

Jornadas de Automática

ArduTC – Un entrenador de bajo coste para TwinCAT

Moreno, F.-A. *, Torres, V.*

*Dpto. de Ingeniería de Sistemas y Automática. Instituto Universitario en Ingeniería Mecatrónica y Sistemas Ciberfísicos (IMECH.UMA).
Universidad de Málaga, Blvr. Louis Pasteur, 35, 29071 Málaga, España.*

To cite this article: Moreno, F.-A., Torres, V. 2024. ArduTC – A low-cost trainer for TwinCAT. *Jornadas de Automática*, 45. <https://doi.org/10.17979/ja-cea.2024.45.10951>

Resumen

La realización de prácticas docentes con PLC en la ingeniería supone un desafío en los centros educativos debido al alto coste, el espacio ocupado y el mantenimiento de los equipos necesarios para llevarlas a cabo. Esto avoca al alumnado a tomar turnos para realizar ejercicios de automatización o, si esto no es posible, al uso de simuladores, reduciendo considerablemente la experiencia práctica. ArduTC nace como un intento de democratización del *hardware* para automatización, desarrollando un sistema de muy bajo coste que convierte una placa de desarrollo de Arduino UNO en una interfaz entre TwinCAT, el *software* comercial de programación de PLC de Beckhoff, y elementos básicos de automatización como LEDs, pulsadores, sensores analógicos y servomotores, sustituyendo en última instancia a un PLC comercial. Nuestro sistema se comunica de manera transparente con un programa TwinCAT que se ejecuta en un PC mediante el protocolo ADS aplicando los cambios en las variables del programa a los pines analógicos y digitales de una placa Arduino y viceversa.

Palabras clave: Controladores lógicos programables, Cuestiones de equilibrio entre formación teórica y práctica, Clases, departamentos, laboratorios y escuelas virtuales, TwinCAT, Arduino

ArduTC – A low-cost trainer for TwinCAT

Abstract

The realization of teaching practices with PLCs in engineering is a challenge in educational centers due to the high cost, the space occupied and the maintenance of the equipment necessary to conduct them. This leads students to take turns to perform automation exercises or, if this is not possible, to the use of simulators, reducing considerably the practical experience. ArduTC was born as an attempt to democratize automation hardware, developing a very low cost system that converts an Arduino UNO development board into an interface between TwinCAT, the commercial PLC programming software from Beckhoff, and basic automation elements such as LEDs, push buttons, analog sensors and servo motors, ultimately replacing a commercial PLC. Our system communicates transparently with a TwinCAT program running on a PC using the ADS protocol applying changes in program variables to the analog and digital terminals on the Arduino board and vice versa.

Keywords: Programmable logic controllers, Balance issues of theoretical-versus-practical training, Virtual classes, departments, laboratories and schools, TwinCAT, Arduino

1. Introducción

Los autómatas programables industriales (*Programmable Logic Controllers*, PLC) son dispositivos electrónicos programables diseñados específicamente para actuar como controlador de un sistema automático de producción. Los PLC son

uno de los dispositivos de control más usados en la industria lo que justifica que la programación de PLC sea una materia comúnmente impartida en la mayoría de estudios de carácter tecnológico en los diversos niveles educativos desde ciclos de secundaria o módulos de formación profesional y bachillera-

*Igual contribución. Autor para correspondencia: famoreno@uma.es
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

to hasta grados universitarios. Tradicionalmente, para impartir las prácticas de esta materia, los centros educativos suelen disponer, en el mejor de los casos, desde maquinaria industrial o equipos didácticos de automatización con PLC valorados en decenas de miles de euros hasta simples entrenadores con PLC conectados a montajes didácticos cuyo precio puede oscilar entre unos cientos a algo menos de dos mil euros.

Probablemente la mejor opción para aprender a programar un PLC es hacerlo con una máquina industrial real como puede ser el caso del «sistema didáctico modular de ensamblaje flexible FMS-200» de SMC International Training (SMC International Training, 2014). Pero, en primer lugar, es una opción de coste elevado, lo que puede ser un obstáculo para determinados centros educativos y, además, requiere de un espacio y un mantenimiento adecuado. Por otro lado, dada la envergadura y tamaño de estos equipos, no es especialmente adecuado para que los alumnos realicen prácticas de iniciación de manera individual por lo que, habitualmente, los alumnos deben turnarse para hacer uso de estos equipos, limitando el tiempo de sus prácticas.

Las alternativas a estos equipos grandes y costosos pasan por el uso de entrenadores: un PLC con un pequeño conjunto de entradas y salidas dispuestos en un determinado soporte que permiten realizar prácticas sencillas de automatización. El problema principal de estos equipos radica en que, a pesar de su sencillez, su coste de adquisición aún puede ser elevado, dificultando que pueda ser comprado en cantidades suficientes como para satisfacer las necesidades docentes y, por supuesto, por parte de los alumnos que quieran practicar en casa.

Una tercera vía va de la mano del uso de simuladores, los cuales permiten trabajar a muchos alumnos simultáneamente pero no proporcionan la experiencia de uso de dispositivos reales. Los avances actuales en la mejora del realismo de los simuladores, orientándose hacia el desarrollo de verdaderos gemelos digitales (Grieves, 2014), buscan reducir esa brecha en la experiencia, pero aún queda camino por recorrer. En cualquier caso, en un área con tan alto grado de experimentalidad como la automatización industrial, separar el aprendizaje del alumno del manejo de algún tipo de *hardware*, puede, de alguna manera, empobrecer el proceso (Shaaban, 2013).

En este contexto, nuestra propuesta, ArduTC, pretende acercar la práctica de la programación de PLC al estudiante de manera que con un desembolso mínimo pueda acceder a la realización de prácticas de automatización utilizando un *software* industrial comercial sobre un *hardware* doméstico. Nuestro sistema está basado en la placa de desarrollo Arduino UNO, que actúa como terminal de entradas y salidas analógicas y digitales que pueden ser gestionadas desde una tarea de PLC de TwinCAT 3 (Beckhoff Automation, 1996), el *software* de configuración y programación de los equipos de la marca Beckhoff. Una implementación anterior de ArduTC, compatible con la versión 2 de TwinCAT, fue desarrollada como parte de un TFG (Huertas-López, 2021) en la Universidad de Málaga, si bien el programa presentado aquí ha sido rehecho desde cero incorporando nuevas librerías de comunicaciones, creando una nueva interfaz y haciéndolo compatible con la última versión de TwinCAT. Este nuevo proyecto puede ser accedido en <https://github.com/vetorres-uma/ArduTC>.

ArduTC no pretende ser una solución que sustituya a las demás sino que tiene la intención de complementarlas permi-

tiendo el acceso a la realización de prácticas a cualquier bolsillo sin renunciar a la extraordinaria motivación que supone para el aprendiz «hacer que las cosas se muevan».

2. Estado del arte

En el mundo de la automatización industrial se denomina entrenador a un PLC con un conjunto de entradas y salidas dispuesto en un determinado soporte. Los entrenadores permiten que los ingenieros de automatización puedan, en su propia oficina, taller o laboratorio, desarrollar y probar, hasta cierto punto, la lógica de control de un proyecto de automatización antes de instalarlo por primera vez en casa del cliente. Los entrenadores, artesanales o comerciales (ver Fig. 1), son también ampliamente utilizados como equipo de prácticas en cursos de automatización impartidos en centros docentes de todo tipo.

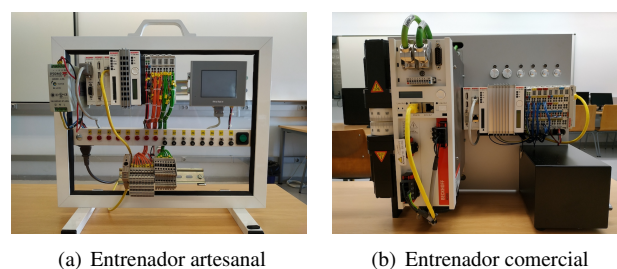


Figura 1: Entrenadores con controladores Beckhoff.

Un entrenador básico con un PC embebido Beckhoff puede tener un precio aproximado de entre 800 € y 3000 €. Al ser sólo una parte de una instalación industrial real (básicamente el controlador) el precio en comparación con un sistema automatizado completo se reduce significativamente. Además, ocupa poco espacio y puede completarse con dispositivos de entrada y salida realizando diferentes montajes, lo que lo convierte en una opción muy atractiva como equipo de prácticas para muchos centros educativos.

Recientemente ha tomado impulso la tecnología de los gemelos digitales que permiten la interacción de un PLC *hardware* o *software* con maquinaria industrial virtual a través de aplicaciones comerciales como Factory I/O (Factory I/O, 2014), EasyPLC Software Suite (Nirtec Studio, 2024), CoppeliaSim (Coppelia Robotics, 2013) o Visual Components (Visual Components, 2016). Alternativamente a estos sistemas comerciales, está creciendo el desarrollo de gemelos digitales a medida utilizando motores de videojuegos (ver Fig. 2), destacando las soluciones basadas en Unity3D (Unity Technologies, 2005) como en (Wang et al., 2021) o Godot (Godot Foundation, 2014), de código abierto, usado en (Badawy et al., 2023). En cualquier caso, todos los sistemas basados en dispositivos virtuales no proporcionan al estudiante la experiencia completa a la hora de practicar con un sistema PLC.

3. El ecosistema de ArduTC

Para su implementación ArduTC requiere de un sistema de PLC suficientemente abierto que disponga de un emulador o simulador de PLC. TwinCAT, el sistema de programación de la compañía Beckhoff, es un excelente candidato pues además

es en gran medida compatible con la norma IEC 61131-3 (International Electrotechnical Commission, 2013) y dispone de una política de licencias conveniente para el uso educativo.

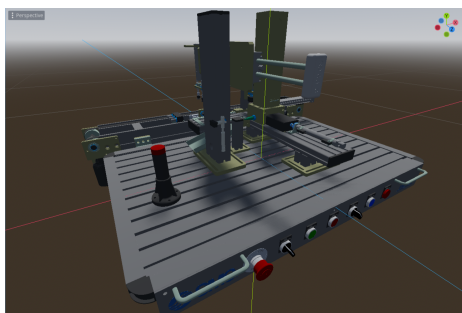


Figura 2: Ejemplo de gemelo digital con Godot.

3.1. Beckhoff

Beckhoff Automation GmbH & Co. KG es un fabricante de tecnología de automatización con sede en Verl en East Westphalia y parte del Grupo Beckhoff que implementa sistemas de automatización para la industria y la automatización de edificios. La característica esencial y diferenciadora de sus sistemas es el uso de la tecnología de control basada en PC que se desarrolla sobre la base de estándares abiertos. Sus terminales electrónicos, el *software* de automatización TwinCAT y el protocolo abierto de comunicación industrial EtherCAT son solo algunos ejemplos de los productos y tecnologías desarrollados por Beckhoff Automation que han alcanzado importancia en el mercado mundial de la automatización.

La mayor parte de las aplicaciones para la programación de PLC industriales son *software* privativo licenciado con precios nada despreciables, aunque algunas disponen de versiones de prueba con limitaciones de uso en cuanto a la funcionalidad o la caducidad temporal. Otras, en cambio, como es el caso de TwinCAT, a pesar de ser privativas disponen de un sistema de licencias que permiten un uso ilimitado en tiempo y funcionalidad para los desarrolladores teniendo únicamente que abonar el importe de las licencias utilizadas cuando el sistema se pone en explotación. Esta filosofía hace de TwinCAT una aplicación muy adecuada para ser utilizada con fines docentes.

3.2. TwinCAT

TwinCAT (*The Windows Control and Automation Technology*), el sistema de programación de Beckhoff, es un completo paquete para la automatización que incluye un *software* de desarrollo de aplicaciones para PLC y control numérico y un *Runtime* que permite ejecutar en tiempo real código TwinCAT en cualquier ordenador con Windows. TwinCAT 2 y TwinCAT 3 son las dos versiones que actualmente están disponibles de TwinCAT. En TwinCAT 2 el entorno de desarrollo está formado por un conjunto de aplicaciones entre las que destacan *PLC Control* y *System Manager*. En TwinCAT 3, por contra, todas las funcionalidades TwinCAT necesarias para el desarrollo de aplicaciones están integradas en el entorno de programación Microsoft Visual Studio.

TwinCAT XAE de TwinCAT 3 es un completo entorno integrado de desarrollo (IDE) de proyectos para PLC que utiliza

el conjunto de lenguajes de la norma IEC 61131-3 y que dispone de herramientas tanto para la edición y prueba de código de control como la configuración del sistema (dispositivos y terminales) y conexión (vinculación) de las variables de entradas y de salidas de las tareas *software* con las señales provenientes de los terminales de entradas y de salidas conectados al controlador a través de los buses de campo.

TwinCAT XAR de TwinCAT 3, también denominado *Runtime*, es una aplicación informática (*SoftPLC*) que emula el comportamiento de un PLC en un ordenador con Windows. Todos los PC embebidos de la marca Beckhoff pueden suministrarse con un *Runtime*, convirtiéndolos en verdaderos PLC capaces de ejecutar código IEC 61131-3, pero también puede ser instalado en un PC convencional (habitualmente junto al XAE) para ejecutar código TwinCAT, adoptando el nombre de *Runtime* local.

El protocolo de *software* libre ADS (Automation Device Specification) de Bekhoff es la capa de transporte utilizada internamente por TwinCAT para intercambiar información entre sus diferentes módulos de *software* (PLC, NC,...). ADS utiliza el protocolo de red TCP/IP y proporciona también una serie de métodos que permiten el intercambio de información entre TwinCAT y otros ordenadores o dispositivos externos. Beckhoff proporciona librerías para utilizar el protocolo ADS en diferentes lenguajes de programación.

Para nuestro propósito, TwinCAT presenta una serie de ventajas fundamentales:

1. Es un excelente *software* comercial que cumple en un alto grado la norma internacional de programación de autómatas programables IEC 61131-3.
2. Dispone de un emulador (*Runtime*) capaz de ejecutar tareas escritas en cualquiera de los lenguajes de la norma IEC 61131-3 en casi cualquier PC con Windows.
3. Es un sistema abierto que permite acceder a sus componentes internos a través del protocolo ADS.
4. A pesar de ser un *software* privativo puede usarse de forma gratuita, indefinida y con toda su funcionalidad a excepción del *Runtime* que dispone de una licencia gratuita limitada temporalmente pero fácilmente renovable de forma indefinida.

3.3. Norma IEC 61131-3

La IEC 61131-3 es la tercera parte de la norma IEC 61131 para controladores lógicos programables que describe un conjunto de cinco lenguajes de programación de PLC: dos textuales (lista de instrucciones y texto estructurado) y tres gráficos (diagramas de contactos, diagramas de bloques funcionales y diagramas funcionales en secuencia).

4. Descripción del sistema ArduTC

ArduTC dispone de dos funcionalidades principales: (i) permitir, de manera similar a TwinCAT, enlazar las variables de entrada y salida analógicas y digitales de una tarea TwinCAT con los pines analógicos y digitales de una placa microcontroladora Arduino UNO; y (ii) servir de interfaz entre el *Runtime* de TwinCAT y la placa microcontroladora para que una tarea IEC 61131-3 de TwinCAT gestione de forma efectiva y transparente las entradas y salidas de la placa Arduino

UNO. De esta forma, ArduTC actúa de intermediario entre las entradas y salidas gestionadas por el sistema Arduino y el programa de control del sistema, que se ejecuta en el *Runtime* (ver Fig. 3). Para cumplir con su finalidad, ArduTC se sustenta en librerías de *software* libre escritas en el lenguaje Python, tanto para la gestión de las comunicaciones con TwinCAT y con la placa de Arduino, como para realizar la interfaz gráfica.

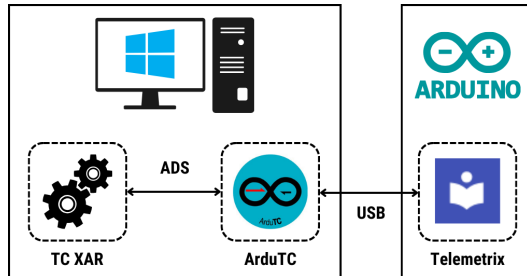


Figura 3: Esquema de comunicación de ArduTC.

Hay que destacar, por supuesto, que, aunque la transmisión de datos mediante ADS entre el *Runtime* de TwinCAT y ArduTC se produzca en tiempo real, la latencia que introduce la conexión USB con el sistema Arduino no garantiza que tal requisito se cumpla. Este sistema de comunicación no puede sustituir, por tanto, al sistema basado en EtherCAT u otro protocolo de bus con los que son compatibles los dispositivos físicos de Beckhoff. Sin embargo, la velocidad de respuesta es suficientemente alta como para poder ser utilizado como equipo de aprendizaje.

4.1. Hardware

ArduTC se sustenta sobre una placa Arduino UNO (Arduino, 2010) conectada mediante USB a un PC con TwinCAT. Arduino UNO es una placa de desarrollo de micro-controlador basada en el microchip ATmega328P desarrollada por Arduino.cc. La placa dispone de un conjunto de pines de entrada y salida analógicas y digitales fácilmente accesibles para facilitar su conexión a una gran variedad de placas de expansión (*shields*) y otros circuitos. El diseño del *hardware* de Arduino UNO se distribuye bajo una licencia Creative Commons. El proyecto Arduino UNO ha contribuido de forma decisiva a la democratización del *hardware* dando un enorme impulso al desarrollo de la comunidad *maker*. Siendo quizás la más popular de las placas micro-controladoras que existen actualmente en el mercado por su versatilidad y por su precio y de las más utilizadas en los proyectos de electrónica desarrollados por aficionados. El precio de una placa Arduino UNO está entre los 3-10 € de un clon y los 20 € de una placa original.

Por todo esto queda, a nuestro entender, plenamente justificado desarrollar una aplicación como ArduTC que pretende universalizar el acceso a las prácticas de programación de PLC para el entorno de desarrollo de *software* industrial comercial TwinCAT y la placa micro-controladora Arduino UNO.

4.2. Software

Para cumplir con su finalidad, ArduTC utiliza tres librerías de *software* libre escritas para el lenguaje Python: (i) *pyads* (Stefan Lehmann, 2015), que implementa el protocolo ADS para la comunicación con TwinCAT, (ii) *Telemetrix* (Alan

Yorinks, 2020), que permite la comunicación con la tarjeta micro-controladora, y (iii) *PyQT5* (Riverbank Computing Limited, 1998) para el desarrollo de la interfaz gráfica de usuario.

4.2.1. *pyads*

El protocolo ADS (*Automation Device Specification*) es la capa de transporte utilizada internamente entre los distintos elementos de TwinCAT y que permite el intercambio de información entre los diferentes módulos de *software* (PLC, NC, ...). El protocolo ADS utiliza el protocolo de red TCP/IP y proporciona también una serie de métodos que permiten el intercambio de información entre TwinCAT y otros ordenadores o dispositivos externos.

Beckhoff no proporciona una librería Python para el protocolo ADS (sí lo hace para otros lenguajes), por lo que haremos uso de la librería *pyads*. Esta librería, en realidad es un contenedor de Python para la biblioteca ADS de TwinCAT, proporcionando una forma *pitónica* de comunicarse con dispositivos TwinCAT a través de la API C (*TcAdsDll.dll*) para el protocolo ADS, proporcionada por Beckhoff para Windows.

4.2.2. *Telemetrix*

Telemetrix es una alternativa moderna al clásico protocolo de comunicación para microcontroladores Firmata (Steiner, 2009) que permite una comunicación eficiente y bidireccional entre un PC y varias familias de tarjetas de desarrollo basadas en microcontrolador (Arduino, STM32 y ESP-8266). Consiste en una API de Python que permite crear una aplicación cliente y dos tipos de aplicación servidor desarrolladas en C++ para comunicarse con el cliente a través de una comunicación serie o wifi. Con esta librería, podemos enviar y recibir datos en tiempo real, con baja latencia y eficiencia en el uso de recursos. Su documentación completa está disponible en GitHub y su sitio web oficial. Para hacer uso de *Telemetrix* hay que descargar la aplicación servidor en la placa micro-controladora.

4.2.3. *pyQT5*

La interfaz de la aplicación se ha desarrollado usando QT-Creator y la librería *PyQT5*, una adaptación de la biblioteca gráfica Qt para el lenguaje de programación Python. En la Fig. 4(a) puede observarse la interfaz diseñada con dos partes diferenciadas: (i) en la zona izquierda se muestran las variables de entrada y salida correspondientes a una tarea TwinCAT y (ii) a la derecha se presentan los pines de la placa Arduino UNO. Esta interfaz permite realizar las siguientes acciones (los elementos de la interfaz asociados con cada acción se han marcado con su número correspondiente en la Fig. 4):

1. Conectarse con el sistema Arduino y leer los pines de E/S disponibles.
2. Leer desde un fichero (.tmc) las variables de E/S de una tarea TwinCAT.
3. Configurar y realizar la conexión entre una tarea que procese el *Runtime* de TwinCAT y la placa Arduino.
4. Realizar la asociación entre las variables de una tarea TwinCAT y los pines de E/S de Arduino.
5. Visualizar “en tiempo real” los valores actuales de las variables TwinCAT y los pines de la placa Arduino.

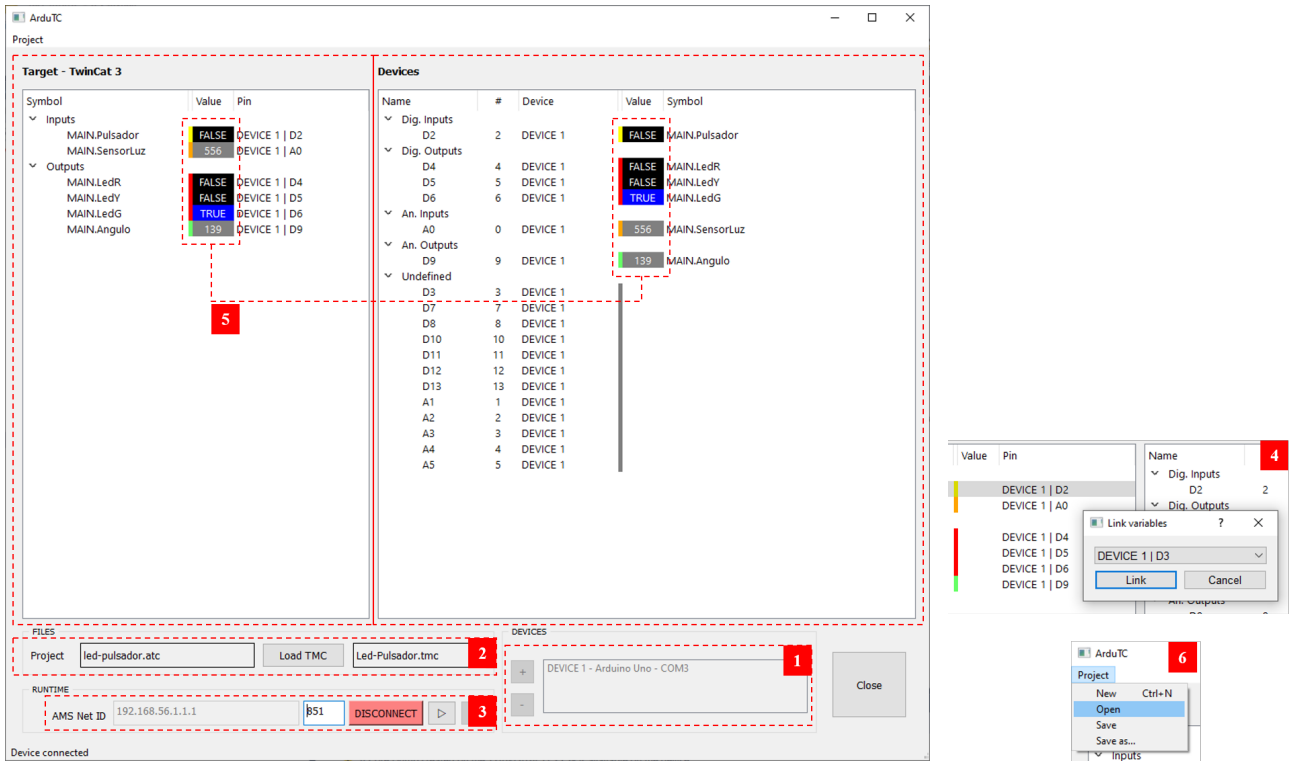


Figura 4: Interfaz de usuario durante la prueba de concepto.

6. Gestionar el proyecto de ArduTC (crear uno nuevo, guardar el actual o cargar uno guardado).

4.3. Funcionamiento

El procedimiento normal para hacer funcionar un proyecto TwinCAT sobre un hardware real Beckhoff incluye las siguientes fases: (i) implementación de la tarea a ejecutar, (ii) compilación del proyecto, (iii) vinculación de las variables de E/S con los canales de los terminales del hardware a utilizar, (iv) activación de la configuración de vinculación en el controlador destino, (v) descarga del proyecto en el Runtime y (vi) puesta en marcha del proyecto. Si se dispone de un controlador (PLC) real, las etapas (iii-vi) se realizan sobre él.

La introducción de ArduTC en sustitución del hardware real, modifica este procedimiento de la siguiente forma. La programación y la compilación se realizan de la misma manera. Una vez compilado el proyecto, TwinCAT genera un fichero .tmc con la información de las variables que son susceptibles de ser vinculadas con terminales de E/S. ArduTC lee este fichero y registra las variables para obtener notificaciones cuando cambien durante la ejecución del proyecto en el Runtime local. Posteriormente, nos conectaremos mediante ArduTC con el dispositivo Arduino y obtendremos la información de los pines de E/S disponibles en la placa. Ahora procederemos a la vinculación entre dichos pines y las variables obtenidas de TwinCAT. Finalmente, las etapas (iv-vi) del procedimiento normal se realizan de la misma forma pero en esta ocasión, en vez de realizarlas sobre un Runtime que reside en un controlador real, lo haremos sobre el Runtime local que reside en el mismo PC. Las diferencias en las etapas a realizar al usar (o no) ArduTC se pueden ver en la Fig. 5.

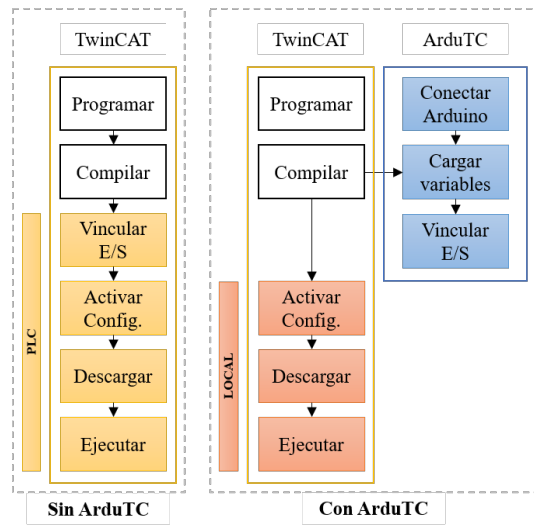


Figura 5: Procedimiento de puesta en marcha de un proyecto TwinCAT sin y con ArduTC.

Una vez el proyecto TwinCAT está en ejecución, ArduTC trabaja de la siguiente forma. Cuando recibe una notificación del Runtime de TwinCAT (a través del protocolo ADS) de que una determinada variable de salida de la aplicación ha cambiado actualiza su valor en la placa micro-controladora mediante las funciones de Telemetrix. De manera análoga, cuando recibe una notificación a través de Telemetrix de que una determinada señal de entrada ha cambiado de valor en la placa micro-controladora, la actualiza en el proyecto TwinCAT mediante el protocolo ADS. Este intercambio de información se realiza de manera transparente para el usuario y, si bien no se

alcanzan los requisitos de tiempo real, el funcionamiento del sistema es idéntico en la práctica al que se obtendría al trabajar con TwinCAT sobre un *hardware* Beckhoff. De esta forma, cuando se está realizando la vinculación de las variables de entrada y salida de la tarea TwinCAT con los pines de la tarjeta Arduino UNO mediante ArduTC, el interfaz y el flujo de trabajo son análogos a los que se utilizan con el propio TwinCAT para la vinculación de estas variables con los canales de los terminales de entrada y salida Beckhoff. Además, en modo de funcionamiento (*online/offline*), al ser totalmente transparente el servicio de intermediación que realiza ArduTC entre TwinCAT y la tarjeta Arduino UNO, el usuario no aprecia diferencia alguna con respecto al uso ordinario de TwinCAT con los terminales de entrada y salida comerciales.

5. Caso de uso

A modo de caso de uso, se ha desarrollado una aplicación para el control de un semáforo que consta de tres LED de colores, un pulsador, un sensor de luz y un pequeño servomotor (ver Fig. 6). El ejemplo mostrado simplemente pretende ilustrar la capacidad de ArduTC de lidiar con señales de entrada y salida tanto analógicas como digitales.

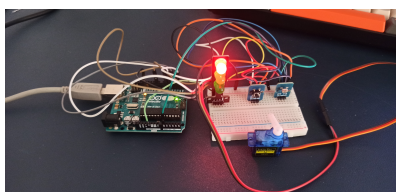


Figura 6: Montaje kit Arduino.

Un pequeño programa en TwinCAT se encarga de realizar una secuencia de luces con el semáforo, con un comportamiento modificable con el pulsador, mientras que se captura el valor analógico proporcionado por el sensor de luz y se utiliza para establecer la posición del microservo escribiendo en la salida analógica correspondiente. En la Fig. 4(a) se muestra la aplicación ArduTC funcionando con el programa TwinCAT descrito, donde se aprecian los valores actuales de las variables y los terminales de la placa Arduino vinculados a ellas.

6. Conclusiones y trabajos futuros

En este artículo hemos presentado ArduTC, un dispositivo de bajo coste basado en Arduino, capaz de hacer de intermediario entre TwinCAT, el *software* de programación de Beckhoff, y un conjunto de elementos *hardware* de manera transparente para el usuario. La placa Arduino hace las labores de cabecera de terminal de entrada/salida y la aplicación desarrollada comunica dicha placa con un *Runtime* local que proporciona TwinCAT y que se ejecuta en el mismo PC.

Con ArduTC, realizando un desembolso reducido para la adquisición de una placa micro-controladora tipo Arduino UNO y algún *hardware* adicional (LED, resistencias, pulsadores, relés, etc.), cualquier alumno de tecnología o ingeniería de cualquier nivel formativo puede realizar prácticas de automatización con PLC utilizando el *software* profesional de programación TwinCAT «haciendo que las cosas se

muevan». ArduTC ha sido puesto a disposición pública en <https://github.com/vetorres-uma/ArduTC>.

Como trabajos futuros podemos destacar: (i) extender el uso de ArduTC a otras plataformas de micro-controlador Arduino (MEGA, Nano, Due, etc.), ESP32, ESP8266 y posibilitar que trabaje simultáneamente con más de una placa micro-controladora; (ii) estudiar la extensión del funcionamiento de ArduTC con otros entornos de programación de PLC (TIA Portal, CX-Programmer, etc.); (iii) realizar un conjunto de montajes típicos para la realización de prácticas de PLC.

Agradecimientos

Este trabajo ha sido desarrollado en el contexto del Proyecto de Innovación Educativo "Mejora del aprendizaje mediante una planta industrial de entrenamiento realista y de bajo coste para la docencia en Industria Inteligente" (PIE101) promovido por la Universidad de Málaga en su programa INNOVA22.

Referencias

- Alan Yorinks, 2020. Telemetrix, repositorio Github. <https://github.com/MrYsLab/telemetrix> [Accedido: mayo de 2024].
- Arduino, 2010. Arduino UNO, web oficial. <https://store.arduino.cc/products/arduino-uno-rev3> [Accedido: mayo de 2024].
- Badawy, I., Bassiuny, A., Darwish, R., Tolba, A., 2023. A digital twin of a remote real-time accessible labs. In: International Conference on Interactive Collaborative Learning. Springer, pp. 200–212.
DOI: https://doi.org/10.1007/978-3-031-52667-1_21
- Beckhoff Automation, 1996. TwinCAT automation software, web oficial. <https://www.beckhoff.com/es-es/products/automation/twincat/> [Accedido: mayo de 2024].
- Coppelia Robotics, 2013. Coppeliassim, web oficial. <https://www.coppeliarobotics.com/> [Accedido: mayo de 2024].
- Factory I/O, 2014. Factory I/O, web oficial. <https://factoryio.com/> [Accedido: mayo de 2024].
- Godot Foundation, 2014. Godot engine, web oficial. <https://godotengine.org/> [Accedido: mayo de 2024].
- Grieves, M., 2014. Digital twin: manufacturing excellence through virtual factory replication. White paper 1 (2014), 1–7.
- Huertas-López, A. J., Junio 2021. Diseño y desarrollo de un dispositivo de entrada y salida de bajo coste para TwinCAT. Trabajo Fin de Grado. Grado en Ingeniería en Tecnologías Industriales, Universidad de Málaga.
- International Electrotechnical Commission, 2013. IEC 61131-3: 2013 programmable controllers-Part 3: programming languages. Geneva, Switzerland.
- Nirtec Studio, 2024. Easyplc software, web oficial. <https://www.nirtec.com/> [Accedido: mayo de 2024].
- Riverbank Computing Limited, 1998. pyqt5, web oficial. <https://www.riverbankcomputing.com/software/pyqt/> [Accedido: mayo de 2024].
- Shaaban, K., 2013. Practical teaching and its importance in teaching civil engineering. QScience Proceedings 2013 (2), 4.
DOI: <https://doi.org/10.5339/qproc.2013.gic.4>
- SMC International Training, 2014. Fms-200 - sistema didáctico modular de ensamblaje flexible, web oficial. <https://www.smctraining.com/es/webpage/indexpage/287> [Accedido: mayo de 2024].
- Stefan Lehmann, 2015. pyads, repositorio Github. <https://github.com/stlehmann/pyads> [Accedido: mayo de 2024].
- Steiner, H.-C., 2009. Firmata: Towards making microcontrollers act like extensions of the computer. In: NIME. pp. 125–130.
DOI: <https://doi.org/10.5281/zenodo.1177689>
- Unity Technologies, 2005. Unity3D, web oficial. <https://unity.com/> [Accedido: mayo de 2024].
- Visual Components, 2016. Visual components, web oficial. <https://www.visualcomponents.com/> [Accedido: mayo de 2024].
- Wang, Z., Han, K., Tiwari, P., 2021. Digital Twin Simulation of Connected and Automated Vehicles with the Unity Game Engine. In: 2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DT-PI), pp. 1–4.
DOI: <https://doi.org/10.1109/DTPI52967.2021.9540074>