Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 180 (2021) 561–570

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Industry 4.0 and Smart Manufacturing

Developing an OPC UA Server for CNC Machines

André Martins[d,*], João Lucas[c,d], Hugo Costelha[a,b,d], Carlos Neves[a,d]

[a] *INESC C, Polytechnic of Leiria, Leiria, Portugal*
[b] *INESC TEC, University of Porto, Porto, Portugal*
[c] *GLN Plast, Maceira, Portugal*
[d] *ESTG, Polytechnic of Leiria, Leiria, Portugal*

Abstract

This paper addresses the concept of Industry 4.0 from the perspective of the molds industry, a key industry in today's industrial panorama. With its constant modernization, several technologies have been introduced, in particular regarding machining equipment. With each brand and model requiring different (proprietary) interfaces and communication protocols, this technological diversity renders the automatic interconnection with production management software extremely challenging. In this paper a methodology to build monitoring solutions for machining devices is defined, based on the main equipment and operations used by molds industry companies. For a standardized approach, OPC UA is used for high-level communication between the various systems. As a key result of this paper, and given the variety of monitoring systems and communication protocols, the developed approach combines various different machine interfaces on a single system, in order to cover a relevant subset of machining equipment currently in use by the molds industry. This kind of all-in-one approach will give production managers access to the information needed for a continuous monitoring and improvement of the entire production process.

Keywords: Industry 4.0, OPC UA, Machine Monitoring, Computer Numeric Control.

1. Introduction

The so called shop floor digitalization is the basis of all developments considered within the scope of the 4th industrial revolution. In fact, the knowledge about the process, preferably obtained automatically, is the base of the architecture of a "Smart Factory" [1].

Nowadays, many industrial sectors, which are mostly formed by SME (Small and Medium Enterprises) use some type of direct data collection in their industrial processes. However, neither its application is universal, nor the level of adhesion from companies is similar, nor the depth of existing implementations is uniform [2]. The best examples are

* Corresponding author.
E-mail address: andre.martins@ipleiria.pt

1877-0509
10.1016/j.procs.2021.01.276

in industries with automatic assembly lines, such as electronics and automotive industries, areas which are extensively automated and where process data, therefore, is more easily available [2].

This availability stems directly from the presence of specific instrumentation needed for the automation of the process. Moreover, more useful data, possibly at a higher-level of abstraction, may be generated by the operation of the automation process.

Automated manufacturing has long since been a part of the molds industry, namely in CNC technologies, such as in milling, turning and electric discharge machining (EDM), with the connection to the machines for program transmission from machining software (DNC) being largely semi-automated. In the reverse direction, i.e., collecting information about the process or equipment states, the current level of automation is still very low, being neither universal nor done with the same depth-level in most companies [3].

In the authors opinion, this gap exists mainly due to the diversity of CNC controllers on the market and to the current lack of a *de facto* standard for machining interfaces, worsened by the fact that most machining equipment is either closed to third-party applications, or offers only a vendor-specific proprietary interface. These facts combined make automated machine data collection extremely difficult. The goal of this work is to develop a standard approach monitoring solution that minimizes this identified gap in machining processes.

To achieve the aforementioned goal, a methodology to build monitoring solutions for machining equipment is defined and tested, based on the main equipment and operations used in molds industry companies in one of the Portuguese machining clusters (known worldwide[1]), and based on OPC UA (Open Platform Communications Unified Architecture) for high-level communication between the various systems. Exploring the equipment characteristics and prioritizing the equipment interaction by means of a communication protocol, SDK (Software Development Kit) or API (Application Program Interface) when available. When this approach is not feasible due to the lack of proper equipment interfaces, a group of messages is defined and implemented, which can be exchanged with the equipment. In the worst case scenario, if the equipment does not support any communication at all, the approach relies on direct access to the sensors already integrated, or externally added, on the CNC machine which allows collecting performance metrics. If needed, these various approaches can be combined in order to acquire a more relevant data set, allowing for an improved monitoring of the system and even performing relevant predictions on process evolution.

There is a significant body of knowledge related to CNC machines monitoring techniques, specially with tool condition monitoring. The work carried by Downey et al. [4] [5], which collects and analyses data from three sensor technologies (force, acoustic and vibration) in a real time production environment, to monitor CNC tool wear. Other works, like the one by René de Jesús et al. [6] and the one by Stavropoulos et al. [7], uses driver current signal analysis as a sensorless approach on tool wear and breakage detection.

In what concerns the use of OPC UA standard on monitoring CNC machines over a wider range of parameters, there are fewer implementations and results available, of which the two most relevant are described here. The work from Mourtziz et al. [8] proposes an OPC UA-based framework for the modelling of milling and lathe CNC machine-tools. They also developed a data acquisition device to allow the integration of legacy machine-tools, without connectivity capabilities, in their holistic framework, presenting a laboratory case study to validate the proposed system. This work, have major developments around data acquisition using external sensors an gives less relevance to the direct data exchange with the CNC controllers.

Liu et al. [9] propose a Cyber-Physical Machine Tools (CPMT) platform based on OPC UA and MTConnect to enable standardized, interoperable and efficient data communication between machine tools and various types of software applications. To demonstrate the advantages of the proposed CPMT platform, different applications were developed, including an OPC UA client, an AR (Augmented Reality) assisted wearable Human-Machine Interface, and a conceptual framework for a CPMT-powered cloud manufacturing environment.

The remainder of this paper includes Section II, where a brief description of the main technology used in the developed work is presented. Following, on Section III, the definition of a methodology and the development of a CNC monitoring solution based on an OPC UA server is described. Section IV details the case studies used to test the developed methodology. Section V presents the obtained results, while Section VI provides for conclusions and future work directions.

[1] https://www.moldmakingtechnology.com/blog/post/state-of-mold-manufacturing-in-portugal

2. OPC UA

The main supporting technology of the developed work is the OPC UA standard. In this section, a brief description and the main characteristics of this standard are presented.

Defined by the international standard IEC 62541, OPC UA is the approach for a communication layer implementation recommended by the German RAMI4.0[2] (Reference Architecture Model Industrie 4.0) industrial platform [10]. OPC UA is a standard that, due to its characteristics, ensures the open connectivity, interoperability, security, scalability and reliability of industrial devices and systems. Based on a service-oriented architecture, where a single server supplies all the information and services with active security, from a given system, OPC UA is much more than a protocol. It has built-in information models and defines basic rules for information exchange and interfaces.

2.1. Open connectivity

OPC UA comes as a successor of OPC Classic[3], but supporting multiple operating systems. OPC Classic can be configured to provide a fair amount of security, but by relying on Microsoft Windows and DCOM/COM functionalities, it is difficult to setup in other operating systems. On the other hand, OPC UA was designed to work in multiple environments and on multiple platforms, with security built into the platform from the start [11].

2.2. Interoperability

In order to allow interoperability, OPC UA uses the Address Space concept to represent all the information generated from a specific system or device. This data is usable and can be interpreted by other systems using the same protocol, with its structure being built on information models, resulting on a standardized way to represent all data and information to be accessible by client systems. OPC UA uses the concept of object to represent system data and behavior. Objects serve as locations to store variables, events and methods, being also able to connect to each other using references. The standard information model is designed to allow type definition so that designers can meet their own application needs and, on top of this, information models can also be combined for specific areas, such as CNC systems [12]. These specific models are called Companion Specifications (CS) and derive from the standard model, inheriting its features, but can also include modifications, which can also be combined, depending on the system needs.

2.3. Security

OPC UA gives a lot of importance to the secure communication between client and server, addressing a broad range of applications, and encompassing different security and timing requirements. The available security modes, shown in Table 1, can enable both digital signature and encryption mechanisms, only digital signature mechanisms, or none of them. An OPC UA client can choose the desired security mechanisms from the ones available by a specific OPC UA server. After a secure channel (and session) has been established between server and client, there is still the need for user credentials represented by another certificate, or by a username/password combination [13]. Bearing in mind that most of the OPC UA applications are within industrial environments, security issues should not be forgotten. The OPC UA security analysis [14], commissioned by the German Federal Office for Information Security, presents recommended measures and procedures that should be followed when implementing OPC UA-based solutions.

2.4. Scalability and Compliance

Various OPC UA profiles are defined, as show in Table 2, with different application scenarios in mind, allowing OPC UA to scale down to a chip level, using the Nano Embedded Device profile, while still retaining its prominent features [16]. On the opposite side, when the implementation of complex scenarios is required, a PC-based server

[2] https://www.plattform-i40.de/
[3] https://opcfoundation.org/about/opc-technologies/opc-classic/

Table 1. OPC UA security modes [15].

Mode	Properties
None	No security.
Sign	Encoded with sender's private key. Only certificate owner has the private key. Anyone can verify the identity. Provides authenticity.
SignAndEncrypt	Adds encryption to sign. Encoding with receiver's public key. Anyone can encrypt. Only the certificate owner can read. Authenticity, confidentiality and integrity.

using the Standard UA profile could be the adequate solution. The OPC Foundation developed the UA Compliance Test Tool (UACTT)[4], as a way to identify which profiles are supported by our application, and to verify if the designed OPC UA server or client is compliant with the OPC UA specification.

Table 2. OPC UA profiles [17].

Profile	Characteristics
Nano Embedded Device	Limited functionality only, for very small devices, e.g. sensors. Only one connection, but without UA security, no subscriptions and no method calls possible.
Micro Embedded Device	Restricted functionality, at least two parallel connections, additional subscriptions and data monitoring, but no UA security and no method calls.
Embedded UA	Basic functionalities of OPC UA are available plus UA security and method calls.
Standard UA	Includes all functionalities for secure information access including UA security. No alarms and no history. PC-based servers should support at least this profile.

2.5. Communication Mechanisms

For information exchange, OPC UA offers two main communication mechanisms: Client-Server and Publish-Subscribe (PubSub) [18]. In the Client-Server mechanism, the client accesses the information provided by the server through defined services. The PubSub mechanism can be applied in different ways: for messaging over local area networks (LAN), data is published by the OPC UA server (publisher) and consumed by multiple authorized OPC UA clients (subscribers), using Time Sensitive Networking (TSN) to allow real-time low delay communications; for messaging over global networks (WAN/Cloud), OPC UA PubSub specification [19] defines mappings on protocols such as, for instance, MQTT[5].

3. Developing an OPC UA Server for CNC machines

With the current market diversity of CNC controllers, and given that some of this equipment is not open to third-party applications, the direct interaction with the controller becomes extremely difficult. On the other hand, recent equipment often has an available communication protocol, and some even have an SDK, an API, or simply the definition of a group of messages that can be exchanged with the equipment to this end. Alternatively to the acquisition methods using the information provided by the CNC controllers, direct access to the sensors already integrated, or

[4] https://opcfoundation.org/developer-tools/certification-test-tools/opc-ua-compliance-test-tool-uactt/
[5] http://mqtt.org/

externally added to the CNC, allows collecting performance metrics, which in turn allows predicting the machine state and/or process over time.

In this section, a methodology to develop a CNC monitoring solution based on an OPC UA server is described, explaining the development of a generic C# class, as well as the chosen approaches for data exchange with the CNC machines.

3.1. Generic C# class

In order to accommodate all this variety of paradigms, a generic C# abstract class (CNCBase) with abstract methods was developed to help modeling the interface for a generic CNC machine, allowing the development of monitoring application which work with different CNC machines. With this type of implementation, each CNC machine specific brand then overrides the generic class with its specific functions from its specific communication interfaces, but exchanging the information available using the same information model, Fig. 1.

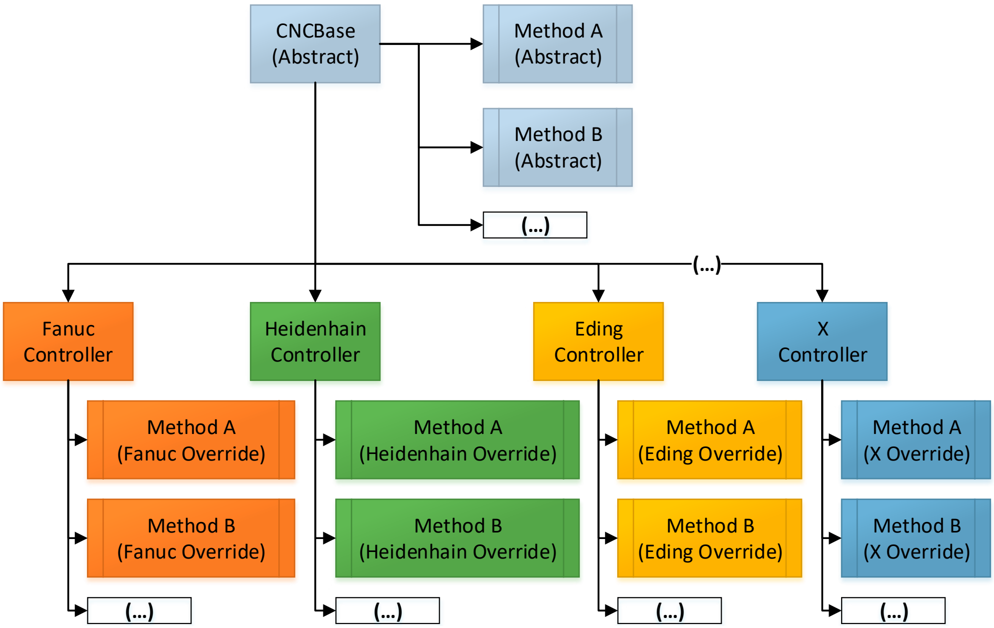

Fig. 1. Generic C# abstract class and methods.

This kind of implementation makes the future incorporation of other CNC machine models straightforward. Another relevant component of our application is the configuration file, which allows the definition of some application parameters, such as the CNC machine model/brand to be monitored, the communication address definition to be used, as well as the machine parameters and frequency to be monitored. The main goal of this file is the definition of run-time function parameters, i.e., without the need to recompile the server application for specific machines, or, for instance, whenever different data is to be acquired, or is to be acquired at different frequencies.

3.2. OPC UA server

With the goal of making the collected information available using a standardized approach, the OPC UA protocol was chosen. OPC Foundation provides several implementations in different programming languages. We chose to use the .NET stack version that is available as open-source project[6], currently with the license that allows developments

[6] https://github.com/OPCFoundation/UA-.NETStandard

within education and research fields. The .NET version of OPC UA is the one with most support and available implementations, backed by the OPC Foundation. Furthermore, the use of the .NET Framework approach allows the developed OPC UA servers on Windows, Linux and macOS (using .NET Core[7]), as well as in embedded systems applications (using nanoFramework[8]). The development of our application is based on the Reference Server[9], given that it was certified to be used in industry by UA CTT.

In order to define the Address Space of our OPC UA Server, the CNC CS [12] were considered, specifically following the OPC UA information model example of a 3-axis machine tool, as shown in Fig. 2. The information model that defines our CNC machines was developed using UAModeler[10] from Unified Automation. This tool provides a graphical and hierarchical representation of the designed model, following the OPC UA notation and syntax.

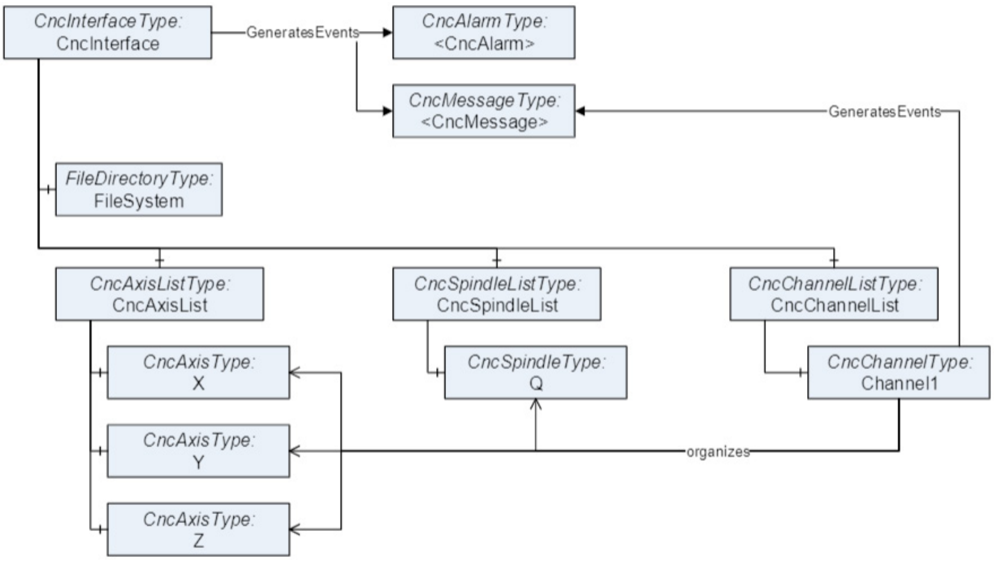

Fig. 2. 3-Axis machine tool OPC UA information model [12].

3.3. Data exchange with the CNC controller

Our implementation prioritize the CNC machine interaction by means of a communication protocol, SDK or API, when available. In addition or alternatively to this, when the above methods are not supported, but when a log system is still available, a group of messages is defined and implemented within the machining code, which can be sent from the equipment during the machining processes. This can be automated, by adding, for instance, in the FANUC case, DPRNT instructions in the CNC machine post-processor phase.

In addition to the acquisition methods refereed above, or when the equipment does not support any type of communication, one can use a direct (hardware-based) access to the sensors already integrated or externally added to the CNC machine, in order to acquire the needed data directly. In these cases, we propose an approach where devices can have wireless and/or wired network support, allowing to take advantages of both wired and wireless-based solutions [20], having in mind the tight requirements often imposed by the industrial environments [21].

The next section will provide further details about the methodology for developing OPC UA servers for CNC machines, by describing the application to specific scenarios, considering three different CNC controller vendors.

[7] https://docs.microsoft.com/en-us/dotnet/core/
[8] https://nanoframework.net/
[9] https://github.com/OPCFoundation/UA-.NETStandard/blob/master/SampleApplications/Workshop/Reference/README.md
[10] https://www.unified-automation.com/products/development-tools/uamodeler.html

4. Case Studies

The specific case studies were selected based on the main equipment and operations used by molds industry companies in the Marinha Grande cluster, in Portugal, and within our research lab. As show in Fig. 3, two types of CNC controllers used in the industry were chosen, namely the FANUC 31i-A and the Heidenhain iTNC530 CNC controllers, and, as part of a parallel project related with a CNC retrofitting process, an Eding CNC controller.

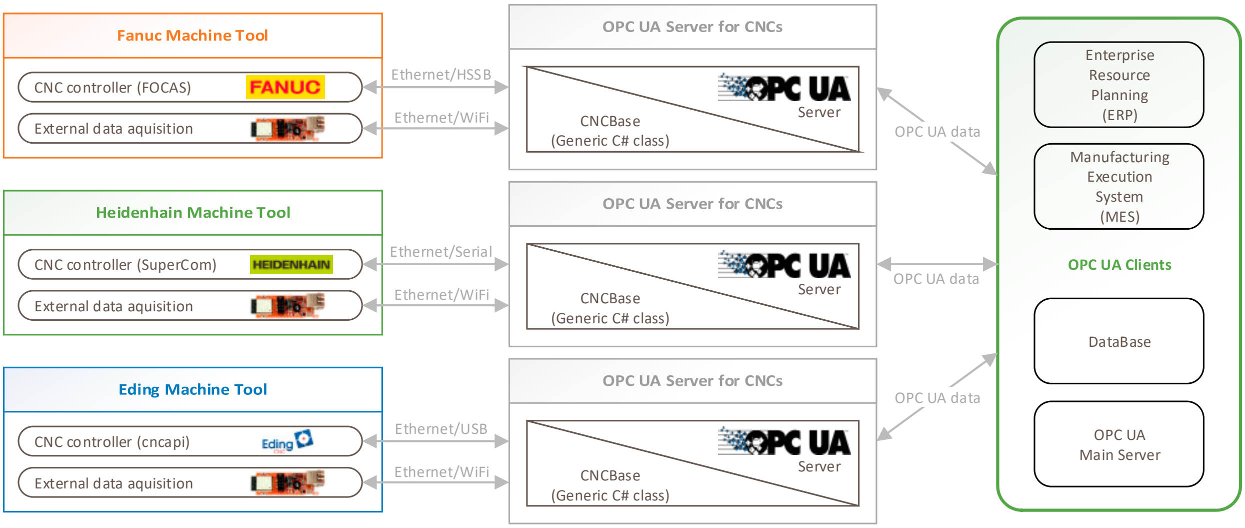

Fig. 3. Functional diagram and interaction between systems.

4.1. FANUC CNC machine

In what concerns data exchange with the 31i-A FANUC controller, the FOCAS (FANUC Open CNC API Specifications) library is used. This library allow data access from the CNC and PMC (Programmable Machine Control) controllers via Ethernet or HSSB (High Speed Serial Bus) [22]. The FOCAS library, in both the 1 and 2 versions, contains numerous functions[11] that enable data exchange with FANUC CNC machines [23]. However, this library is not supported by all the FANUC controller families, being restricted to machines from the i series. This library can be used for development of Windows and Linux-based applications, as well as Android and iOS mobile platforms.

Furthermore, FANUC controllers provide several commands that can output variable values and various characters through the RS232 or embedded Ethernet port to externally connected devices. These commands are called External Output Commands, and there are four of them: POPEN (Port Open) and PCLOS (Port Closed) are used to "connect" and "disconnect" to the output port, respectively; between these two instructions one can use BPRNT and DPRNT commands to output data in binary or ASCII format, respectively [24]. These commands can be added in the post-processor phase, allowing the acquisition of some machine parameters when the FOCAS library is not supported by the controller, or when the FOCAS library does not provide access to the needed information.

4.2. Heidenhain CNC machine

The solution used to exchange data with the Heidenhain iTNC530 controllers was based on the SuperCom Heidenhain communication library[12]. This Adontec library allows communication with Heidenhain TNC controllers through the serial or Ethernet (TCP/IP) ports, similarly to FOCAS for FANUC controllers. This library contains functions to

[11] https://www.inventcom.net/fanuc-focas-library/general/general
[12] https://adontec.com/

build data connections to one or more Heidenhain controllers. SuperCom is event-driven, and enables file transfer to and from the Heindenhain TNC, listing, creating and deleting folders, renaming and deleting files, reading TNC configuration data, retrieving machine status, machine data and process data, as well as reading and writing memory registers, among other functionalities. The amount of information available may vary between the different controller series. The library also contains direct access functions that can be used to retrieve or modify data directly from the controller connected PLC memory. This library can be used for developing both Windows and Linux-based applications.

4.3. Eding CNC machine

Our research lab as been recently involved in a parallel project [25] related with a CNC retrofit, using an Eding controller board[13]. In this case, data is exchanged with the controller through the USB or Ethernet (TCP/IP) interface using the cncapi[14] communication library. This library allows sending and receiving data by sharing its internal functions, which are also used by the graphical user interface that allows viewing and controling the Eding CNC controller. An OPC UA server was implemented using the methodology described above, in order to work in parallel and communicate with the Eding software. It is possible to exchange data with all the Eding controllers using this library, independently of the equipment to be controlled. Note, however, that the library only works on Windows. The Eding library is not available in C#, but only in C++. As such, an existing C++-based code wrapper for C# was used[15], so data could be accessed from our developed OPC UA server, built using C#.

4.4. External data acquisition

To acquire additional data and applying the above mentioned sensing techniques, we use the IoT development board ESP32-PoE[16], from Olimex. Built around the ESP32-WROOM-32 module [26], it supports WiFi, BLE and 100Mb Ethernet with Power-Over-Ethernet (PoE). It further includes a LiPo battery connector, a MicroSD card slot, GPIO (General Purpose Input Output) headers and an UEXT (Universal EXTension) connector that allow the connection of any device using I2C, SPI or RS232 communication. This board is also available in a galvanic isolated version for the PoE power and in an industrial version[17], supporting an extended temperature range from -40°C to 85°C.

5. Tests and Results

This section presents several results that were obtained from the application of the developed methodologies to the case studies described above.

In case of FANUC, the first tests were made using the NCGuide[18] CNC simulator. Some FOCAS functions are not available on NCGuide and because of this limitation, a test was conducted in an industrial environment, using our system combined with a database working as a datalogger of a real CNC machine.

With Eding, tests were made only in the CNC simulator from Eding, which is, in fact, the same software that is used in a real scenario i.e. when connected to a CNC controller board.

Regarding Heidenhain, tests were only made with iTNC530 CNC simulator, the main limitation of the trial version being the maximum of 100 lines of machine instructions. Due to this limitation, in this test, unlike the others, the CNC axis are moved manually, reflecting these changes in cutter position in the OPC UA server variables.

[13] https://www.edingcnc.com/products.php

[14] The cncapi and its documentation is only available after having installed the Eding CNC Software.

[15] https://www.oosterhof-design.com/cncapi-netframework/

[16] https://www.olimex.com/Products/IoT/ESP32/ESP32-POE/open-source-hardware

[17] https://www.olimex.com/Products/IoT/ESP32/ESP32-POE-ISO/open-source-hardware

[18] https://www.fanucamerica.com/products/cnc/software/cnc-guide

5.1. OPC UA information model

The hierarchically structured address space of the OPC UA server of one of the CNC machines is shown in Fig. 4 as an example. Server address spaces can be accessed with appropriate OPC UA generic clients, with the UAExpert[19] from United Automation being used in this case.

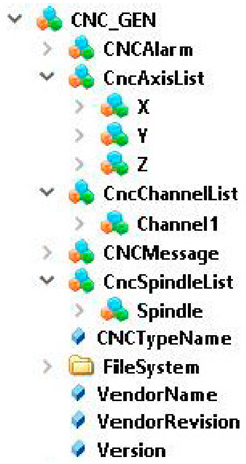

Fig. 4. CNC OPC UA server address space structure.

A video obtained during some of this tests is available in the following link: https://goo.gl/h7GVy3.

6. Conclusions and Future Work

Given the variety of monitoring systems and communication protocols, multi-vendor solutions must take an approach where the various systems available on the market are combined on a single system, allowing the data to be acquired from a wider range of machining equipment in the molds industry. Future work in this field will include data mining and machine learning techniques in order to detect faults and predict events on CNC machines.

We are also developing the interconnection of MES (Manufacturing Execution System) and ERP (Enterprise Resource Planning) software with shop floor data supplied from our OPC UA Servers, using KPIs (Key Performance Indicators) and OEE (Overall Equipment Effectiveness) metrics to determine efficiency and productivity of the machining processes. Here, the single OPC UA-based server interface plays an important role in guaranteeing a common communication and data model for CNC machines within a given company.

Recently, the Universal Machine Tool Interface (UMATI)[20] working group was established, in which OPC Foundation, the German machine tool builders association (VDW) and other industrial partners, are developing the OPC UA Companion Specification for Machine Tools. The aim is to develop an OPC UA information model as a universal communication interface for machine tools to "external" communication partners such as MES, ERP, automation systems or the cloud, planned to be released in the second half of this year[21]. Our developed methodology combines well with those developments. By adapting our interface with the resulting developments of this work group, we will guarantee that the systems based on our solution will work and support the integration, side-by-side, with future products that support this yet to be defined standard.

[19] https://www.unified-automation.com/products/development-tools/uaexpert.html
[20] https://opcfoundation.org/markets-collaboration/umati/
[21] https://opcua.vdma.org/en/viewer/-/v2article/render/47927388

Acknowledgements

This work was developed under the project TOOLING4G (POCI-01-0247-FEDER-024516), supported by Programa Operacional Competitividade e Internacionalização (POCI), Portugal 2020 and Fundo Europeu de Desenvolvimento Regional (FEDER). This project was also financed by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia, within projects UIDB/00308/2020 and UIDB/50014/2020.

References

[1] Ray Y. Zhong, Xun Xu, Eberhard Klotz, and Stephen T. Newman. Intelligent Manufacturing in the Context of Industry 4.0: A Review. *Engineering*, 2017.
[2] Reinhard Giessbauer, Evekyn Lübben, Stefan Schrauf, and Steve Pillsbury. *Global Digital Operations Study 2018 - How industry leaders build integrated operations ecosystems to deliver end-to-end customer solutions*. PwC, 2018.
[3] Carlos Neves, Hugo Costelha, and Luís Bento. Indústria 4.0 - A quarta revolução industrial. *O Molde nº118*, 2018.
[4] Jonathan Downey, Sebastian Bombiński, Mirosław Nejman, and Krzysztof Jemielniak. Automatic multiple sensor data acquisition system in a real-time production environment. In *Procedia CIRP*, volume 33, 2015.
[5] Jonathan Downey, Denis O'Sullivan, Miroslaw Nejmen, Sebastian Bombinski, Paul O'Leary, Ramesh Raghavendra, and Krzysztof Jemielniak. Real Time Monitoring of the CNC Process in a Production Environment- the Data Collection & Analysis Phase. In *Procedia CIRP*, volume 41, 2016.
[6] Romero Troncoso René de Jesús, Herrera Ruiz Gilberto, Terol Villalobos Iván, and Jáuregui Correa Juan Carlos. Driver current analysis for sensorless tool breakage monitoring of CNC milling machines. *International Journal of Machine Tools and Manufacture*, 43(15), 2003.
[7] P. Stavropoulos, A. Papacharalampopoulos, E. Vasiliadis, and G. Chryssolouris. Tool wear predictability estimation in milling based on multi-sensorial data. *International Journal of Advanced Manufacturing Technology*, 82(1-4), 2016.
[8] Dimitris Mourtzis, Nikolaos Milas, and Nikolaos Athinaios. Towards Machine Shop 4.0: A General Machine Model for CNC machine-tools through OPC-UA. In *Procedia CIRP*, volume 78, pages 301–306. Elsevier B.V., 1 2018.
[9] Chao Liu, Hrishikesh Vengayil, Yuqian Lu, and Xun Xu. A Cyber-Physical Machine Tools Platform using OPC UA and MTConnect. *Journal of Manufacturing Systems*, 51:61–74, 4 2019.
[10] Pablo Felipe Soares De Melo and Eduardo Paciencia Godoy. Controller Interface for Industry 4.0 based on RAMI 4.0 and OPC UA. In *2019 IEEE International Workshop on Metrology for Industry 4.0 and IoT, MetroInd 4.0 and IoT 2019 - Proceedings*, pages 229–234. Institute of Electrical and Electronics Engineers Inc., 6 2019.
[11] Paul Hunkar. OPC UA vs OPC Classic. Technical report, DSInteroperability, 2014.
[12] VDW and OPC Foundation. OPC UA Companion Specification for CNC Systems. Technical report, VDW and OPC Foundation, 2017.
[13] Salvatore Cavalieri, Marco Giuseppe Salafia, and Marco Stefano Scroppo. Integrating OPC UA with web technologies to enhance interoperability. *Computer Standards and Interfaces*, 61:45–64, 1 2019.
[14] Bundesamt für Sicherheit in der Informaationstechnik. OPC UA Security Analysis. Technical report, für Sicherheit in der Informaationstechnik, Bundesamt, 2017.
[15] Charles Varlei Neu, Ina Schiering, and Avelino Zorzo. Simulating and detecting attacks of untrusted clients in OPC UA networks. In *ACM International Conference Proceeding Series*, pages 1–6, New York, New York, USA, 11 2019. Association for Computing Machinery.
[16] Jahanzaib Imtiaz and Jurgen Jasperneite. Scalability of OPC-UA down to the chip level enables "internet of Things". In *IEEE International Conference on Industrial Informatics (INDIN)*, 2013.
[17] VDMA and Fraunhofer IOSB-INA. Industrie 4.0 Communication Guideline Based on OPC UA. Technical report, VDMA, Fraunhofer IOSB-INA, 2017.
[18] Peter Drahos, Erik Kucera, Oto Haffner, and Ivan Klimo. Trends in industrial communication and OPC UA. In *Proceedings of the 29th International Conference on Cybernetics and Informatics, K and I 2018*, volume 2018-January, pages 1–5. Institute of Electrical and Electronics Engineers Inc., 4 2018.
[19] OPC Foundation. OPC UA Specification Part 14 - PubSub 1.04, 2018.
[20] Lisa Underberg, Rudiger Kays, Steven Dietrich, and Gerhard Fohler. Towards hybrid wired-wireless networks in industrial applications. In *Proceedings - 2018 IEEE Industrial Cyber-Physical Systems, ICPS 2018*, pages 768–773. Institute of Electrical and Electronics Engineers Inc., 6 2018.
[21] Gianluca Cena, Adriano Valenzano, and Stefano Vitturi. Hybrid wired/wireless networks for real-time communications. *IEEE Industrial Electronics Magazine*, 2(1):8–20, 3 2008.
[22] Fanuc and Inventcom. FANUC Open CNC API Specifications, 2018.
[23] Sri H Atluru and A Deshpande. Data to Information: can MTconnect deliver the promise. *37th Annual North American Manufacturing Research Conference (NAMRC 37)*, 2009.
[24] P Smid. *Fanuc CNC Custom Macros: Programming Resources for Fanuc Custom Macro B Users*. Industrial Press, 2005.
[25] João Lucas. Desenvolvimento de servidor OPC UA para sistema CNC, 2019.
[26] Espressif. ESP32-WROOM-32 Datasheet. Technical report, Espressif, 2019.