

**UNIVERSIDAD AMERICANA**  
**Facultad de Ingeniería y Arquitectura**

---



---

**Capacitación ARDUINO Informe**  
**Tecnico:**  
“Aplicación Móvil de Eficiencia en Producción – EfiDuino”

---

**Autor:**

Kristel Geraldine Villalta Porras. Cif: 23020027

**Supervisor:**

Ing. Ned Lacayo Thompson

**Fecha:**

Viernes 23 de Mayo 2024

**Managua, Nicaragua**

## Índice

<b>Introducción.....</b>	<b>2</b>
<b>Planteamiento del Problema.....</b>	<b>3</b>
<b>Caracterización del Problema.....</b>	<b>4</b>
<b>Delimitación del Problema.....</b>	<b>5</b>
<b>Problema Científico e Interrogantes Científicas Problema Científico.....</b>	<b>5</b>
<b>Interrogantes Científicas.....</b>	<b>5</b>
<b>Objetivo.....</b>	<b>6</b>
<b>Objetivo General.....</b>	<b>6</b>
<b>Objetivos Específicos.....</b>	<b>6</b>
<b>Justificación.....</b>	<b>7</b>
<b>Marco Teórico.....</b>	<b>8</b>
<b>Arduino como Plataforma de Prototipado.....</b>	<b>8</b>
<b>Sensores y Componentes Electrónicos.....</b>	<b>8</b>
<b>Sensor Infrarrojo (IR).....</b>	<b>8</b>
<b>Motor DC con Encoder.....</b>	<b>8</b>
<b>Módulo Bluetooth HC06.....</b>	<b>8</b>
<b>Otros componentes.....</b>	<b>9</b>
<b>Aplicación Móvil iOS: EfiDuino.....</b>	<b>9</b>
<b>Conexión y Circuito Eléctrico.....</b>	<b>9</b>
<b>Implementación Técnica en Arduino.....</b>	<b>10</b>
<b>Presupuesto Estimado.....</b>	<b>12</b>
<b>Conclusión.....</b>	<b>14</b>
<b>Bibliografía.....</b>	<b>15</b>

## **Introducción**

En la actualidad, la eficiencia en los procesos productivos es un factor clave para la competitividad de cualquier empresa o taller. Poder medir en tiempo real la cantidad de productos procesados y la velocidad de los sistemas mecánicos permite tomar decisiones informadas que optimicen el rendimiento general del sistema.

El presente proyecto, titulado Monitor de Eficiencia en Producción – EfiDuino, surge como una solución tecnológica accesible y funcional que combina electrónica, programación embebida y desarrollo móvil. Utilizando un microcontrolador Arduino, sensores y un módulo Bluetooth, el sistema detecta y cuenta productos en una línea de producción, mide la velocidad del motor mediante un encoder, y transmite esta información a una aplicación móvil iOS desarrollada en Swift.

La aplicación, denominada EfiDuino, permite visualizar las estadísticas en tiempo real, generar gráficos dinámicos y exportar los resultados en diferentes formatos. Esta solución busca ser útil tanto en contextos industriales como educativos, demostrando que es posible implementar sistemas de monitoreo eficientes con herramientas de bajo costo y alta adaptabilidad.

## **Planteamiento del Problema**

En muchas líneas de producción, especialmente en pequeñas y medianas empresas, no existe un sistema automatizado que permita monitorear en tiempo real la eficiencia del proceso. Esta falta de información dificulta la toma de decisiones orientadas a mejorar la productividad, ya que los datos suelen registrarse de forma manual o simplemente no se registran.

Además, no contar con un mecanismo que mida cuántos productos se procesan por minuto ni con registros de la velocidad del motor impide identificar cuellos de botella, tiempos muertos o caídas en el rendimiento. La ausencia de estos indicadores puede derivar en ineficiencia operativa, aumento de costos y menor competitividad.

Por otro lado, muchas de las soluciones existentes para monitoreo industrial son costosas, complejas o requieren conocimientos técnicos avanzados para su implementación. Esto crea una brecha tecnológica para talleres, laboratorios y pequeñas líneas de montaje que buscan mejorar sus procesos sin grandes inversiones.

Por tanto, se hace necesario desarrollar una herramienta de bajo costo, fácil implementación y alta utilidad, que permita monitorear, registrar y analizar la eficiencia de producción en tiempo real, con capacidad de adaptación a distintas configuraciones de línea y que pueda integrarse con dispositivos móviles para una mayor accesibilidad.

## **Caracterización del Problema**

El problema se manifiesta principalmente en entornos de producción donde no se utilizan sistemas automáticos para medir el rendimiento de la línea de ensamblaje o manufactura. En estos espacios, la eficiencia se determina de manera empírica o simplemente no se evalúa, lo que limita la capacidad de detectar fallos, retrasos o pérdidas de productividad.

Asimismo, la falta de datos en tiempo real impide hacer análisis comparativos o históricos, dejando a las empresas sin herramientas efectivas para planificar mejoras, mantener estándares de calidad o medir el cumplimiento de metas de producción.

La ausencia de digitalización en este aspecto suele estar asociada a tres factores clave:

- Costos elevados de implementación de soluciones industriales.
- Complejidad técnica de sistemas comerciales.
- Falta de personal capacitado para configurar o mantener equipos avanzados.

Este problema afecta principalmente a pequeñas industrias, centros de formación técnica, talleres y laboratorios que no cuentan con recursos suficientes para adquirir o mantener tecnología de monitoreo avanzada.

## **Delimitación del Problema**

Desarrollo de un Sistema de Monitoreo de Eficiencia Productiva Mediante Arduino y Comunicación Bluetooth: Aplicación en una Línea de Producción Simulada con Visualización en iOS durante el Primer Semestre de 2025.

## **Problema Científico e Interrogantes Científicas**

### **Problema Científico**

¿Cómo puede desarrollarse un sistema accesible y funcional que permita monitorear en tiempo real la eficiencia de una línea de producción, utilizando tecnología de bajo costo como Arduino y comunicación Bluetooth, y visualizando los resultados en una aplicación iOS?

### **Interrogantes Científicas**

1. ¿Qué tipo de sensores permiten un conteo preciso de productos en movimiento?
2. ¿Cómo se puede calcular la eficiencia productiva en función de una meta predefinida utilizando hardware embebido?
3. ¿Qué mecanismos de comunicación permiten transmitir datos desde un Arduino a una aplicación móvil de forma confiable?
4. ¿Cómo puede una app iOS interpretar, graficar y exportar datos de producción en tiempo real?
5. ¿Qué limitaciones presenta un sistema de bajo costo en comparación con soluciones industriales más complejas?

## **Objetivo**

### **Objetivo General**

Desarrollar un sistema de monitoreo de eficiencia productiva utilizando un microcontrolador Arduino, sensores de conteo y velocidad, y comunicación Bluetooth, con visualización y exportación de datos en una aplicación iOS, aplicado a una línea de producción simulada durante el primer semestre de 2025.

### **Objetivos Específicos**

1. Diseñar e implementar un sistema de detección de productos en una línea de producción simulada utilizando un sensor infrarrojo y un encoder.
2. Programar un algoritmo en Arduino capaz de calcular la eficiencia productiva con base en una meta establecida de producción por minuto.
3. Desarrollar una aplicación móvil iOS que reciba, procese y visualice los datos de eficiencia.

## **Justificación**

La optimización de los procesos productivos es una necesidad constante en la industria moderna, especialmente en un entorno cada vez más competitivo y automatizado. Las empresas buscan mejorar la eficiencia, reducir tiempos muertos y minimizar desperdicios, por lo que contar con herramientas de monitoreo accesibles, precisas y en tiempo real se vuelve indispensable. Sin embargo, muchas de estas soluciones requieren inversiones elevadas en sistemas industriales complejos, lo que las hace inalcanzables para pequeñas y medianas empresas o entornos educativos.

Ante esta necesidad, el presente proyecto propone el desarrollo de un sistema de monitoreo de eficiencia productiva accesible y funcional, utilizando un microcontrolador Arduino y sensores económicos, en conjunto con una aplicación móvil iOS que permite visualizar, analizar y exportar los resultados. Esta solución es aplicable tanto en contextos simulados como en ambientes reales de baja escala.

El sistema permite medir la cantidad de productos procesados en un período determinado, calcular la eficiencia en función de una meta establecida y monitorear la velocidad del motor mediante un encoder. Además, la comunicación Bluetooth ofrece una conexión inalámbrica directa con la aplicación EfiDuino, lo que elimina la necesidad de equipos intermedios y facilita la movilidad del usuario. La app permite visualizar gráficos en tiempo real y exportar los datos, mejorando así el acceso a la información para la toma de decisiones.

La implementación de este proyecto no solo tiene un impacto positivo en términos de eficiencia productiva, sino también en el ámbito educativo y tecnológico. Al integrar hardware, software y conectividad inalámbrica, se promueve el aprendizaje multidisciplinario, el desarrollo de habilidades en electrónica y programación, y la innovación en soluciones de automatización accesibles.



## **Marco Teórico**

El desarrollo de soluciones tecnológicas accesibles y eficientes ha cobrado una gran importancia en entornos educativos e industriales. En este sentido, el uso de plataformas como Arduino permite la creación de prototipos funcionales de bajo costo que pueden adaptarse a diferentes necesidades. Este proyecto se basa en la implementación de un sistema de monitoreo de eficiencia en una línea de producción simulada, combinando electrónica básica, sensores, comunicación inalámbrica y visualización de datos en tiempo real a través de una aplicación móvil.

A continuación, se describen los principales fundamentos técnicos que sustentan este proyecto:

### **Arduino como Plataforma de Prototipado**

Arduino es una plataforma de código abierto basada en hardware y software fácil de usar. Su versatilidad y bajo costo la han convertido en una herramienta ideal para estudiantes, educadores y desarrolladores. En este proyecto se utilizó un Arduino UNO R4 MÍNIMA, que permite controlar sensores, leer datos del entorno y comunicarse con otros dispositivos

### **Sensores y Componentes Electrónicos**

#### **Sensor Infrarrojo (IR)**

Se utilizó para la detección de productos que pasan frente a él a menos de 30 cm de distancia. Este sensor permite el conteo de objetos durante un período de tiempo determinado.

#### **Motor DC con Encoder**

El motor impulsa la cinta transportadora simulada, y el encoder integrado permite medir la velocidad de rotación. Esta información es clave para evaluar el rendimiento del sistema mecánico.

#### **Módulo Bluetooth HC06**

Este módulo permite la transmisión de datos entre el sistema Arduino y la aplicación móvil EfiDuino mediante comunicación Bluetooth. La conexión inalámbrica ofrece comodidad y evita el uso de cables adicionales.

### **Otros componentes**

Incluye jumpers, una fuente de alimentación externa (laptop ) y una estructura básica que simula la línea de producción.

### **Aplicación Móvil iOS: EfiDuino**

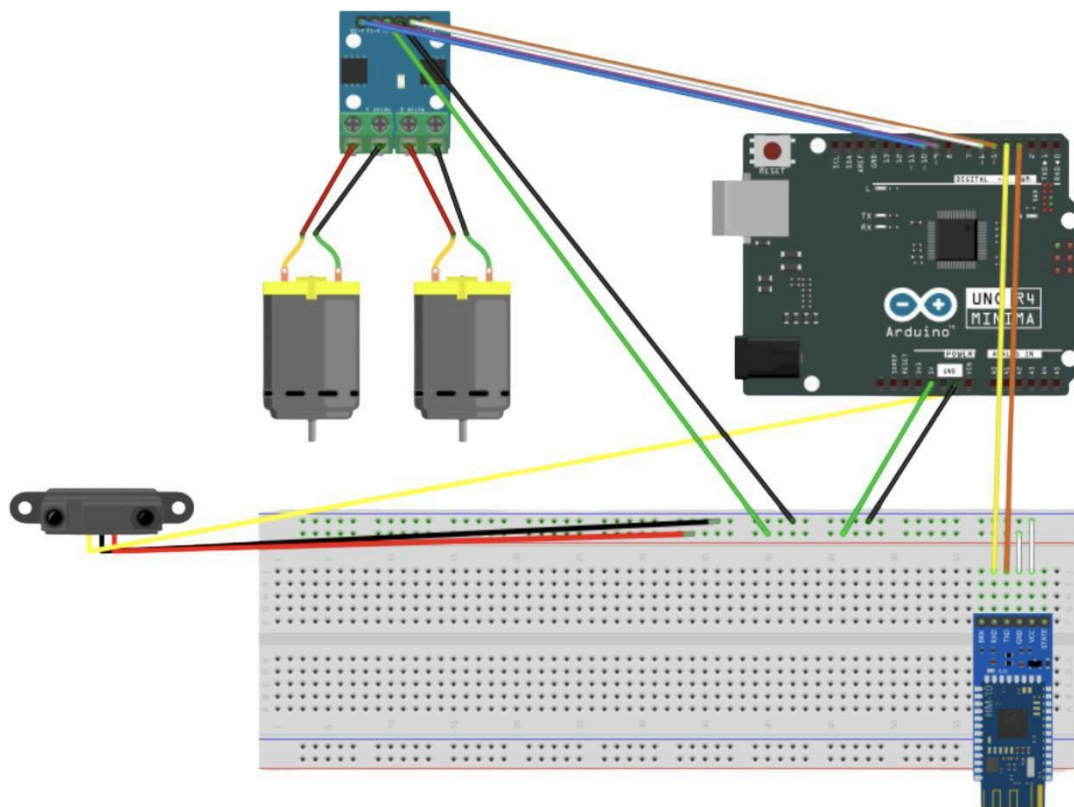
La app EfiDuino, desarrollada en Swift, permite recibir y visualizar los datos enviados por Arduino en tiempo real. Entre sus funciones se incluyen:

- Visualización de estadísticas de producción.
- Gráficos de eficiencia.
- Exportación de resultados en PDF, PNG y Excel.

Esta integración entre hardware y software fomenta un enfoque interdisciplinario entre electrónica y desarrollo móvil.

### **Conexión y Circuito Eléctrico**

El sistema ha sido cableado y probado correctamente. Las conexiones se establecen entre los pines del Arduino, el sensor IR, el motor y el módulo Bluetooth, conforme al siguiente esquema:



## Implementación Técnica en Arduino

El sistema de monitoreo se programó en lenguaje C++ utilizando el entorno de desarrollo Arduino IDE. A continuación, se detalla el código fuente que permite realizar el conteo de productos mediante un sensor infrarrojo, calcular pulsos por segundo con un encoder y transmitir los datos vía Bluetooth al dispositivo móvil.

Este código refleja la lógica implementada en el microcontrolador para capturar los datos relevantes de producción:

```
#include <SoftwareSerial.h>
// Pines del sensor infrarrojo
#define sensorIR A0
int conteo = 0;
bool objetoDetectado = false;

// Tiempo de ejecución (60 segundos)
unsigned long tiempoInicio;
const unsigned long duracion = 60000;
bool enEjecucion = true;
```

```

// Configuración del módulo Bluetooth JDY-31
const int bluetoothTx = 3;
const int bluetoothRx = 4;
SoftwareSerial bleSerial(bluetoothTx, bluetoothRx);

// Motor DC con encoder
const int motorA = 9;
const int motorB = 10;
const int encoderPin = 2; // Pin de interrupción

volatile int pulsos = 0;

void contarPulsos() {
    pulsos++;
}

void setup() {
    Serial.begin(9600);
    bleSerial.begin(9600);

    pinMode(motorA, OUTPUT);
    pinMode(motorB, OUTPUT);

    pinMode(encoderPin, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(encoderPin), contarPulsos, RISING);

    tiempoInicio = millis();
    Serial.println("Iniciando conteo por 1 minuto...");
}

void loop() {
    if (bleSerial.available()) Serial.write(bleSerial.read());
    if (Serial.available()) bleSerial.write(Serial.read());

    digitalWrite(motorA, HIGH);
    digitalWrite(motorB, LOW);

    if (enEjecucion) {
        unsigned long tiempoActual = millis();
        if (tiempoActual - tiempoInicio >= duracion) {
            enEjecucion = false;
            float eficiencia = (conteo / 20.0) * 100.0;
            Serial.println("\nTiempo finalizado.");
        }
    }
}

```

```

    Serial.print("Total de productos detectados: ");
    Serial.println(conteo);
    Serial.print("Eficiencia: ");
    Serial.print(eficiencia);
    Serial.println("%");
    return;
}

float volts = analogRead(sensorIR) * 0.0048828125;
int distancia = 13 * pow(volts, -1);
if (distancia <= 15) {
    if (!objetoDetectado) {
        conteo++;
        objetoDetectado = true;
    }
} else {
    objetoDetectado = false;
}

Serial.print("Distancia: ");
Serial.print(distancia);
Serial.println(" cm");

Serial.print("Pulsos: ");
Serial.println(pulsos);
pulsos = 0;

delay(200);
}
}

```

Este fragmento de código forma la base lógica del sistema de monitoreo, y su correcta ejecución es crucial para el funcionamiento sincronizado entre el hardware y la aplicación EfiDuino.

**Nota:** Este sistema fue desarrollado como parte de la capacitación de Arduino, pero el código y los componentes electrónicos representan un módulo independiente de la aplicación móvil, la cual se desarrolla en paralelo como una interfaz de visualización.

## Presupuesto Estimado

Componente	Precio Aproximado (USD)
Arduino UNO R4 MÍNIMA	\$20.00
Sensor Infrarrojo (IR)	\$10.00
Motor DC con Encoder	\$10.00
Módulo Bluetooth HC06/JDY-31	\$15.00
Jumpers y Protoboard	\$20.00
<b>Total</b>	<b>\$75.00</b>

## **Conclusión**

El proyecto logró cumplir con el objetivo principal de desarrollar un sistema funcional capaz de monitorear la eficiencia en una línea de producción simulada, utilizando Arduino y comunicación Bluetooth. La integración de sensores, un motor con encoder y un módulo inalámbrico permitió capturar datos en tiempo real, calcular el rendimiento del sistema y transmitir la información hacia una aplicación móvil desarrollada en iOS.

La app EfiDuino implementó este sistema al ofrecer una interfaz intuitiva para visualizar estadísticas, generar gráficos dinámicos y exportar los resultados en diversos formatos, facilitando así la interpretación de datos y su documentación. Esta solución demuestra cómo, con recursos accesibles, es posible implementar herramientas que pueden tener aplicaciones reales en entornos productivos o académicos.

Además, el proyecto promueve el aprendizaje interdisciplinario entre electrónica, programación embebida y desarrollo de software móvil, y ofrece una base sólida para futuras mejoras como el análisis de eficiencia por intervalos más cortos tomando en cuenta la integración con sensores adicionales.

## Bibliografía

Banzi, M., & Shiloh, M. (2014). *Getting Started with Arduino* (3rd ed.). Maker Media, Inc. Monk, S. (2016). *Programming Arduino: Getting Started with Sketches* (2nd ed.). McGraw-Hill Education.

Apple Inc. (2024). *Swift Programming Language Documentation*. Recuperado de <https://developer.apple.com/swift>

Apple Inc. (2024). *Human Interface Guidelines*. Recuperado de <https://developer.apple.com/design/human-interface-guidelines/>

GitHub. (2024). *Repositorio del Proyecto: Production Efficiency Monitor*. Recuperado de [Repositorio en GitHub – Production Efficiency Monitor](#)