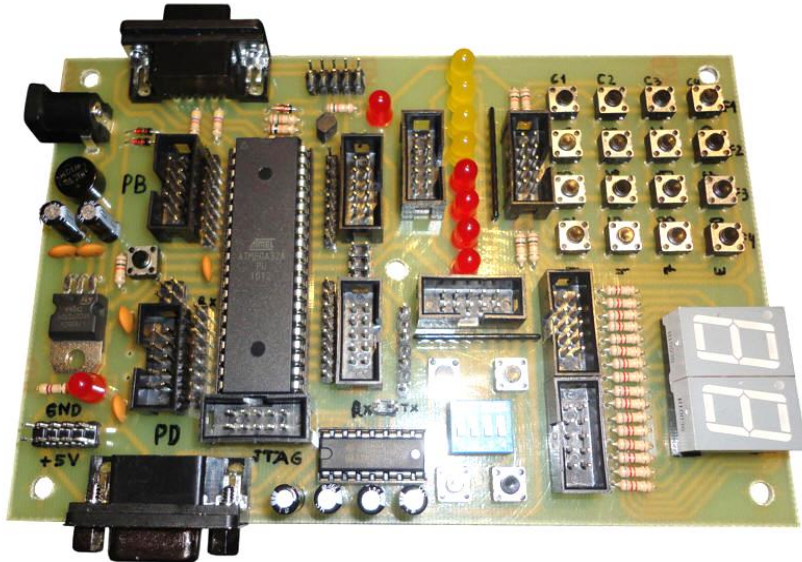
# ¿Qué herramientas vamos a usar en la materia?



# ?

# Placas de desarrollo MCU AVR

- Mi KIT ATMEGA32 (2008)

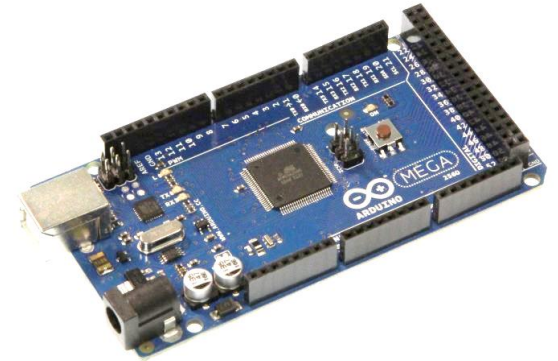


¿Qué es ARDUINO?  
Recomendado ver:

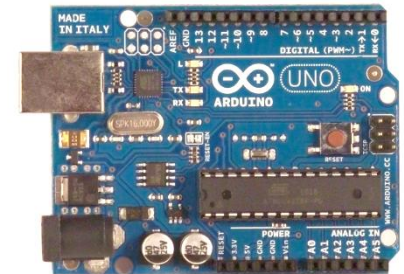


<https://www.arduino.cc/en/Guide/Introduction>

ARDUINO MEGA



ARDUINO UNO



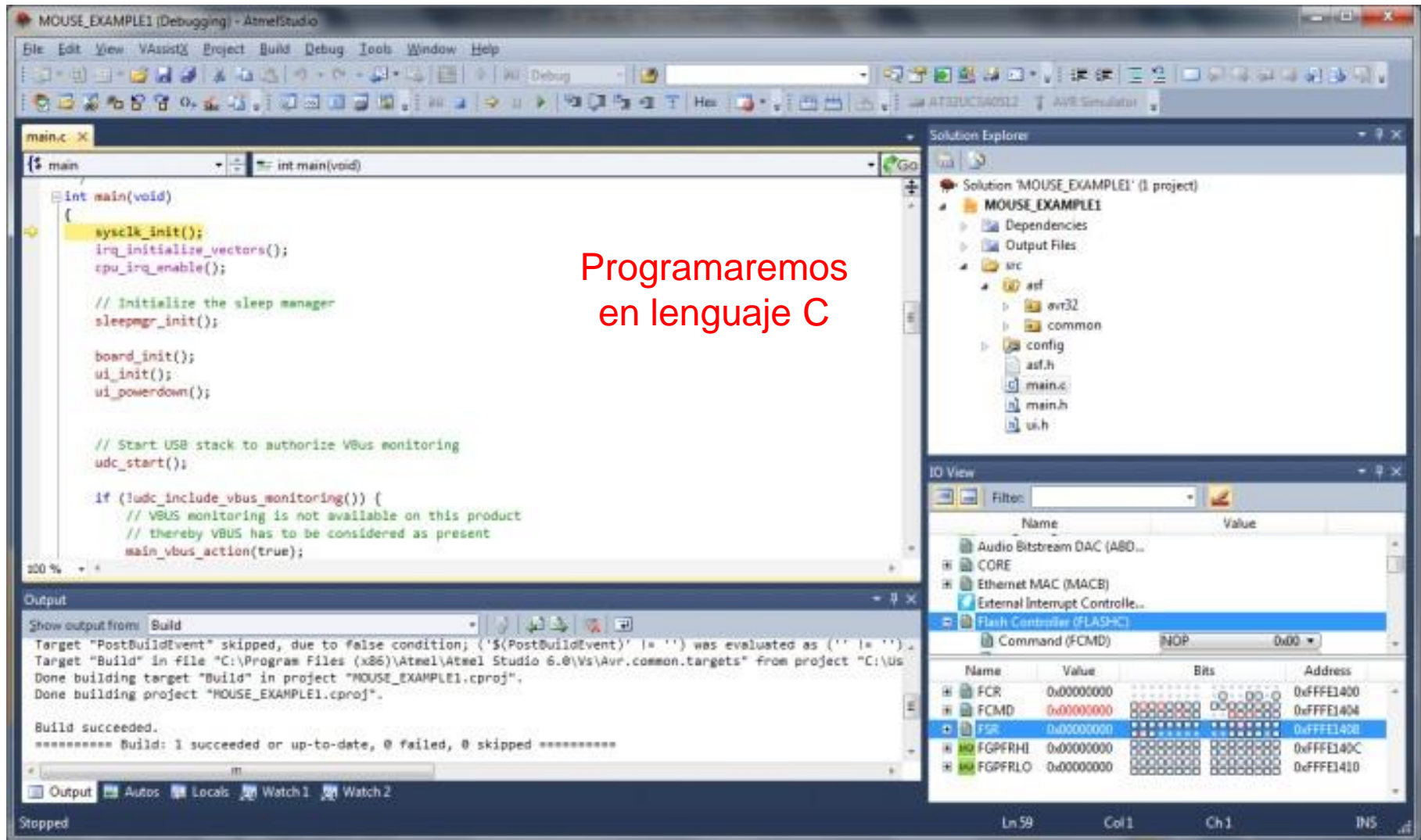
ARDUINO NANO



# Herramientas de desarrollo para AVR

- Entorno y compilador Microchip Studio

Programaremos en lenguaje C



The screenshot displays the Atmel Studio IDE interface for a project named 'MOUSE\_EXAMPLE1'. The main window shows the C source code for 'main.c', which includes initialization functions for the system clock, sleep manager, board, UI, and USB stack. The code is as follows:

```
int main(void)
{
    sysclk_init();
    irq_initialize_vectors();
    cpu_irq_enable();

    // Initialize the sleep manager
    sleepmgr_init();

    board_init();
    ui_init();
    ui_powerdown();

    // Start USB stack to authorize VBus monitoring
    udc_start();

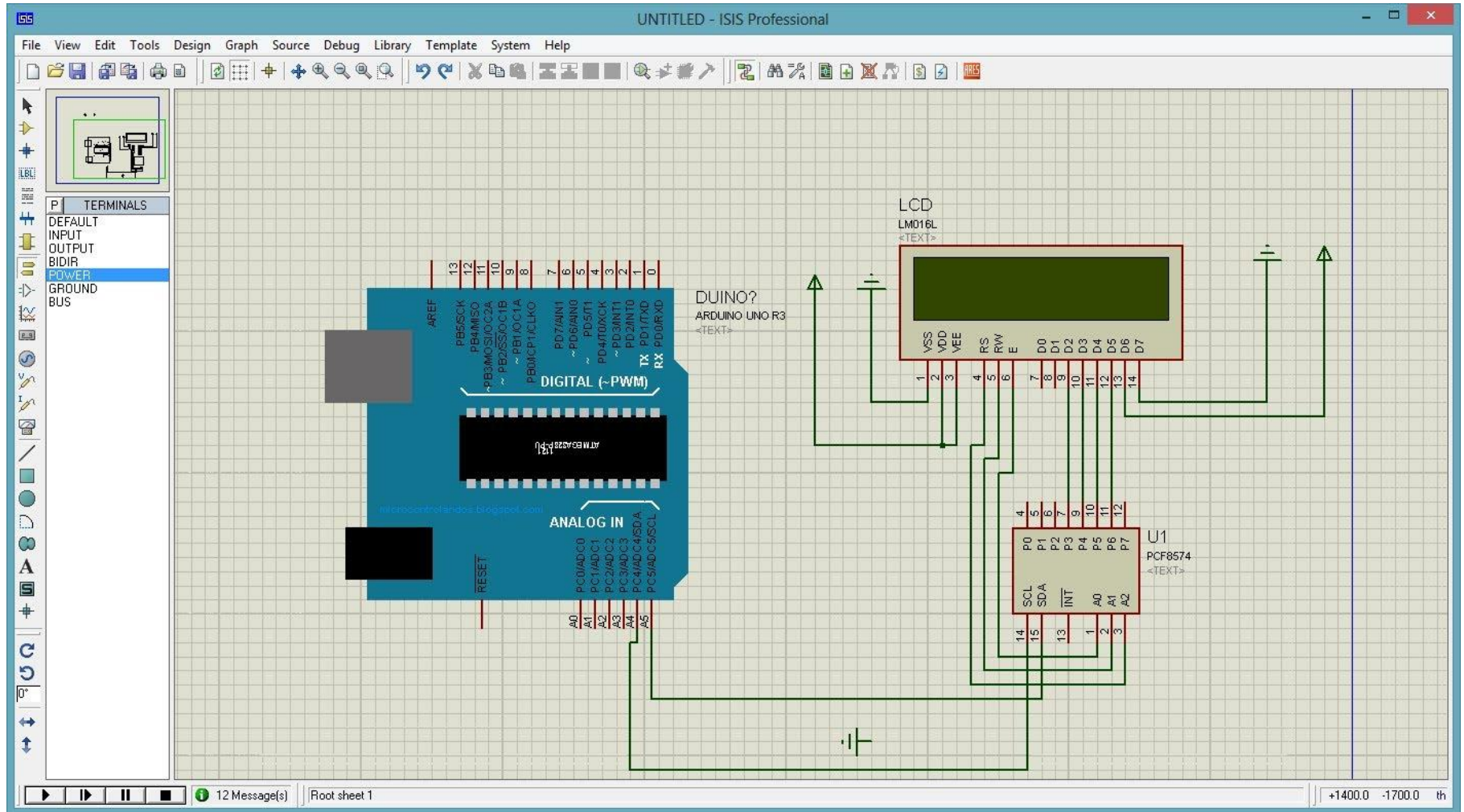
    if (!udc_include_vbus_monitoring()) {
        // VBUS monitoring is not available on this product
        // thereby VBUS has to be considered as present
        main_vbus_action(true);
    }
}
```

The Output window shows the build process, indicating that the target 'Build' was skipped due to a false condition, and the build succeeded. The IO View window displays the hardware components and registers, including the Flash Controller (FLASHC) and the Command (FCMD) register.

Name	Value	Bits	Address
FCR	0x00000000		0xFFE1400
FCMD	0x00000000		0xFFE1404
FSPR	0x00000000		0xFFE1408
FGPFRHI	0x00000000		0xFFE140C
FGPFRLO	0x00000000		0xFFE1410



# Herramientas de Simulación y Depuración– Proteus



# Bibliografía

- **The AVR Microcontroller and Embedded System.** Mazidi - Naimi. 1ra Ed. 2011.
- **The AVR Microcontroller and Embedded System.** Mazidi - Naimi. 2da Ed. 2017 (<https://nicerland.com/avr/>)
- **Los Microcontroladores AVR de ATMEL.** Felipe Espinoza. 2012 (pdf de uso libre)
- **Recursos web:** manuales, hojas de datos, notas de aplicación, ...

