

Uso de tarjetas Arduino en la educación superior tecnológica

Palacios Acosta - Ever A ¹; Proaño Lapuerta - Edison A ²

RESUMEN

La presente investigación hace referencia al uso de tarjetas Arduino en la educación superior, ante el deseo de poder desarrollar proyectos en aras de una educación de calidad y un perfeccionamiento prominente, se quiere conocer si a dichas tarjetas se las puede implementar en aplicaciones industriales y en la educación, para ello, primero se realizan encuestas a empresas para conocer si están al tanto de la plataforma Arduino y si disponen de sistemas implementados con dichas tarjetas, en segundo lugar, se realiza una aplicación industrial con HMI, con dos arquitecturas diferentes, una en Arduino y otra en un PLC, por último se elabora un análisis comparativo de las aplicaciones para determinar si la tarjeta Arduino UNO es aplicable en la industria.

Palabras Clave: Arduino, Educación Superior Tecnológica, Aplicación Industrial, Automatización, PLC, HMI.

1. Docente del Instituto Superior Tecnológico Luis Tello Carrera de Tecnología Superior en Automatización, Esmeraldas epalacios@institutos.gob.ec
2. Docente del Instituto Superior Tecnológico Luis Tello Carrera de Tecnología Superior en Automatización, Esmeraldas eproaño@institutos.gob.ec

Fecha de recepción: 25 / 04/ 2022

Fecha de aceptación: 25/ 06/ 2022

Use of Arduino boards in higher technology education

ABSTRACT

This research refers to the use of Arduino cards in higher education, in view of the desire to be able to develop projects for the sake of quality education and prominent improvement, we want to know if these cards can be implemented in industrial applications and in education, for this, first companies are surveyed to find out if they are aware of the Arduino platform and if they have systems implemented with said cards, secondly, an industrial application with HMI is carried out, with two different architectures, one in Arduino and another in a PLC, finally a comparative analysis of the applications is made to determine if the Arduino UNO board is applicable in the industry.

Keywords: Arduino, Technological Higher Education, Industrial Application, Automation, PLC, HMI.

INTRODUCCIÓN

La presente investigación hace referencia al uso de tarjetas Arduino en la educación superior, considerando que esta es una plataforma de fácil acceso en cuanto a precios y que aportan para la enseñanza a diferentes niveles educativos (Romo, Rogelio, & Martí, 2015).

La característica principal de esta plataforma es su software y hardware libres, lo que permite que sea utilizado en la enseñanza (Romo, Rogelio, & Martí, 2015). Ahora bien, lo que se necesita saber es si Arduino es una buena herramienta en los procesos de formación en la educación superior tecnológica, tomando en cuenta que los procesos industriales deben cumplir con una serie de características, las mismas que se desea comprobar si la plataforma a la que se hace referencia las puede cumplir.

Para analizar esta problemática es necesario indicar, porque algunas instituciones de educación superior tecnológica se acogen a la enseñanza basada en esta plataforma libre (Aguirre & Garcia, 2017). Una de las principales causas es la falta de talleres adecuados para la enseñanza en este nivel y otra no menos importante es que los estudiantes pueden acceder a esta plataforma

gratuitamente en relación del software y a precios módicos haciendo referencia al hardware.

Basándose en estos antecedentes lo que se realizó fue una encuesta a dos de las empresas con las que la institución tiene convenios para prácticas, esta encuesta fue sobre si conocían de aplicaciones dentro del entorno industrial implementadas con Arduino, con los datos obtenidos se realizó la aplicación básica “cambio de giro de un motor AC con un sistema HMI”, utilizando elementos industriales como son, un Controlador Lógico Programable (PLC), InTouch Wonderware en su versión de prueba y un motor AC y a su vez la misma aplicación usando elementos como, Arduino UNO, Monitoriza versión educativa, un módulo de relés y el mismo motor AC, con el fin de comparar su funcionamiento y demostrar si los estudiantes al usar Arduino en el entorno de formación van a ser capaces de entender, manejar y dar mantenimiento fácilmente un proceso industrial en el ámbito laboral.

METODOLOGÍA

Tomando como base las encuestas realizadas en dos industrias y una aplicación utilizada en la industria como es el control de un motor a través de un

sistema *HMI*, se contempla una fase de diseño de las dos aplicaciones en la cual la una se basa en la utilización de *PLC* como tarjeta de adquisición de datos, con un software de desarrollo de este tipo de sistemas altamente conocido en la industria como es *InTouch* y la otra aplicación basada en *Arduino* como tarjeta de adquisición de datos, un módulo relé como interface de potencia y el software *Monitoriza* en versión estudiantil como complemento del sistema a estudiar.

Análisis de Encuestas

A través de los diálogos realizados con las empresas de la ciudad de Esmeraldas con las cuales el Instituto tiene convenios de prácticas, se decidió trabajar en base a una encuesta que permitirá conocer si los jefes talleres del área de automatización han utilizado, utilizan o estarían dispuestos a implementar procesos con *Arduino* dentro de sus áreas de trabajo; obteniendo los siguientes resultados:

Los jefes de talleres manifestaron conocer y haber utilizado la plataforma de *Arduino* en simulaciones de procesos de enseñanza en el ámbito de su formación profesional, en cual se indica que se trabajó con una de las placas más básicas como es el *Arduino Uno*, además manifiesta que al haber trabajado en pocos proyectos con *Arduino* no se pudo verificar la vida útil del mismo, por último, se indica que dentro la empresa que labora no se ha realizado trabajos basados en *Arduino*, pero que si estarían dispuestos a realizar pequeños proyectos que no impliquen riesgos mayores o proyectos que contemplen seguridades secundarias.

En base a estos resultados obtenidos de las encuestas y tomando en consideración que se recomendó que se realice una aplicación que no implique mayores riesgos, se tomó la decisión de implementar un sistema *HMI* para manejo y cambio de giro de un motor monofásico AC.

Aplicación industrial mediante PLC SIEMENS y Wonderware InTouch

Se procede a la elaboración de una aplicación sencilla pero común en la industria, como se había dicho con anterioridad se tomará para esta primera fase softwares industriales por lo se los utilizará en su versión de prueba o demo, que se los irá describiendo conjuntamente con el desarrollo de la aplicación, la cual es su sistema *HMI* contará con 3 pulsadores, un paro de emergencia, el motor y luces piloto para visualización de arranque del motor.

Para Realizar esta aplicación se utilizarán tres programas industriales *LOGO!Soft Comfort*, *KEPserverEX* e *InTouch*. Estos permitirán mediante su interacción crear un sistema de adquisición de datos y supervisión para el proceso ya establecido, y de esta manera poder analizar cómo es el funcionamiento, aplicabilidad y respuesta de comunicación.

Programación con Logo!Soft Comfort V8.2: Este es un software que permite crear los programas con las instrucciones para el PLC (Barrera, Mendoza, Neira, & Pardo, 2019). En este se elaboró una aplicación en diagrama de funciones (FUP), que permite cambiar de giro a un motor monofásico, como se muestra en la Fig. 1

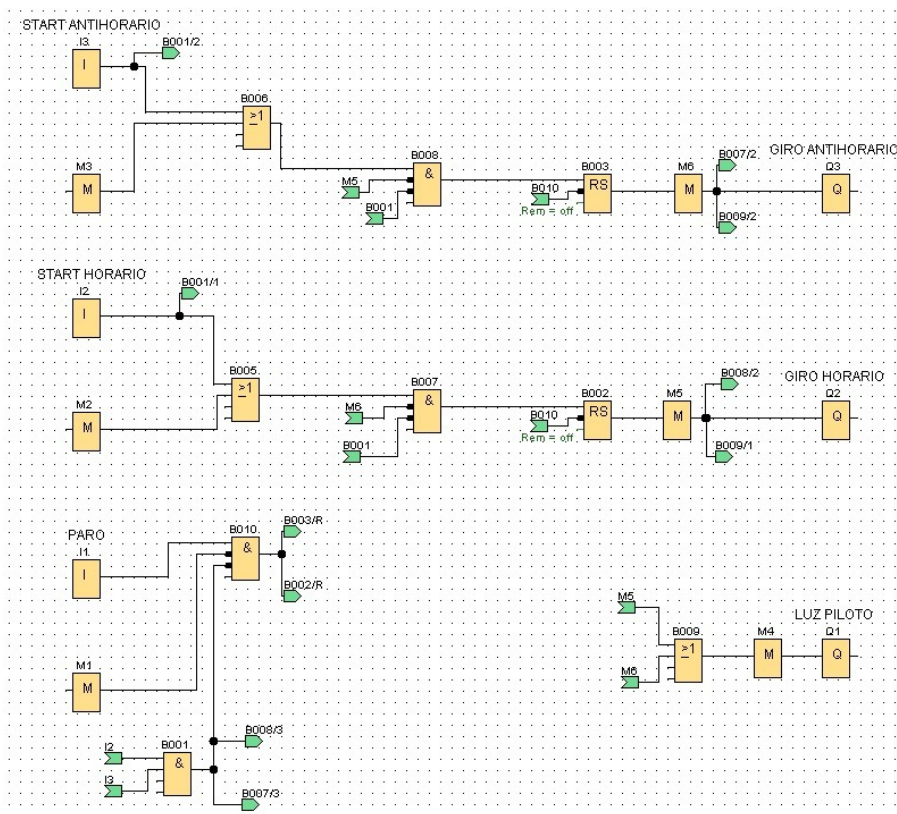


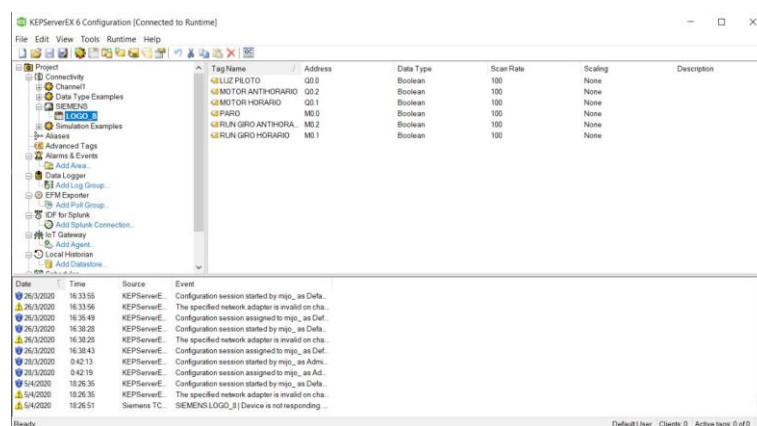
Fig. 1. Diagrama de funciones del cambio de giro de un motor.

Una vez realizado este proceso se procede al siguiente software, que permitirá establecer la comunicación entre el PLC el sistema HMI.

KEPServerEx V6.6: El software KepsServer EX creado por Kepware Technologies. Es uno de los mejores programas para interactuar del PLC con el

HMI (Sujith & Hemalatha, 2014). En este software se crearon las TAG's a utilizar en el OPC Server, es decir, se asigna un nombre a cada marca de entrada y salida del programa del PLC para que el HMI pueda acceder a dicho servidor y así leer o escribir cualquier variable asignada, esto lo vemos en la Fig. 2.

Fig. 2. Creación de TAG's para el OPC server.



InTouch V10.0: InTouch es un tipo de software SCADA disponible. A fin de que Organice datos de procesos y bases de datos, Wonderware proporciona el Administrador de procesos y el Administrador de acceso SQL (Wikacsono, Lim, & Sutanto, 2016). En

este software se realiza el HMI que cuenta con 3 pulsadores, el paro de emergencia, el giro horario y el giro antihorario; se visualiza una luz piloto cuando arranca el motor y 2 indicadores en forma de flechas que indican el sentido del giro, como se muestra en la Fig. 3.

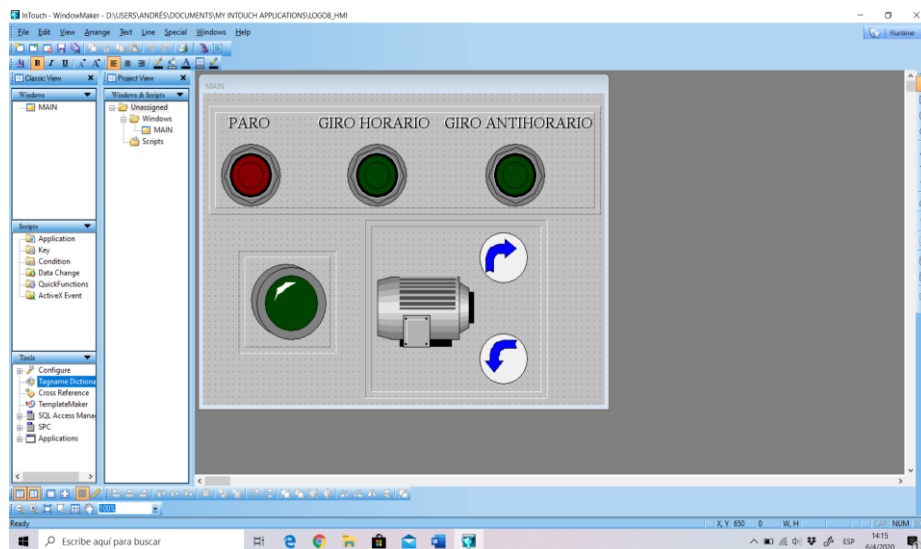


Fig. 3. Aplicación en Wonderware Intouch

Una vez realizada la comunicación por OPC server se procede a la prueba de funcionamiento, en la Fig. 4 se muestra el

sistema funcionando en giro horario, esto se puede hacer presionando un pulsador físico NA o en virtual.



Fig. 4. Funcionamiento giro horario.

Para el giro antihorario primero se debe parar el proceso y proceder a presionar el otro pulsador NA en físico o Virtual, y como en todo proceso se tiene un paro de

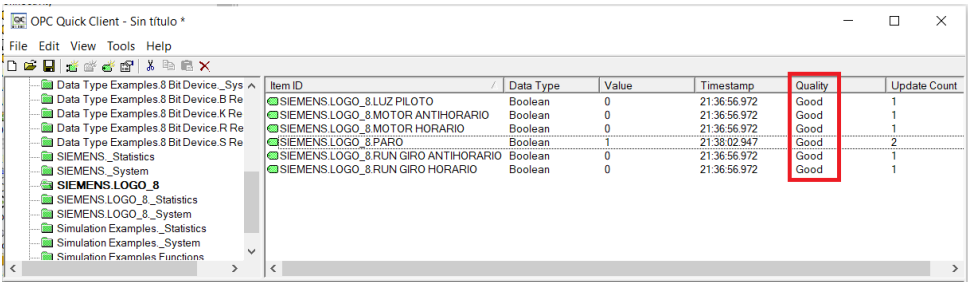
emergencia que al presionar pulsador NC de “paro” físico o virtual el motor y la luz piloto se apagan, en la Fig. 5 se muestra el giro antihorario.



Fig. 4. Funcionamiento giro antihorario.

Después de realizar las pruebas de funcionamiento se obtuvo una buena comunicación con el OPC server, para

sincronizar en tiempo real con el HMI en InTouch, esto se observa en la Fig. 5.



Item ID	Data Type	Value	Timestamp	Quality	Update Count
SIEMENS.LOGO_8.LUZ PILOTO	Boolean	0	21:36:56.972	Good	1
SIEMENS.LOGO_8.MOTOR ANTIHORARIO	Boolean	0	21:36:56.972	Good	1
SIEMENS.LOGO_8.MOTOR HORARIO	Boolean	0	21:36:56.972	Good	1
SIEMENS.LOGO_8.PARO	Boolean	1	21:38:02.947	Good	2
SIEMENS.LOGO_8.RUN GIRO ANTIHORARIO	Boolean	0	21:36:56.972	Good	1
SIEMENS.LOGO_8.RUN GIRO HORARIO	Boolean	0	21:36:56.972	Good	1

Fig. 5. Comunicación mediante OPC server.

Aplicación industrial mediante Arduino UNO y Monitoriza ACIMUT

Una vez realizada y comprobada la aplicación que de aquí en adelante le llamaremos PLC-InTouch, se procede a la elaboración de la misma aplicación, en este caso se basará en el uso de Arduino Uno y Monitoriza que son usados más para enseñanza que en la industria, estos software son de uso libre el primero y el segundo tiene una versión gratuita educativa, que se los irá describiendo conjuntamente con el desarrollo de la aplicación, la cual es su sistema HMI contará con 3 pulsadores, un paro de emergencia, el motor y luces piloto para visualización de arranque del motor.

La comunicación que se realizó fue una Modbus RTU, para lo cual se necesito un OPC server para Arduino, el cual permitió la comunicación y mejor manejo de datos.

Arduino IDE: El software Arduino de código abierto (IDE) hace que sea fácil escribir código y subirlo a la placa. Se ejecuta en Windows, Mac OS X y Linux. El entorno está escrito en Java y se basa en el procesamiento y otro software de código abierto. Este software se puede usar con cualquier placa Arduino

(Arduino , 2020). En este software se elaboró una aplicación sencilla de cambio de giro de motor por casos, en la misma se incluyó los protocolos para la comunicación a través del protocolo MODBUS RTU, como se indica en los algoritmos.

Algoritmo 1. Programación en C, realizada en Arduino, cambio de giro de motor y configuración de TAG's.

```
void setup()
{
    configure_mb_slave(COMM_BPS,
    PARITY, 0);

    pinMode(izql, OUTPUT);
    pinMode(derl, OUTPUT);
    pinMode(ledPin, OUTPUT);
}

void loop()
{
    if(update_mb_slave(MB_SLAVE, regs,
    MB_REGS))

        wdog = millis();

    if ((millis() - wdog) >
    5000) {

        regs[MB_CTRL] = 0;

    }

    i = i +1;
    if (i > 10000) i=0;
```

```

        regs[2] = i;
        switch (regs[MB_CTRL]) {
        case 0:
            digitalWrite(izq1,
LOW);
            digitalWrite(der1,
LOW);
            digitalWrite(ledPin, HIGH);
            break;
        case 1:
            digitalWrite(izq1,
LOW);
            digitalWrite(der1,
HIGH);
            break;
        case 2:
            digitalWrite(izq1,
HIGH);
            digitalWrite(der1,
LOW);
            break;
        case 3:
            digitalWrite(izq1,
LOW);
            digitalWrite(der1,
LOW);
            digitalWrite(ledPin, LOW);
            break;
        default:
            if (millis() -
tprev > regs[MB_TIME]) {
                if (LOW ==
digitalRead(izq1)&&LOW ==
digitalRead(der1)) {
                    digitalWrite(ledPin, HIGH);
                    regs[MB_CNT]++;
                } else {
                    digitalWrite(ledPin, LOW);
                }
                tprev =
millis();
            }
        }
    }
}

```

Algoritmo 2. Programación en C, realizada en Arduino, Configuración de funciones del Modbus RTU

```

unsigned int Txenpin = 0;
enum {
    FC_READ_REGS = 0x03,
    FC_WRITE_REG = 0x06,
    FC_WRITE_REGS = 0x10,
    FC_READ_DEV_ID = 0x2B
};
const unsigned char fsupported[] =
{ FC_READ_REGS, FC_WRITE_REG,
  FC_WRITE_REGS, FC_READ_DEV_ID };
enum {
    MAX_READ_REGS = 0x7D,
    MAX_WRITE_REGS = 0x7B,
    MAX_MESSAGE_LENGTH = 256
};
enum {
    RESPONSE_SIZE = 6,
    EXCEPTION_SIZE = 3,
    CHECKSUM_SIZE = 2
};
enum {
    NO_REPLY = -1,
    EXC_FUNC_CODE = 1,
    EXC_ADDR_RANGE = 2,
    EXC_REGS_QUANT = 3,
    EXC_EXECUTE = 4
};
enum {
    SLAVE = 0,
    FUNC,
    START_H,
    START_L,
    REGS_H,
    REGS_L,
    BYTE_CNT
};
unsigned int crc(unsigned char
*buf, unsigned char start,
unsigned char cnt)
{

```

```

        unsigned char i, j;
        unsigned temp, temp2, flag;

        temp = 0xFFFF;

        for (i = start; i < cnt;
i++) {
            temp = temp ^
buf[i];

            for (j = 1; j <= 8;
j++) {
                flag = temp
& 0x0001;

                temp = temp
>> 1;

                if (flag)

temp = temp ^ 0xA001;
            }}
            temp2 = temp >> 8;
            temp = (temp << 8) | temp2;
            temp &= 0xFFFF;
            return (temp);
        }

        void build_read_packet(unsigned
char slave, unsigned char function,
unsigned char count, unsigned char
*packet)
        {
            packet[SLAVE] = slave;
            packet[FUNC] = function;
            packet[2] = count * 2;
        }

        void build_write_packet(unsigned
char slave, unsigned char function,
unsigned int start_addr,
unsigned char count,
unsigned char *packet)
        {
            packet[SLAVE] = slave;
            packet[FUNC] = function;

            packet[START_H]
start_addr >> 8;

```

```

            packet[START_L]
start_addr & 0x00ff;

            packet[REGS_H] = 0x00;
            packet[REGS_L] = count;
        }

```

Entre los dos algoritmos se realizan las funciones necesarias para declaración de TAG's y la comunicación maestro - esclavo con un protocolo Modbus RTU, de decidió no poner el código completo de las dos partes tanto declaración como configuración de la comunicación por motivos de que es bastante amplia; por lo que esta transcrito lo principal de cada parte del programa.

OPC Server for Arduino/Genuino: Es un servidor OPC libre para Arduino. Un servidor OPC es un componente que permite comunicar cualquier software SCADA con cualquier dispositivo de manera abstracta (Arduino , 2020). A través de este software se configuro el puerto con el que se va a trabajar o con el que se va a comunicar Arduino con el computador, se lo hace de esta manera para obtener una mejor comunicación, en la Fig. 6 se muestra la configuración del puerto.

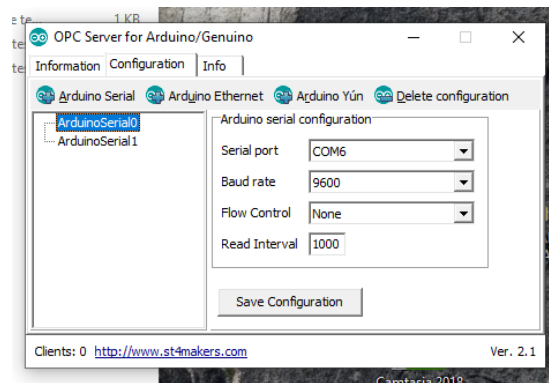


Fig. 6. Configuración del puerto a utilizar con el OPC Server for Arduino.

Monitoriza ACIMUT: Esta versión de "Acimut Monitoriza for Arduino" es

totalmente funcional y libre de todo tipo de restricciones de uso, tanto en cuanto número de variables a controlar como de clientes-puestos de monitorización que la versión comercial sí que tiene. La única limitación que tiene es que solo se puede conectar a dispositivos Arduino, en cambio la versión comercial permite la comunicación con cualquier tipo de tarjeta de adquisición de datos (Acimut S.L., 2010). En este software se desarrolló el HMI el cual cuenta con 4 pulsadores uno para inicio y fin del proceso, otro para mando a la derecha, otro para mando a la izquierda y un último para paro de emergencia, además, se cuenta con la simulación del movimiento del motor, leds indicadores y un indicador de estado, como se puede observar en la Fig. 7.

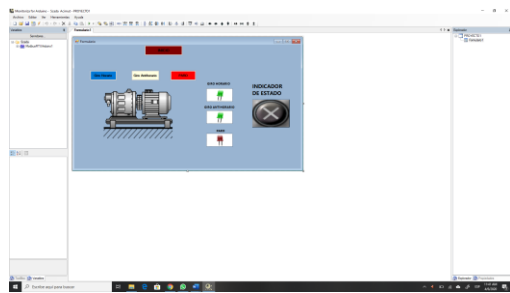


Fig. 7. Desarrollo de la aplicación en Monitoriza ACIMUT.

Como se puede observar se tiene un botón de inicio indicadores del tipo de giro o si esta en paro a continuación en la Fig. 8 se muestra el sistema en giro horario.

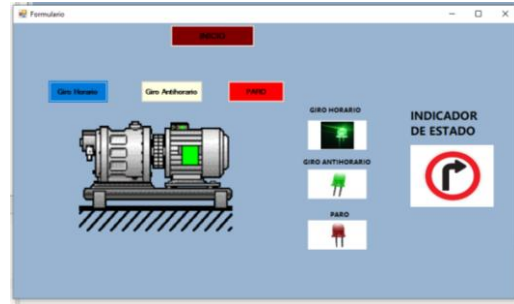


Fig. 8. Aplicación en funcionamiento giro sentido horario.

Una vez terminado las aplicaciones y realizado las pruebas de funcionamiento se procede a la comparación de parámetros, considerando de aquí en adelante como nombre de esta aplicación Arduino-HMI.

RESULTADOS

Según Wikacsonso y otros, 2016, las características más importantes que debe tener un sistema HMI o SCADA son las que se han listado en la tabla 1, las cuales se utilizaron para poder hacer una comparativa rápida que nos permita evidenciar cuál de las aplicaciones propuestas fueron más eficientes y con esto poder llevar a cabo el análisis final (Wikacsonso, Lim, & Sutanto, 2016).

Para poder realizar este análisis los investigadores se proponen como valores de ponderación 5 Cumple completamente, 4 Cumple, 3 cumple medianamente, 2 cumplimiento bajo, 1 casi no cumple, 0 no cumple, como se observa en la Tabla 1.

Tabla 1. Cuadro comparativo de las observaciones en base a los resultados obtenidos

CARACTERÍSTICAS		ARDUINO		-	PLC -HMI		OBSERVACIÓN
		HMI					
Tiempo de respuesta a eventos		Media	3		Alta	5	La respuesta de la aplicación con Arduino es más lenta que la de PLC
Maneja protocolo OPC		si	5		si	5	El protocolo OPC es el más usado actualmente en el ámbito industrial
Factibilidad de adquisición en el mercado		Si - Versión gratis	4		No - se necesita licencias	2	El sistema elaborado con Arduino es de software libre que además se creó en un software de versión gratuita industrial, en cambio para el sistema con PLC se necesita licencias de los programas tanto para la programación como para el HM
Servidores a tiempo real		Medio	3		Alta	5	El sistema elaborado con Arduino necesita módulos extras para el manejo adecuado de funciones en tiempo real
Visualización gráfica dinámica		Media	3		Alta	5	El sistema elaborado con Arduino tiene un tiempo de reacción mayor con respecto al de PLC, por lo que el dinamismo se reduce.
Programación de eventos		Si	5		Si	5	
Procesamiento de alarmas		Si	5		Si	5	
Robustez		Media	3		Alta	5	El sistema elaborado con Arduino no tiene la misma robustez del elaborado con PLC debido a que la funcionalidad del primero es educativa.
Permite librerías externas		Si	5		No	0	El sistema elaborado con Arduino permite uso de librerías de terceros como es .NET framework de Windows
TOTAL		36			37		

En base al análisis realizado en el cuadro comparativo y valorativo de las características de cada uno de los sistemas elaborados y a pesar de obtener un buen puntaje con una mínima diferencia de un punto de la aplicación PLC – HMI, sobre Arduino – HMI, se observa que una de las características principales que se necesita tener en un proceso industrial real y de alto nivel es la robustez, que es la características en la cual es proceso realizado con Arduino más fallas tiene, por lo que se concluye que el sistema se puede elaborar con Arduino para fines educativos más no

industriales y el realizado con PLC puede ser usado en los dos ámbitos establecidos como son industrial y educativo, teniendo así que un sistema de entrenamiento puede ser realizado en base de Arduino para amenorar costos y así enseñar sistemas Scada a los estudiantes en instituciones donde no se tienen los recursos o laboratorios necesarios. Además, Arduino al ser una tarjeta educativa no soporta demasiadas entradas y salidas ya que empieza a fallar de manera aleatoria, en cambio el PLC cuenta con módulos que permiten aumentar las entradas y salidas y siguen funcionando adecuadamente, como se observa en la Fig. 9.

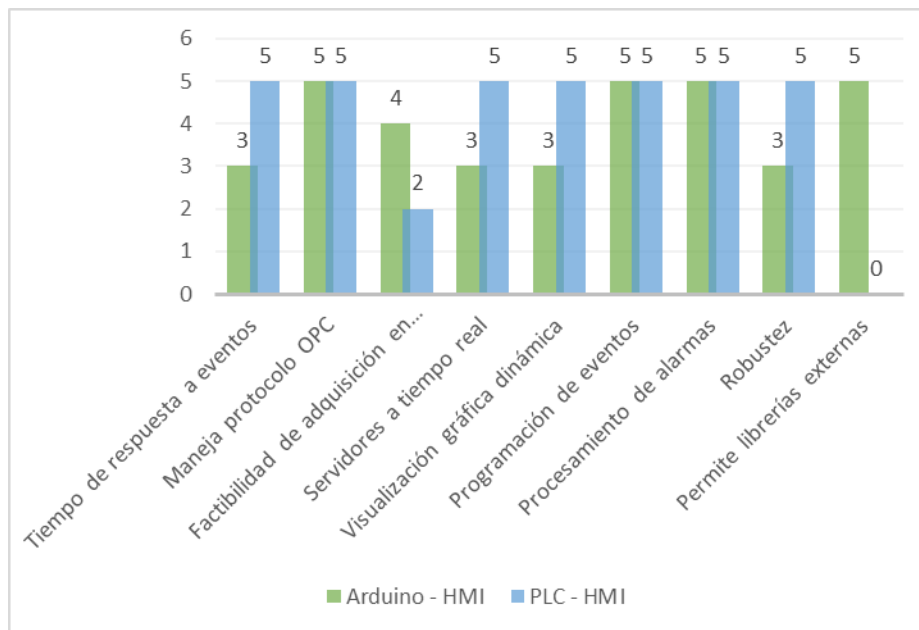


Fig. 9. Grafica de barras de la comparación entre las aplicaciones.

CONCLUSIONES Y TRABAJOS FUTUROS

El desarrollo de esta investigación a permitido entender que Arduino es una herramienta fácil de usar y accesible, pero que actualmente no se lo esta explotando a su máximo nivel, ya que si se hace una

investigación más exhaustiva de esta plataforma gratuita se puede observar variedad de proyectos que se están a alta escala es decir a nivel industrial y en todo tipo de industria.

En cuanto a las encuestas realizadas a los jefes de taller de las empresas con las que se tienen convenios, los resultados han dado a entender que si estarían dispuestos

a que se desarrollen aplicaciones para ensayos o pruebas dentro de las industrias donde ellos laboran con el fin de demostrar su aplicabilidad y a futuro implementar sistemas de ensayo a baja escala de sistemas reales, con el fin de capacitar al personal.

En el desarrollo de las aplicaciones en cuanto a complejidad las dos aplicaciones tienen su nivel de complejidad por lo que sin importar que sistema se esté usando en la enseñanza de aplicaciones industriales, se van a obtener buenos resultados y el estudiante será capaz de entender la lógica de funcionamiento de los sistemas industriales, para cuando lleguen al campo laboral se les haga más fácil entender los procesos, su manejo y su mantenimiento.

Como se pudo observar en la comparación de las aplicaciones la aplicación realizada con PLC es mejor en cuanto a robustez, pero tomando en cuenta la pequeña diferencia que existió entre los totales, se puede decir que Arduino si es una herramienta que se puede usar en la educación superior tecnológica, pudiéndola aplicar a los primeros niveles de estudio y darle un plus al finalizar los niveles usando algo mas avanzando como un PLC.

En conclusión, el resultado de esta investigación indica que Arduino y PLC son aplicables en la educación superior tecnológica y en la industria; considerando que se debe hacer una investigación más a fondo donde se ponga a prueba dos paralelos de un mismo nivel de una carrera tecnológica, en donde en uno de los niveles se enseñe sistemas SCADA o comunicación industrial haciendo uso del proceso Arduino – HMI y en el otro curso la misma materia con el proceso de PLC – HMI.

REFERENCIAS

- Acimut S.L. (2010). <http://www.acimut.com/>. (SCADA Acimut Monitoriza) Recuperado el 03 de 2020, de <http://www.acimut.com/monitoriza/monitorizaforarduino.html>
- Aguirre, F., & García, B. (2017). Proyectos ARDUINO con estrategias de enseñanza soportadas en blended learning. *EduQ@2017*. San Luis.
- Arduino . (2020). *arduino.cc*. Recuperado el 16 de 03 de 2020, de <https://www.arduino.cc/en/main/software#>
- Barrera, D., Mendoza, J., Neira, L., & Pardo, A. (2019). Enlace entre PLC Siemens LOGO! 12/24 RCE y Wonderware InTouch para supervisión y control de nivel. En EasyChair (Ed.), *17th LACCEI International Multi-Conference for Engineering, Education, and Technology*. Jamaica. doi:to be inserted by LACCEI
- Romo, R., Rogelio, M., & Martí, R. (2015). Plantas virtuales basadas en Arduino para el Laboratorio Docente de Informática Industrial. *Actas de las XXXVI Jornadas de Automática* (págs. 696 -703). Bilbao: Comité Español de Automática de la IFAC (CEA-IFAC). doi:978-84-15914-12-9
- Sujith, J., & Hemalatha, B. (April de 2014). Fault Identification and Protection of Induction Motor using PLC and SCADA.

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 3(4), 8748 - 8754.

Wikacsono, H., Lim, R., & Sutanto, W. (2016). Perancangan SCADA Software dengan Wonderware InTouch Recipe Manager dan SQL Access Manager pada Simulator Proses Pencampuran Bahan. *Jurnal Teknik Elektro*, 8(1), 38 - 45.