

在智慧製造上的 Low-Code SCADA 系統建置

鄭琮生

南臺科技大學電子工程系

tscheng@stust.edu.tw

摘要

全球製造業近年進入了第四次的工業革命，促使智慧製造觀念的產生，亦即透過生產數據的蒐集並搭配 AI 進行生產分析與提升產線之稼動率。另外，2050 淨零碳排的環保議題以及碳足跡盤查的需求也促使各界致力於紀錄詳實的生產歷程記錄。因此，為因應全球化的競爭，各國致力於相關政策的推動並協助各產業進行數位轉型。如何將產線工具機上的數據蒐集並整合至 SCADA 系統進行儲存與利用，為數位轉型的第一步。實務上，因工廠內設備數量多，且須依需求進行客製化，因此資訊系統的整合曠時費力，導入 SCADA 系統更加困難。在台灣，中小企業缺乏系統整合的專業資訊人才，如何加速資訊系統導入，成了各產業迫切的課題。本研究整理 OPC UA 資料交換標準的資料格式，並結合其於常用的圖像呈現方式與跨平台的使用者介面，開發一套 Low-Code SCADA 作業平台，使用者僅須具備基本資訊能力即可進行開發，經實驗驗證，本系統可減低 80%以上的開發成本，有效加速企業資訊平台整合與導入效能。

關鍵詞：工業 4.0、數位轉型、低程式碼開發、數據採集與監控系統、資料標準化

Designing and Implementing a Low-Code SCADA for Smart Manufacturing

Tsung-Sheng Cheng

Department of Electronic Engineering, Southern Taiwan University of Science and Technology

Abstract

The SCADA system is responsible for integrating the production information of an entire factory, which is the first step to enter smart manufacturing. To build up the system, existing development methods require the integration of multiple technologies, such as programming, database, and UI design. This research implemented a Low-code SCADA development system by combining the OPC UA standard and the display interface of common data types. In this system, the display elements were pre-built as modular icons. Through the data definition of OPC UA, we came to know the meaning of each data item. When developing SCADA, the developer can simply drag the pre-built icons onto the canvas and bind with data to complete quickly. In practice, this system can reduce the time and labor costs required by actual SCADA development.

Keywords: Industry 4.0, Digital Transformation, Low-Code Develop, SCADA, OPC UA

壹、緒論

工業 1.0 使用蒸氣為動力，出現機械代替勞力；工業 2.0 以電氣為主要動力，進入電氣化時代，於 20 世紀後期進入了第三波革命，以電腦協助人力製造，使得產線上的機器設備進入了數位控制的時代，但應用仍侷限於各機台的自主生產控制，整體產線仍缺乏機台與機台之間的協作與整合。工業 4.0 一詞在 2011 年於德國漢諾威工業博覽會上首次被提出[1]，並在短時間內得到德國政府、企業、研究機構的認同，2013 年德國政府更將工業 4.0 定義為國家戰略來重點發展。世界上主要工業國家也都立刻跟進，提出了對應的戰略方針，例如美國的「國家製造創新網路」、日本的「工業價值鏈」，中國大陸的「中國製造 2025」以及韓國的「製造業創新 3.0」[2]，而台灣則於 2015 年，由行政院提出了生產力 4.0 發展方案[3]，顯示世界各國對於第四波工業革命的重視程度。

工業 4.0 與 3.0 最大的差異點為，在新的時代中，以往各自獨立的運作的機器、工件和系統必須透過互相連接達到協作的目的，並透過雲端技術收集生產過程中的數據資料，以利更進一步導入 AI 技術，使生產線上的組件和系統都將能夠獲得自我控制與預測能力，除了可以為管理階層提供完整的工廠與產線狀態，以利做出正確決策之外，亦可透過數據監看與 AI 分析，預先掌握零組件的健康程度做出診斷，並使工廠管理部門問題發生前就可以預先進行因應與維護工作，實現預知停機與排程維護，提高設備生產稼動率。而產業進入智慧化的第一步必須是數位化，亦即所謂的數位轉型。

企業數位轉型步伐緩慢，IT 管理的技能短缺和系統複雜性增加，是目前企業進行數位轉型的最大瓶頸[4]。根據 ESG 研究，IT 技能短缺的問題約有 34% 是因為組織缺乏 IT 編排和自動化技能；33% 缺乏雲架構/規劃技能；32% 缺乏 IT 架構/計劃技能[5]。ESG 在年度技術支出意向調查中指出，75% 之 IT 決策者認為 IT 比兩年前更加複雜[6]，可見 IT 的技術能力缺乏是導致現有企業無法快速邁入工業 4.0 的因素之一。根據經濟部中小企業處統計，目前中小企業約有 140 萬家左右，佔台灣全體企業比重高達 97%，這些業者由於整體 IT 能力的缺乏，導致無法解決技術問題 [7]，本研究透過整合國際主要的工具機資料交換標準，以及 Web 技術建置一個低代碼（low-code）的 SCADA 平台，使業者可以採用最少的專業人力快速跨入工業 4.0 的領域。

貳、現有技術與標準

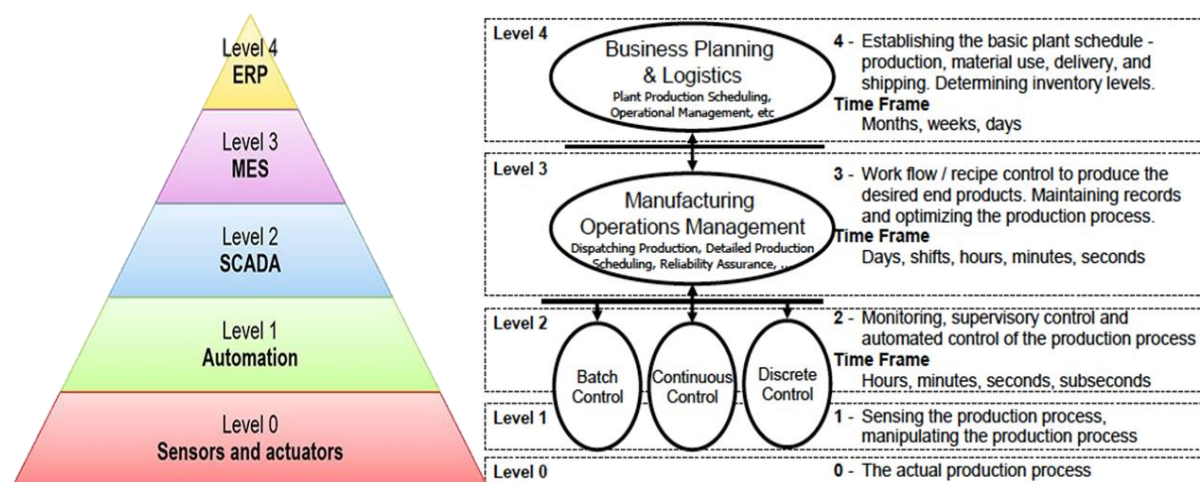
一、SCADA 系統

將產線上的人、機等生產過程中所產生的資訊蒐集、整合，以方便後續的分析、決策等應用，此類整合的功能在系統上被通稱為 SCADA（資料採集與監控系統，supervisory control and data acquisition）。依照國際自動化學會所定義的 ISA-95 國際標準，企業和控制系統之間的自動化介面如圖 1 所示，SCADA 屬於 Level 2，介於資訊（information technology）與實體技術（operational technology）之間，主要負責實體產線資料的監看與採集[8]。

因此 SCADA 系統的建立乃是產業數位轉型的第一步，而所收集的資料除了可作為 AI 分析應用之外，最重要的是將全場數據提供給管理者，常見的應用為建立中央戰情室，即時查看各產線數據與機器設備參數，同時對資料進行初步整理與儲存。SCADA 同時作為人與數據直接面對的管道，因此需要考量對於「人」提供即時易用的 UI 介面[5]，亦即資料的可視化。因為 SCADA 負責整合場內大量的機器設備，設計上必須盡可能的將所擷取到的設備參數轉化為圖形。例如，將數據與時間整合成曲線圖、將多個相似資料整理成柱狀圖以進行比較，甚至需要進一步對資料作處理，例如當數據超過警戒值時需要由系統發出通知。這些 UI 介面在現有的 SCADA 系統上都需要透過具有相當資訊能力的工程師進行設計與開發，而可視化的圖形與美工部分則需要另請美工專長人員協助，並且必須符合每個場域的實際狀況。因此每次重新建置或修改產線時，都必須再由程式開發與美工人員進行開發，而系統驗證時間都需數週之久。

圖 1

ISA-95 架構與功能說明



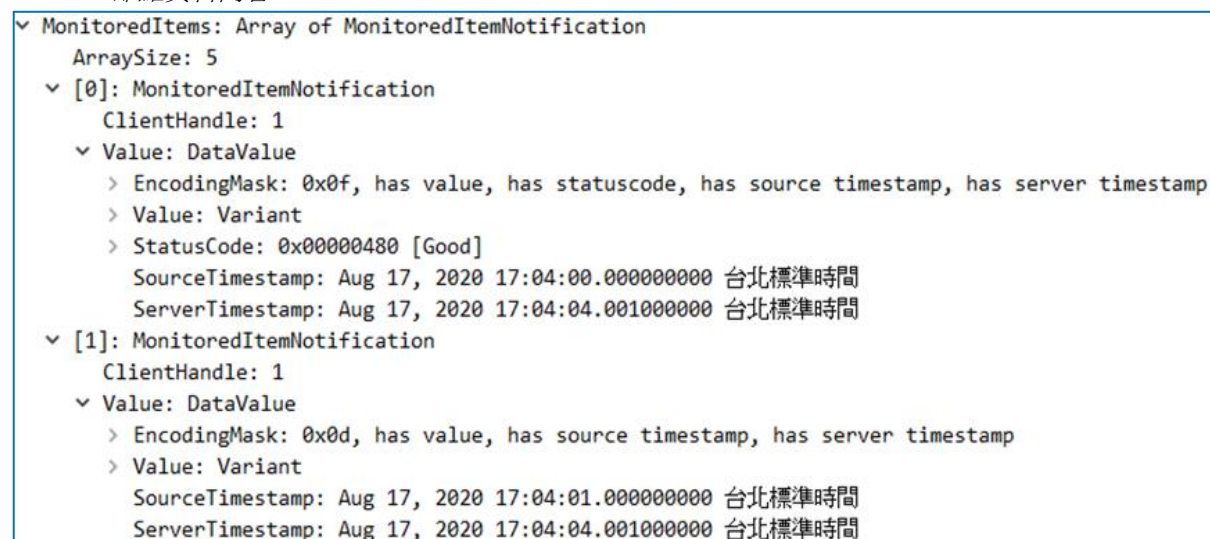
二、OPC UA 標準

在工業 4.0 的時代，機器設備間需要透過網路進行資料交換與溝通，但不同品牌的工具機多以自行定義的資料結構或通訊方式進行開發，甚至同廠牌的不同設備也無法溝通，因此在整合上需要耗費大量的時間來進行個別轉譯，不僅需要大量人力，也容易發生錯誤。OPC 統一架構（OPC UA）為解決此一問題而誕生[9]，OPC UA 是一種工業通信協議，已在 IEC 62541 中標準化，此標準因具備開放性、跨平台、安全性等特點，被許多的工具機業者所支援。綜觀世界各主要工業國家對未來發展所提出的相關計畫如德國的 Industry 4.0、美國 AMP 計畫、中國十二五計畫等，均建議以此做為資料交換格式[1]。而 OPC 基金會則持續透過舉辦活動和提供合規認證的基礎設施和工具，推動標準的不斷改進、配套規範的開發以整合現有信息模型。因此本研究採用此資訊架構作為交換標準，可達到最大的通用性。

OPC UA Part 5 規範了可以連接的物件類型（object type）[10]，每個物件類型都可以視為一個組件（component），其已有基本定義，而一些特定的 Object Type 已經作為某些服務的入口點，因此各 Request 可以透過這些被定義好的物件取得所需服務，而其他物件則可透過使用附加組件進行擴展。OPC UA 透過點位（node）編號來存取資料的物件[11]，並進而取得該點位的相關資料與數值，所讀取之節點資料如圖 2 所示。

圖 2

OPC UA 節點資料內容



三、Low-Code 技術

低程式碼（low-code）開發平台是產業分析公司 Forrester Research 於 2014 所提出的名稱[12]，此概念可以溯及到 1990 年代以及 2000 年代初期，以模型驅動設計（model-driven design）、自動化程式碼生成（automatic programming）以及視覺化程式設計的概念[13]。

低程式碼開發平台無須透過傳統的程式設計方式產生應用程式，而是針對特定應用而設計開發的，常應用於互動性的使用者介面（例如網頁應用程式），除了在一些特殊的情形下需要編寫程式之外，幾乎可以完全透過拖拉以及簡單參數設定完成。此類平台的好處是可以減少傳統程式碼的編寫時間。因為低程式碼的特性，參與者只需具備基本的資訊能力，不用有深厚的程式設計技巧，即可參與軟體的開發。各應用領域的從業人員可直接依照需求建立應用程式，省去與程式設計師反覆溝通的時間，更可以讓系統因符合現有工作流程而降低設定、訓練及環境準備等成本。現有的 SCADA 系統因為各公司的產線架構、生產流程均不相同，因此無法以套裝軟體方式直接使用，需針對每一應用場域進行客製化的開發，甚至因為訂單的少量多樣，造成每一次換線都需要重新調整系統，阻礙了企業進入工業 4.0 的期程。

本研究整合 OPC UA 標準，針對製造業的通用需求，導入 Low-code 開發平台的概念，開發符合智慧製造領域資料監看、蒐集、資料預處理以及示警功能的 SCADA 系統。

參、系統設計與規劃

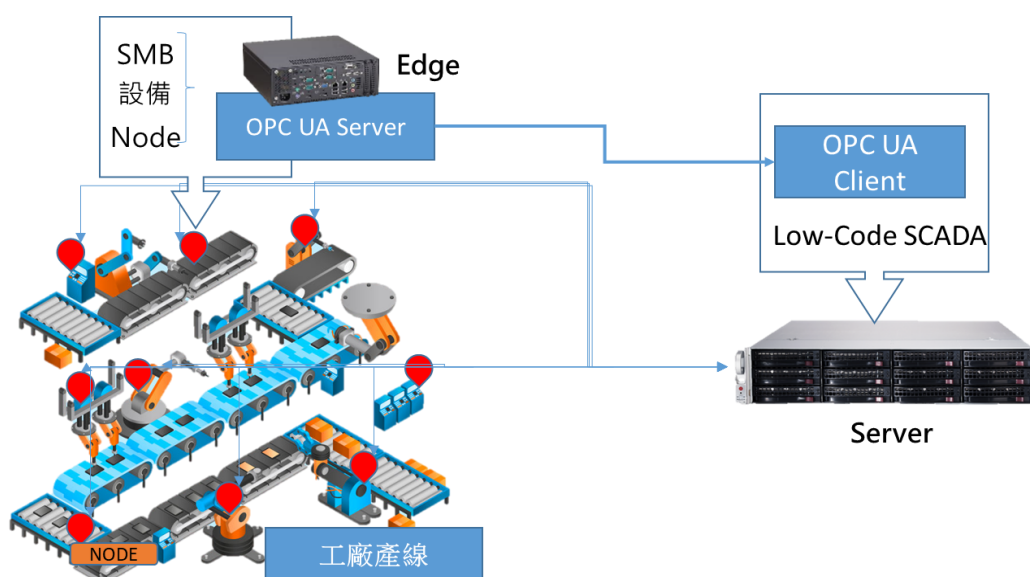
為了建立圖形化的監看介面，必須先建置符合需求的硬體環境、整合可用的軟體工具，再依照所接收的資料類別建立對應的顯示圖示。OPC UA 對於資料類型的定義內容雖然種類繁多[15]，且各種資料類型都有不同的呈現方式，但在實際使用上，仍可整理出常用的類型進行匹配後預先建置。例如，文字就是單純的顯示、布林值可能是燈號而數值可能需轉換成折線圖、圓餅圖等。

一、硬體架構

SCADA 系統主要目的為監看生產線上的機台資料，所建置的系統必須連結工廠設備以取得即時資訊。OPC UA 的架構分為資訊提供端（OPC UA server）與資訊接收端（OPC UA client），因此在硬體架構上必須兼具兩者的特性。現行工廠中多為新、舊機台混用的環境，少部分設備具有內建的 OPC UA server 功能可直接連接使用，而大部分的舊型機台多不支援 OPC UA，因此需要加上一台工業電腦（IPC）進行透過表頭轉譯，或先連接 SMB 智慧機上盒（smart machine box）進行資料轉換，此階段只須符合 OPC UA 規範即可，Low-Code SCADA 可透過 IP 進行連接，並透過 OPC UA 取得資料並解析。

圖 3

硬體系統架構圖



本研究所建立的 Low-Code SCADA 系統建置於伺服器端，因需同時接收與處理多部 OPC UA server 的資訊，且有大量的數據運算、儲存與可視化圖形處理需求，因此採用較高級的伺服器主機，本研究所建置的硬體規格如表 1 所示，可依所需監控點位數量與服務內容調整。

表 1

Low-Code SCADA 硬體規格表

	CPU	RAM	Storage	Network
規格	i7 以上	32G	4TB	1G，固定 IP

二、軟體架構

傳統 SCADA 系統 UI 的開發需結合美工設計、程式開發、資料整合等技術，而 Low-Code 是近來重要的系統設計概念，使用者透過預先設定、模組化、標準化及拖曳式的編輯工具，可降低開發者的負擔。因為製造產線的行業性質，其所需監控的數據多為帶有時間戳記的 Key-Value 樣式，如圖 2 所示。若要作為後續 AI 應用則需要較高的取樣頻率，會對資料庫造成較大的負擔。尤其是傳統的關聯式資料庫，若以每秒鐘取樣一次，每個月每個點位就會產生 2,592,000 筆數據，而這樣的資料通常不會用來進行檢索，而僅用來作為 AI 分析之用。這樣的資料型態適合放置於非關聯式的資料庫中，因此系統設計上除監控的顯示以及最新的 100 筆資料外，資料的儲存以 No-SQL 為主。因系統仍有登入、驗證以及設定等需求，所以仍保留關聯式資料庫，採兩者並存的方式規劃。

在應用 UI 方面，整個應用系統構建在單個集成開發環境上。這包括前端、後端、數據庫以及與現有系統或服務的集成。在 Low-Code 系統上，前端可拖曳的 UI 是必須的，這部分本研究使用標準 HTML 5、JavaScript 和 CSS 擴展 UI 的可用性，預先建置跨設備工作的模板和預構建的物件內容，並以 MCV 架構進行開發，基本軟體需求如表 2 所示。

表 2

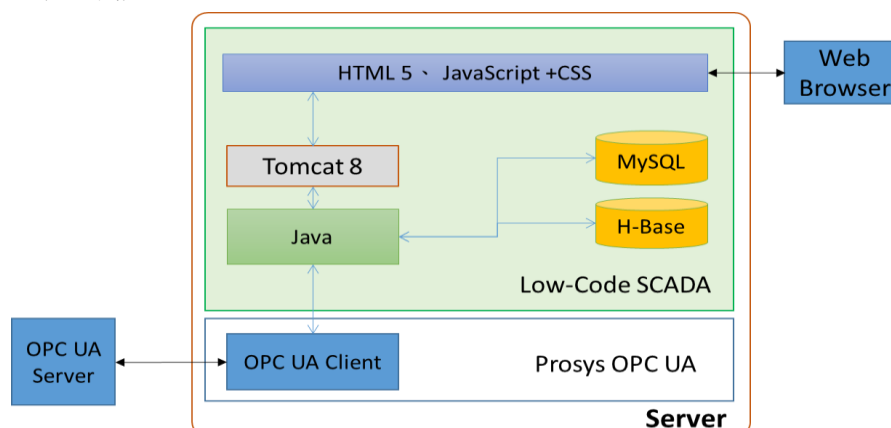
Low-Code SCADA 軟體規格表

	OS	網頁伺服器	資料庫	開發語言	IDE	其他	OPC UA
規格	ubuntu	Tomcat 8	MySQL& HBase	Java 8、HTML 5、JavaScript & CSS	VS Code	gzip	Prosys OPC UA

為實現跨平台的通用性，系統主要以 Java 開發，而開發的 IDE 輔助工具則使用 Visual Studio Code，因其免費且易用，所開發完成的程式則佈建到 Apache Tomcat 網頁伺服器上執行編譯，前端則整合 HTML 5、JavaScript 與 CSS 進行 UI 開發，其他相關軟體如 gzip 用來將使用者上傳的圖形檔案解壓縮，而 Prosys OPC UA 則為 Prosys 公司所開發的 OPC UA Client 的 SDK 套件，用以加速開發連接、接收與解析 OPC UA Server 的功能。各軟體間的關聯性與架構如圖 4 所示。

圖 4

軟體系統架構圖



三、系統設計與開發

在 OPC UA 的規範中，可以透過 LDS (local discovery server) 與 LDS-ME (local discovery server with multicast extensions) 技術在 LAN 網段中快速找到可用的 OPC UA Server 並連接，而在 SCADA 客製化的過程中最耗費人力與時間的是針對各廠域的現況進行美工編繪，以利使用者使用，程式開發師則需撰寫程式將所收到的資料映射至場域圖的個別位置，並依不同需求轉換成易讀的圖形。依 OPC UA 標準，SCADA 系統可能收到的資料格式共 25 種，如表 3 所示。

表 3

OPC UA Built-in Types

ID	Name	Description
1	Boolean	A two-state logical value (true or false).
2	SByte	An integer value between -128 and 127 inclusive.
3	Byte	An integer value between 0 and 255 inclusive.
4	Int16	An integer value between -32 768 and 32 767 inclusive.
5	UInt16	An integer value between 0 and 65 535 inclusive.
6	Int32	An integer value between -2 147 483 648 and 2 147 483 647 inclusive.
7	UInt32	An integer value between 0 and 4 294 967 295 inclusive.
8	Int64	An integer value between -9223372036854775808 and 9223372036854775807 inclusive.
9	UInt64	An integer value between 0 and 18 446 744 073 709 551 615 inclusive.
10	Float	An IEEE single precision (32 bit) floating point value.
11	Double	An IEEE double precision (64 bit) floating point value.
12	String	A sequence of Unicode characters.
13	DateTime	An instance in time.
14	Guid	A 16-byte value that can be used as a globally unique identifier.
15	ByteString	A sequence of octets.
16	XmlElement	An XML element.
17	NodeId	An identifier for a node in the address space of an OPC UA Server.
18	ExpandedNodeId	A NodeId that allows the namespace URI to be specified instead of an index.
19	StatusCode	A numeric identifier for an error or condition that is associated with a value or an operation.
20	QualifiedName	A name qualified by a namespace.
21	LocalizedText	Human readable text with an optional locale identifier.
22	ExtensionObject	A structure that contains an application specific data type that may not be recognized by the receiver.
23	DataValue	A data value with an associated status code and timestamps.
24	Variant	A union of all of the types specified above.
25	DiagnosticInfo	A structure that contains detailed error and diagnostic information associated with a StatusCode.

表 3 中 ID 1 為布林值，ID 2 至 9 為不同長度的整數，ID 10 與 11 為不同精度的浮點數，在呈現的設計上與整數類似，ID 12 與 15 為字串形式資料，ID 13 為時間，其他多為描述資料或點位狀態的資訊，可以字串方式涵蓋。因此在系統設計上主要以數值、文字、布林、時間這幾種呈現方式處理，而因應不同的場域需求，必須可以讓使用者上傳圖片，例如機器的外觀識別或是廠內產線的平面圖以增加資料的易讀性。

這些資料中，對於 SCADA 最重要也最常使用的是數字 (Integer and Float)。以單一數值而言，在大部分的設備上都會透過指針、儀表等形式呈現即時數值。在 SCADA 上，因為系統具有整合功能，可以

進一步與時間或是其他的數據結合，作為折線圖或是直方圖顯示，讓使用者可以較為方便觀察，例如，整合同一點位的資料時間與數值形成折線圖。

為達到統系統建置完成後的穩定性與可用性，本研究採用瀑布式（waterfall model）開發流程進行，以規劃、分析、設計、測試等管理與控制的流程進行，並於每階段測試後進行系統除錯與修改，以確保階段所開發的成果無誤。

肆、成果與驗證

本研究建置完成後交付給臺南市某系統整合廠商進行實測，透過一個實際的產線廢水處理管線案例進行 SCADA 系統建置，同時以傳統的工程師加美編設計師開發方式，與以本研究所建構的 Low-Code SCADA 系統進行開發，實際系統使用情形與建立的步驟說明如下：

第一步需建立 OPC UA 連線：要建立 SCADA 即時監看系統的第一步需要先與所要連結的設備建立連線，除了需要知道設備的 OPC UA Server 的 IP 位置外，OPC UA 所規範的預設連接 port 為 4840 或 4843，除非有特別設定否則可直接由這兩個接口進行連接，當 OPC UA 連線順利完成後即可瀏覽該 server 上可用的點位內容，其主要識別為 ID 與該點位的描述，這部分應於 OPC UA server 建置時設定容易識別的描述，如圖 5 所示。

圖 5

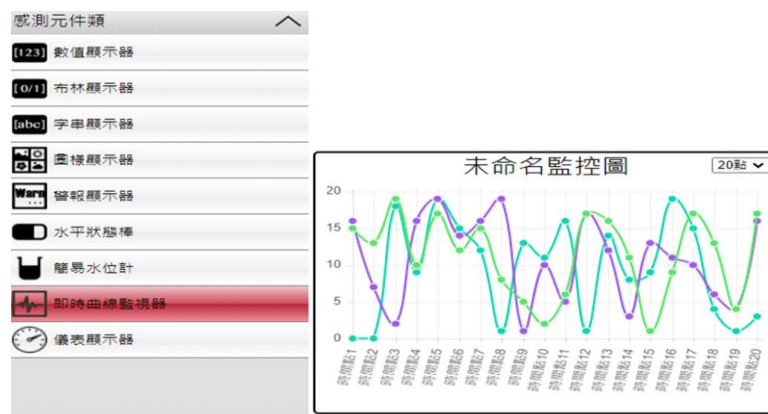
OPC UA 連線設定與點位預覽



第二步設定顯示圖樣並與 OPC UA 點位綁定：SCADA 系統主要的顯示內容為數值、文字、布林、時間與圖片，所以透過與實際場域考察與 SI 業者經驗，將現有機器常用的 UI，整理成常用的圖像物件，使用者可只需擊物件就可以動態產生一個預設圖形至畫面上，當系統接收到 OPC UA 的資料時，因為該資料已經經過標準化，所以可知道所代表之意義。例如，所收編號 x 的點位數值是 50，屬性被定義為某管線的壓力值，而在 SCADA 系統上想呈現的是對於該數據長時間的變化監視，此時可拖拉是先建置的折線圖套件拖曳至畫布上，並連結點位 x，即可將此數值有意義的呈現在系統上，如圖 6 所示。

圖 6

預設的常用圖示-折線圖



經分析，本系統在實務上可有效縮短開發時間與人力的主要原因為，使用者為該公司對客戶的主要窗口，最了解客戶需求，在傳統的開發過程中需要進行大量的會議討論，以同步各部門的技術與想法，這會消耗大量的時間成本。本研究所建置的系統可讓專案經理人進行開發，因其本身就是對該專案最了解的人員，可直接將想法實建，無須與技術研發人員和美編人員進行三方同步溝通，即可節省大量時間。又因該系統可由個人獨自操作，因此也可省去美工與程式完成後的整合與除錯工作，加上透過 OPC UA 的標準可以快速連接產線設備並擷取資料，因此大幅的縮短了 SCADA 系統建置的人力與時間。可預期使用者在經多次使用熟練度提升後，將可再縮短開發時間。

智慧製造已是世界趨勢，然中小企業因受限於規模以及成本考量，進入數位化的程度緩慢。或者是購買了先進的機台，卻無法發揮智慧製造的優勢，其主要原因是因為缺少了資訊整合能力，故無法進入整廠的智慧化。本研究所開發之系統經驗，經實務驗證可有效降低整廠資訊整合的成本，縮短系統開發時間，希望可以協助業界儘早進入資訊化與智慧化的世界，享受智慧化所帶來的美好成果。

參考文獻

- [1] 李坤敏、鄭琮生、陸一宏、胡曉（2018）。智慧製造的資訊標準與 SCADA 技術整合應用實例。*機械工業雜誌*，425，8-13。
- [2] 汪建南、馬雲龍（2016）。工業 4.0 的國際發展趨勢與台灣因應之道。*國際金融參考資料*，96，133-155。
- [3] 行政院科技會報辦公室（2016）。行政院生產力 4.0 發展方案。*台灣經濟論衡*，47-62。
- [4] 吳文治（2021）。企業 IT 維運面臨轉型瓶頸智慧基礎架構是解方。*網管人*，191，40-43。
- [5] Enterprise Strategy Group. (2020/1/7). *ESG master survey results: 2020 technology spending intentions survey*. ESG RESEARCH. <https://www.esg-global.com/research/esg-master-survey-results-2020-technology-spending-intentions-survey>
- [6] Bill Lundell. (2021/1/18). *ESG research report: 2021 technology spending intentions survey*. ESG RESEARCH. <https://www.esg-global.com/research/esg-research-report-2021-technology-spending-intentions-survey>
- [7] 廖家宜（2018）。台灣中小企業韌性度全球居冠 惟面對智慧製造仍心有餘力不足。Digitimes。 https://www.digitimes.com.tw/iot/article.asp?cat=158&cat1=20&cat2=10&cat3=10&id=0000524466_2MQ5ZRDE5SCW7D7UZH34R.
- [8] Scholten, B. (2007). *The road to integration: A guide to applying the ISA-95 standard in manufacturing*. ISA.
- [9] OPC Foundation (2017/11/22). OPC UA OPC 10000-1 - Part 1: Overview and concepts. <https://reference.opcfoundation.org/v104/Core/docs/Part1/>
- [10] OPC Foundation (2017/11/22). OPC UA OPC 10000-5 - Part 5: Information model. <https://reference.opcfoundation.org/v104/Core/docs/Part5/>
- [11] OPC Foundation (2017/11/22). OPC UA OPC 10000-6 - Part 6: Mappings. <https://reference.opcfoundation.org/Core/Part6/v104/docs/>
- [12] OPC Foundation (2017/11/22). OPC UA OPC 10000-8: OPC unified architecture. <https://reference.opcfoundation.org/Core/Part8/v104/docs/>
- [13] Richardson, C., and Rymer, J. (2014/6/9). New development platforms emerge for customer-facing applications: Firms choose low-code alternatives for fast, continuous, and test-and-learn delivery.

<https://www.forrester.com/report/New-Development-Platforms-Emerge-For-CustomerFacing-Applications/RES113411>.

- [14] 夏啟峻、鄭琮生、丁一華、陳蘊彥、梁儷齡（2020）。數據科學在智慧製造上的應用。科學發展，563，37-43。