NUEVAS REALIDADES PARA LA EDUCACIÓN EN INGENIERÍA: CURRÍCULO, TECNOLOGÍA, MEDIO AMBIENTE Y DESARROLLO
13 - 16 DE SEPTIEMBRE 2022 CARTAGENA DE INDIAS, COLOMBIA
Encuentro Internacional de Educación en Ingeniería ACOFI

# Emulación de sistemas como una metodología para la enseñanza de estrategias de control

**Jaiber Evelio Cardona Aristizábal, José Gabriel Hoyos Gutiérrez**

**Universidad del Quindío**
**Armenia, Colombia**

## Resumen

Cuando se están enseñando estrategias de control, es importante que la teoría sea aplicada tanto a nivel de simulación como en una implementación física. La simulación permite corroborar que la estrategia de control está bien aplicada y la implementación física permite enfrentarse con limitaciones como saturaciones, no linealidades, discretización, etc.

La simulación es de acceso más general en el sentido de que, al contar con un computador, se puede realizar en horarios más flexibles, en comparación con la implementación física donde se debe contar con un sistema prototipo al que no se tiene un acceso directo. Entre las limitaciones para acceder al sistema físico está la disponibilidad de horarios de los laboratorios, el número de estudiante esperando usar los sistemas y los horarios propios del estudiante.
Como opción para la implementación física se propone emular un sistema de prueba a través de un sistema embebido, de esta manera el control se haría sobre el sistema emulado acercándose al sistema real en una proporción mayor que la sola simulación.

Si el prototipo se emula a través de un sistema de bajo costo, como un Arduino, le daría al estudiante la facilidad de contar con la disponibilidad, sin esperar los turnos de laboratorio.
Como componente adicional se propone también usar un sistema de bajo costo, que puede ser también un Arduino, que permita remplazar una tarjeta de adquisición de datos o el controlador; en ese sentido el estudiante debería contar con 2 sistemas Arduino, uno para emular el sistema de prueba y otro para realizar el control. El sistema que realiza el control puede ser un sistema con un control embebido o un computador que usa el segundo sistema como tarjeta de adquisición de datos.

Esta estrategia se viene usando en la enseñanza del curso Sistemas de Control Automático del programa de ingeniería electrónica, en los cuales se desarrollan 7 laboratorios. En el primer laboratorio se enseña como emular sistemas en una tarjeta Arduino y como usar un Arduino como tarjeta de adquisición de datos. Luego, en cuatro laboratorios se diseñan e implementan estrategias de control las cuales son: PID, compensadores en adelanto y atraso, control RST y control en variables de estado con observador; estos laboratorios son desarrollados sobre un sistema Arduino emulado, pudiendo usar el otro Arduino como tarjeta de adquisición o usando las tarjetas de adquisición disponibles en el laboratorio. En los dos laboratorios restantes se hacen pruebas sobre sistemas prototipo presentes en el laboratorio o elaborados por los estudiantes, donde se aplica cualquiera de las técnicas ensayadas en emulación.

Si bien no se utiliza la emulación en todos los laboratorios, ya que se quiere que también tengan la experiencia de controlar un prototipo, dicha emulación permite que los estudiantes tengan una mayor flexibilidad en los laboratorios correspondientes, además de permitir que el profesor asigne diferentes sistemas a controlar a cada estudiante, evitando posibles copias de las soluciones presentadas y un mayor compromiso de parte de los estudiantes.

**Palabras clave**: emulación de sistemas; control discreto; modelado de sistemas

## *Abstract*

*When control strategies are teaching, it is important to apply the theory in both at the simulation level and in a physical implementation. The simulation allows corroborating that the control strategy is well applied, and the physical implementation allows dealing with limitations such as saturations, non-linearities, discretization, among others.*

*The simulation has more general access in the sense that, by having a computer, it can be carried out at more flexible times, in the other hand, the physical implementation requires an available prototype system, usually without direct access. Among the limitations to access the physical system are the availability of laboratories, the number of students waiting to use the systems and the own schedules of the students.*

*As an option for the physical implementation, it is proposed to emulate a test system through an embedded system, in this way the control would be done on the emulated system, approaching the real system in a greater proportion than the simulation alone.*

*If the prototype is emulated through a low-cost system, such as an Arduino, students will have the facility to carry out laboratory practices, without waiting for laboratory shifts.*

*As an additional component, it is also proposed to use a low-cost system, which can also be an Arduino, which allows replacing a data acquisition card or the controller; In this sense, the student should have two Arduino systems, one to emulate the test system and the other to carry out the control. The system that performs the control can be a system with an embedded control or a computer that uses the second system as a data acquisition card.*

*This strategy has been used in the teaching of the Automatic Control Systems course of the electronic engineering program, in which seven laboratories are developed. The first lab teaches how to emulate systems on an Arduino card and how to use an Arduino as a data acquisition card. Then, in four laboratories, control strategies are designed and implemented, which are: PID, lead–lag compensators, RST control and state space control with observers; These labs are developed on an emulated Arduino system, being able to use the other Arduino as a data acquisition card or using the data acquisition cards available in the laboratory. In the two remaining laboratories, tests are carried out on prototype systems present in the laboratory or systems elaborated by the students, where any of the techniques tested in emulation are applied.*

*Emulation is not used in all labs, because it is wanted that the students also have the experience of controlling a prototype. Emulation allows students greater flexibility in corresponding labs, as well as allowing the teacher to assign different systems to the labs groups, avoiding possible copies of the solutions presented and increasing the commitment of the students.*

***Keywords****: systems emulation; discrete control; systems modeling*

## 1. Introducción

En los cursos de Automatización y control de la Universidad del Quindío se da importancia a la parte práctica, la cual permite pasar del componente teórico y simulado, a los montajes con sistemas reales. La parte práctica considera dos horas semanales programadas, en la cual, por la disponibilidad de equipos, los estudiantes deben tomar turnos para utilizar los sistemas de control. Si dentro de las horas programadas no alcanzan a realizar la práctica, deben solicitar una práctica libre, en horarios adicionales, donde deben coincidir la disponibilidad del laboratorio, la disponibilidad del sistema de control y el horario del estudiante. Esto sumado al tiempo que se requiere para instalar y desinstalar los sistemas de control para poder ser usados y el acompañamiento que se requiere por parte del docente para una práctica exitosa.

La condición expuesta limita los alcances que se pueden lograr con la práctica, debiendo postular parte de los esquemas de control en simulación y solo una parte con una implementación física. Dentro de las prácticas, puede suceder también que algunos estudiantes compartan sus desarrollos con otros estudiantes menos dedicados, por lo cual, estos últimos no se apropian de los conocimientos, sino que simplemente usan los resultados ya anteriormente encontrados por otros grupos. Para poder contrarrestar esta condición, se requerirían de diferentes sistemas que puedan ser distribuidos entre diferentes grupos, de forma que cada grupo tenga una asignación diferente.

Para reducir el impacto de lo expuesto, se propone usar la emulación de sistemas dinámicos, que consiste en reproducir de forma física el comportamiento dinámico de un sistema. Está estrategia está siendo ampliamente utilizada en la experimentación debido a las ventajas en cuanto evitar daños que pueda producirse al manipular sistemas reales y a reducir costos asociados a la elaboración de prototipos. Se tienen reportes de emulación de sistemas dinámicos en temas tan diversos como máquinas de inducción (Tanaku et al, 2022), bicicletas eléctricas (Radrizzani et al, 2021), paneles fotovoltaicos (Suwito et al, 2021), Grids (Jia et al, 2021), entre otros.

La propuesta es emular un sistema dinámico en un sistema microprocesado, de forma que dicha emulación en hardware se use como el sistema a controlar, en la academia esta estrategia presenta las siguientes ventajas:

- Permite que cada estudiante tenga una asignación diferente, lo cual evita posibles copias entre grupos e incluso con estudiantes de cursos anteriores.
- Permite asignar tanto sistemas lineales como no lineales, pudiendo ensayar estrategias de control desde las básicas hasta las avanzadas
- Permite asignar tanto sistemas conocidos como desconocidos, por lo cual puede incluirse modelamiento a partir de datos adquiridos.
- Permite, si se usan sistemas de bajo costo, que los estudiantes tengan mayor flexibilidad al poder realizar las prácticas sin necesidad de desplazarse al laboratorio.
- Permite que todas las estrategias de control enseñadas puedan ser fácilmente ensayadas, no solo en simulación sino también en un montaje real.

Aunque estas ventajas generan una mayor apropiación de los conocimientos, el control sobre los prototipos reales es también importante, por lo que unas prácticas se realizan sobre los sistemas emulados y otras sobre los sistemas reales, acorde con la disponibilidad de estos últimos. También se motiva a los estudiantes a realizar sus propios prototipos en los que puedan ensayar una o varias estrategias de control.

En la siguiente sección se presenta la propuesta, tal como se está trabajando en el Programa de Ingeniería Electrónica, posteriormente se muestran los resultados obtenidos y posteriormente se presentan las conclusiones.

## 2. Propuesta de emulación de sistemas

Para la emulación de sistemas se propone usar un Arduino UNO, al cual se le programa el modelo de un sistema, como entrada se usa un ADC, usualmente A0, y como salida el PWM, al PWM se le conecta un filtro RC para suavizar la señal, dicho filtro debe tener entre 5 y 10 veces la frecuencia de muestreo con la que se realiza la implementación.

El primer modelo asignado a los estudiantes es una función de transferencia continua, el modelo se pasa a función de transferencia discreta y se implementa en ecuaciones de diferencia. Un ejemplo de este proceso se muestra en la figura 01, donde se tiene una función de transferencia continua G(s), el equivalente discreto G(z), el equivalente en ecuaciones de diferencia y(z) y el código correspondiente en Arduino

$$G(s) = \frac{0.02S^2 + 0.07926972S + 0.0112424}{S^2 + 0.114286S + 0.012220}$$

$$G(z) = \frac{0.02z^2 + 0.040280z - 0.0496675}{z^2 - 1.880467z + 0.892003}$$

$$Tm = 1\ s$$

EmulaSistema01 Arduino 1.8.8

Archivo Editar Programa Herramientas Ayuda

EmulaSistema01

```
const int analogInPin = A0; // Entrada análoga (Entrada del sistema)
const int digPin = 10;      // Pin para PWM (Salida del sistema)
const int Tm = 1000;        // Tiempo de muestreo en ms
float U = 0, U_1=0, U_2=0;  // Entrada del sistema y retrasos
float Y = 0, Y_1=0, Y_2=0;  // Salida del sistema y retrasos
void setup(){}
void loop() {
    // lectura del ADC y pasar del rango [0;1023] al rango [0;5]
    U = analogRead(analogInPin)*5.0/1023.0;
    // Ecuación de diferencia
    Y = 0.02*U+0.040280*U_1-0.0496675*U_2+1.880467*Y_1-0.892003*Y_2;
    U_2 = U_1;  // Guardar U(k-2) para el siguiente ciclo
    U_1 = U;    // Guardar U(k-1) para el siguiente ciclo
    Y_2 = Y_1;  // Guardar Y(k-2) para el siguiente ciclo
    Y_1 = Y;    // Guardar Y(k-1) para el siguiente ciclo
    if (Y<0) Y=0; // Límite inferior de la señal a 0v
    if (Y>5) Y=5; // Límite superior de la señal a 5V
    // convertir del rango [0;5] al rango [0;255] para escribir en el PWM
    analogWrite(digPin,Y*255/5);
    delay(Tm); // Espera el tiempo de muestreo definido
}
```

Guardado.

5 Arduino/Genuino Uno

$$y(k) = 0.02u(k) + 0.040280u(k-1) - 0.0496675u(k-2) + 1.880467y(k-1) - 0.892003y(k-2)$$

Figura 01. Emulación de una función de transferencia en Arduino

Otro modelo para asignar a los estudiantes está en variables de estado continuas, el modelo se pasa a variables de estado discretas y se implementa. Un ejemplo de este proceso se muestra en la Figura 02, donde se tiene un sistema en variables de estado continuas, el equivalente discreto y el código correspondiente en Arduino

$$\dot{X} = \begin{bmatrix} 0 & 0.04 \\ -0.305500 & -0.114286 \end{bmatrix} X + \begin{bmatrix} 0 \\ 0.25 \end{bmatrix} U$$

$$Y = \begin{bmatrix} 1.099800 & 0.307936 \end{bmatrix} X + 0.02\ U$$

$$X(k+1) = \begin{bmatrix} 0.9941222 & 0.0377220 \\ -0.2881018 & 0.8863448 \end{bmatrix} X(k) + \begin{bmatrix} 0.0048100 \\ 0.2357625 \end{bmatrix} u(k)$$

$$y(k) = \begin{bmatrix} 1.099800 & 0.307936 \end{bmatrix} X(k) + 0.02\ u(k)$$

$$T_m = 1s$$

EmulaSistemaVE01 Arduino 1.8.8

Archivo Editar Programa Herramientas Ayuda

EmulaSistemaVE01 §

```
const int analogInPin = A0; // Entrada análoga (Entrada del sistema)
const int digPin = 10;      // Pin para PWM (Salida del sistema)
const int Tm = 1000;        // Tiempo de muestreo en ms
double u,y;                 // Entrada y salida del sistema
double x1=0, x2=0, x1s, x2s; // Estados presentes y siguientes del sistema
void setup() {}
void loop() {
    // lectura del ADC y pasar del rango [0;1023] al rango [0;5]
    u = analogRead(analogInPin)*5.0/1023.0;
    // Ecuaciones de estado
    x1s = 0.9941222*x1+0.0377220*x2+0.0048100*u;
    x2s =-0.2881018*x1+0.8863448*x2+0.2357625*u;
    y = 1.099800*x1+0.307936*x2+0.02*u;
    x1=x1s; // Actualización de estado x1
    x2=x2s; // Actualización de estado x2

    if (y<0) y=0; // Límite inferior de la señal a 0v
    if (y>5) y=5; // Límite superior de la señal a 5V
    // convertir del rango [0;5] al rango [0;255] para escribir en el PWM
    analogWrite(digPin,y*255/5);
    delay(Tm); // Espera el tiempo de muestreo definido
}
```

21 Arduino/Genuino Uno

Figura 02. Ejemplo de emulación de una función de transferencia continua en Arduino

Otra posibilidad es estregar un archivo ". hex" para cargar directamente en el Arduino, en ese caso los estudiantes no conocen el código interno y deben enfrentarse a una función de transferencia desconocida.

El modelo emulado en Arduino UNO representa el sistema que se desea controlar; para aplicar alguna de las estrategias de control se llevan a cabo los pasos mostrados en la figura 03.

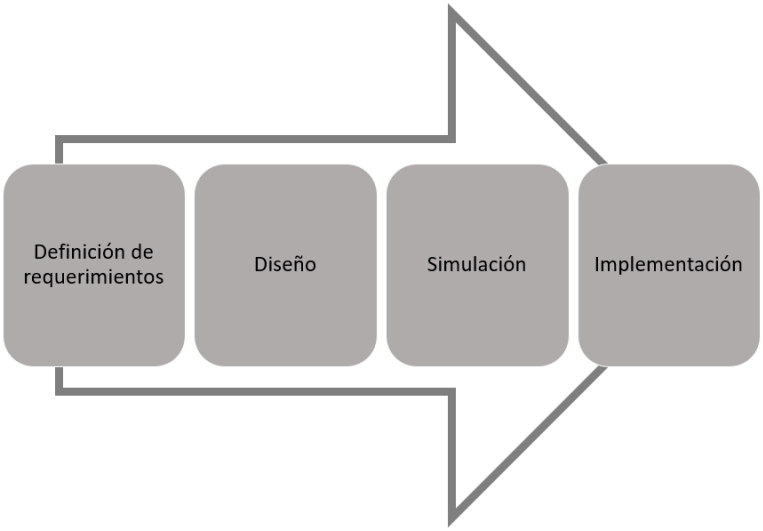


Figura 03. Pasos para controlar el sistema emulado.

Primero se definen los requerimientos de la estrategia de control a implementar, se diseña la estrategia de control según la teoría respectiva, se simula usando Simulink de Matlab y por último se implementa usando LabVIEW.

Para la implementación en LabVIEW se deben adquirir y enviar datos desde y hacia el sistema emulado, en este propósito se pueden usar tarjetas de adquisición de datos presentes en el laboratorio, por ejemplo, en la figura 03 se muestra la conexión entre la tarjeta National Instruments USB-6009 y el Arduino UNO.

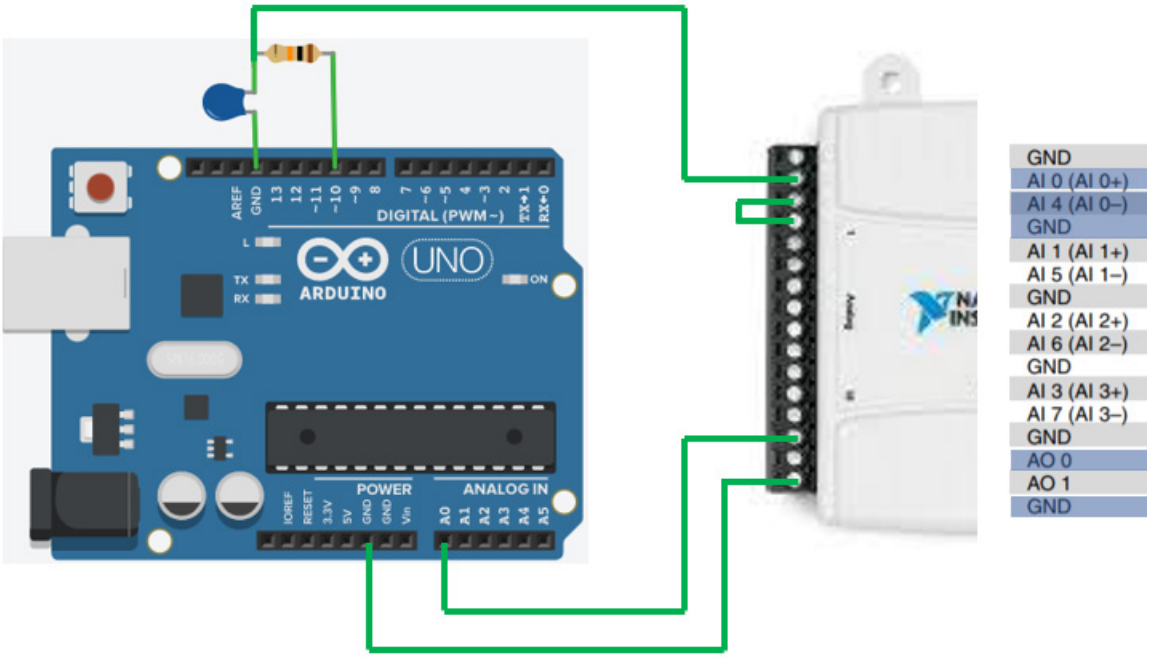


Figura 04. Conexión de la tarjeta USB-6009 con Arduino UNO.

En la figura 04, la entrada del sistema emulado A0 del Arduino UNO, está conectada a la salida de la tarjeta de adquisición de datos AO0, la salida del sistema emulado, salida del filtro RC, se conecta a la entrada de la tarjeta, el AI0. Como la entrada es diferencial AI4 se conecta a tierra junto con la tierra del Arduino UNO y de la tarjeta.

Como opción adicional la salida de 5V de la tarjeta de adquisición puede ser conectada al pin de 5V del Arduino como alimentación de voltaje.

Debido a que las tarjetas de adquisición deben ser también solicitadas en el laboratorio, limitando su uso a horarios restringidos, se propone usar otro Arduino UNO como tarjeta de adquisición de datos, en este caso el Arduino envía datos a través de un PWM y recibe datos a través del ADC A0. Esta opción permite que, contando con dos Arduinos, los estudiantes puedan realizar su práctica con mayor flexibilidad de horarios y materiales.

La conexión del Arduino que emula el sistema y el Arduino que remplaza la tarjeta de adquisición de datos se muestra en la figura 05. Se nota que ambos Arduinos tienen filtro RC, por lo que cualquiera puede usarse como emulador del sistema o como tarjeta de adquisición, aunque el que funciona como tarjeta de adquisición debe estar conectado al computador.

El código del Arduino como tarjeta de adquisición se muestra en la figura 06.

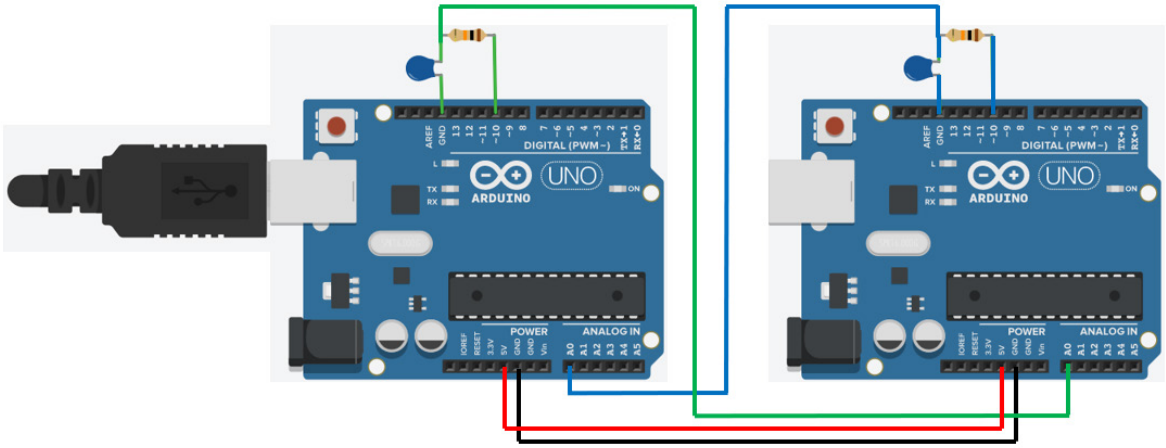


Figura 05 conexión de dos Arduinos UNO

AnalogInPWMOut Arduino 1.8.8

Archivo Editar Programa Herramientas Ayuda

AnalogInPWMOut

```
const int digPin = 10;       // Define PWM en pin 10
const int analogInPin = A0;  // Define como entrada el ADC A0
char seleccion;              // Variable para caracter de selección
int Sensor = 0;              // Variable para el valor del ADC A0
char Control = 0;            // Variable para la señal a enviar

void setup() {
  Serial.begin(9600);            // inicia comunicación serial
 }

void loop() {

  if (Serial.available() > 0) {                // Verifica si hay un nuevo byte para leer
    seleccion = Serial.read();                 // Lee el nuevo byte

    if (seleccion=='A')                        // 'A' indica lectura del ADC
        {
          Sensor = analogRead(analogInPin);    // Leer el dato de 10 bits
          Serial.write(Sensor/256);            // Envía parte alta del dato
          Serial.write(Sensor%256);            // Envía parte baja del dato
        };
    if (seleccion=='D')                        // 'D' indica que se escribirá en el PWM
        {
          while(Serial.available()==0);        // Espera a que se envíe un byte
           Control = Serial.read();            // Leer dato del número
           analogWrite(digPin,Control);        // Escribir en PWM
        };
  };
}
```

Guardado.

9 Arduino/Genuino Uno

Figura 06 Código del Arduino UNO como tarjeta de adquisición.

El código presentado en la figura 6 debe ser complementado con un programa desde LabVIEW, dicho programa debe enviar un carácter 'A' para indicar al Arduino que se desea leer el pin A0, a lo que el Arduino responde enviando dos caracteres que corresponden a 10 bits, también desde el LabVIEW se debe enviar un carácter 'D' seguido de un byte para indicar el valor que se escribirá en el PWM.

## 3. Resultados

Bajo este esquema se probaron con implementación física, a partir de la emulación de sistemas, los laboratorios correspondientes a control PID por ubicación de polos, Reglas de Ziegler Nichols, compensadores en adelanto y atraso, control RST (Landau, 1993) y control en variables de estado con observadores (Ogata, 1995). A cada grupo de estudiantes se les asignó un sistema diferente, logrando que se enfocaran en sus propios diseños.

Los grupos, con la implementación, lograron evidenciar fenómenos que no son directos en simulación, como saturaciones, ruido, señales de control limitadas y efecto windup de la integral.

Los grupos de trabajo no pueden copiar directamente los diseños de otros grupos, al ser sistemas diferentes, lo cual aumento el compromiso, el entendimiento y evito la copia.

Aunque todos los tipos de controladores fueron probados de forma emulada, también se asignaron proyectos en sistemas prototipo donde pudieron aplicar cualquiera de los métodos de control vistos, interactuando sobre sensores y actuadores según el sistema correspondiente.

## 4. Conclusiones

En este trabajo se presentó una propuesta para emular sistemas en función de transferencia como una posibilidad para realizar prácticas con los estudiantes del área de Automatización y Control. El uso de sistemas de bajo costo genera flexibilidad en los montajes prácticos al permitir que cada estudiante pueda realizar sus pruebas sin la necesidad de estar en el laboratorio y evita la copia entre estudiantes al permitir asignar sistemas diferentes entre una cantidad infinita que se pueden emular.

Esta estrategia permitió probar de forma práctica todos los temas presentados en el curso, es decir, control PID, compensadores, RST, variables de estado y observadores.

La emulación de sistemas se usa como forma de implementar estrategias de control de forma física, permite una mayor comprensión y compromiso, y logra un mayor entendimiento que se evidencia al realizar controles sobre prototipos reales.

## 5. Referencias


- Jia M., Averous N. R. and De Doncker R. W., "Evaluation and Comparison of Different Topologies for a Grid Emulator," 2021 IEEE 22nd Workshop on Control and Modelling of Power Electronics (COMPEL), 2021, pp. 1-8, doi: 10.1109/COMPEL52922.2021.9646040.
- Landau I.D. (1993). Identification et Commande des Systemes, (seconde edition), Hermes, Paris.
- Ogata K. (1995) Discrete-Time Control Systems. Prentice Hall. Universidad de Michigan.
- Radrizzani S., Panzani G., Corno M. and Savaresi S. M., "A concurrent controllers integration strategy to enhance the riding experience in bicycle driveline emulators," 2021 IEEE Conference on Control Technology and Applications (CCTA), 2021, pp. 1043-1048, doi: 10.1109/CCTA48906.2021.9659131.
- Suwito S., Rivai M., Ashari M. and Mustaghfirin M. A., "Simple and Fast Response Photovoltaic Panel Emulator using Transistor Current Source," 2021 International Conference on Electrical and Information Technology (IEIT), 2021, pp. 223-227, doi: 10.1109/IEIT53149.2021.9587441.
- Tanuku G. and Pillay P., "Emulation of induction machines subject to industrial grid harmonics," 2022 IEEE International Conference on Power Electronics, Smart Grid, and Renewable Energy (PESGRE), 2022, pp. 1-6, doi: 10.1109/PESGRE52268.2022.9715819.


## Sobre los autores


- **Jaiber Evelio Cardona Aristizábal**: Ingeniero Electrónico, Máster en Automatización, Doctor en Ingeniería de la Universidad del Valle. Profesor titular de la Universidad del Quindío. jaibercardona@uniquindio.edu.co
- **José Gabriel Hoyos Gutiérrez**: Ingeniero Electricista, Máster en Ingeniería Eléctrica, Doctor en Ingeniería de la Universidad Nacional de Colombia. Profesor asistente de la Universidad del Quindío. josegabrielh@uniquindio.edu.co


---


Los puntos de vista expresados en este artículo no reflejan necesariamente la opinión de la Asociación Colombiana de Facultades de Ingeniería.

Copyright © 2022 Asociación Colombiana de Facultades de Ingeniería (ACOFI)