Global Journal of Computer Science and Technology: G Interdisciplinary
Volume 20 Issue 3 Version 1.0 Year 2020
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

A Distributed-Ledger, Edge-Computing Architecture for Automation and Computer Integration in Semiconductor Manufacturing

By Da-Yin Liao

Abstract- Contemporary 300mm semiconductor manufacturing systems have highly automated and digitalized cyber-physical integration. They suffer from the profound problems of integrating large, centralized legacy systems with small islands of automation. With the recent advances in disruptive technologies, semiconductor manufacturing has faced dramatic pressures to reengineer its automation and computer integrated systems. This paper proposes a Distributed-Ledger, Edge-Computing Architecture (DLECA) for automation and computer integration in semiconductor manufacturing. Based on distributed ledger and edge computing technologies, DLECA establishes a decentralized software framework where manufacturing data are stored in distributed ledgers and processed locally by executing smart contracts at the edge nodes. We adopt an important topic of automation and computer integration for semiconductor research & development (R&D) operations as the study vehicle to illustrate the operational structure and functionality, applications, and feasibility of the proposed DLECA software framework.

Keywords: distributed ledger technology; smart contract; edge computing; automation; computer-integration; semiconductor manufacturing.

GJCST-G Classification: C.2.4

ADISTRIBUTEDLEDGEREDGECOMPUTINGARCHITECTUREFORAUTOMATIONANDCOMPUTERINTEGRATIONINSEMICONDUCTORMANUFACTURING

Strictly as per the compliance and regulations of:

A Distributed-Ledger, Edge-Computing Architecture for Automation and Computer Integration in Semiconductor Manufacturing

Da-Yin Liao

Abstract- Contemporary 300mm semiconductor manufacturing systems have highly automated and digitalized cyber-physical integration. They suffer from the profound problems of integrating large, centralized legacy systems with small islands of automation. With the recent advances in disruptive technologies, semiconductor manufacturing has faced dramatic pressures to reengineer its automation and computer integrated systems. This paper proposes a Distributed-Ledger, Edge-Computing Architecture (DLECA) for automation and computer integration in semiconductor manufacturing. Based on distributed ledger and edge computing technologies, DLECA establishes a decentralized software framework where manufacturing data are stored in distributed ledgers and processed locally by executing smart contracts at the edge nodes. We adopt an important topic of automation and computer integration for semiconductor research & development (R&D) operations as the study vehicle to illustrate the operational structure and functionality, applications, and feasibility of the proposed DLECA software framework.

Keywords: distributed ledger technology; smart contract; edge computing; automation; computer-integration; semiconductor manufacturing.

I. Introduction

As applications and technologies in semiconductors continue to advance, semiconductor manufacturing must aggressively evolve to fulfill the changing technical and business requirements in the semiconductor industry. Contemporary 300mm semiconductor manufacturing systems have highly automated and digitalized cyber-physical integration. They are an ideal example in realizing *Industrie 4.0*, *Smart Manufacturing,* or *Digital Enterprise* [1-3], all the widely used terms describing *the Factory of the Future*, the convergence of disruptive and innovative technologies of information, operations, and data. While the industry is experiencing its fourth revolution, semiconductor manufacturing systems get prepared in moving forward to their paradigm shift toward the future semiconductor factory (a.k.a. a *wafer fab*, or commonly called a *fab*).

Semiconductor manufacturing deals with the production of integrated circuits (IC) products. Based on semiconductor process technologies, semiconductor manufacturing goes over a sequence of processing stages through circuit design, wafer fabrication, assembly/packaging, and final testing. Semiconductor manufacturing aims to meet all the specified requirements on product functionality, quality, cost, reliability and durability, and regulations on the environment, safety, and health (ESH). Semiconductor manufacturing is a both technology- intensive and capital-intensive business. The fabrication of semiconductor products costs a lot of money. Building a new 300mm production fab now could cost several billion dollars. The investment skyrockets even upwards to twenty billion dollars for the most advanced process node of 3nm semiconductor technology. Semiconductor manufacturers use semiconductor wafers for the fabrication of IC devices. A wafer serves as the substrate for semiconductor devices to build in and upon. It goes through hundreds of processing steps, including processes of oxidation, doping, implantation, etching, deposition, and photolithographic patterning. The entire manufacturing process involves several hundreds of sophisticated equipment (or tools) for processing and metrology of semiconductor wafers [4,5]. Semiconductor wafer fabrication consists of the production of the discrete, batch, and continuous flow processes. Due to the considerations on ergonomics, safety and fab efficiency, 300mm semiconductor manufacturing regards the Automated Material Handling System (AMHS) as a must [6-8].

Effective manufacturing of semiconductor products demands a high level of automation and computer integration in allocation, coordination, and collaboration among system dynamics as well as flows of data, information, command, control, communication, and materials. Automation in semiconductor manufacturing uses a hierarchical control architecture design [9]. In the lower level of the hierarchy, there are embedded controllers that provide real-time control and analysis of fabrication equipment. Each equipment utilizes sensors for *in situ* monitoring and characterization. In the higher level, more complex, context-dependent combination of the operations in process and metrology equipment as well as the movement of materials are handled, sequenced, and executed. International Technology Roadmap for

Author: Straight & Up Intelligent Innovaetions Group Co., San Jose, CA, USA. e-mail: eliao@miicg.com

Semiconductors 2.0 [10] defines ten functionality areas for fab integration in semiconductor manufacturing. They are *Fab Operations (FO)*, *Production Equipment (PE)*, *Material Handling Systems (MHS)*, *Fab Information & Control Systems (FICS)*, *Facilities*, *Augmenting Reactive with Predictive (ARP)*, *Big Data (BD)*, *Control Systems Architectures (CSA)*, *Environmental Safety and Health (ESH)*, and *Yield Enhancement (YE)*. Among these ten functionality areas, FO is the major driver of requirements and actions for required semiconductor fab services. FICS is the facilitator of the integration in semiconductor manufacturing. It covers computer hardware and software, manufacturing execution and decision support systems, fab scheduling, automation of equipment and material handling systems, and process control. In semiconductor manufacturing practices, FICS is carried out by fab CIM (Computer-Integrated Manufacturing) systems.

Semiconductor equipment automation deals with the control and sequencing of job track in/out, process start/stop in equipment, collection of measurement data, change of processing variables, and selection/validation of processing recipes. To facilitate equipment automation in semiconductor manufacturing, standardization of communication protocols among various semiconductor equipment vendors is crucial. Semiconductor Equipment and Materials International (SEMI) [11] defines a collection of common equipment behaviors and communications capabilities of semiconductor equipment. SEMI published the famous SECS (SEMI Equipment Communications Standard) standards of the SECS-I (also, called SEMI E4) in 1980 and the SECS-II (also called SEMI E5) in 1982, respectively. SECS is a point-to-point, layered protocol via RS-232 serial communication. SECS consists of three levels. They are Message Protocol (SECS-II), Block Transfer Protocol (SECS-I), and Physical Link (RS-232). In 1994, SEMI announced the TCP/IP HSMS (High-Speed SECS Message Services, SEMI E37) standards. HSMS is SECS-II over TCP/IP. In 1992, SEMI announced the GEM (Generic Model for Communications and Control of Manufacturing Equipment, SEMI E30) standard which defines a standard implementation of SECS-II for all semiconductor manufacturing equipment and a generic set of equipment behavior and communications capabilities that provide the functionality and flexibility to support the manufacturing automation programs of semiconductor manufacturers. The GEM standard defines which SECS-II messages should be used, in what situations, and what the resulting activity should be. Integration of equipment of different process types from independent equipment suppliers with the same solution methodology and software is thus possible.

Both SECS and GEM are communication interface protocols used for communication between a semiconductor manufacturing equipment and a fab host computer. The SECS/GEM standards define messages, state machines, and scenarios to enable fab automation software (or Equipment Automation Program, EAP) to control and monitor the manufacturing equipment. EAP automates equipment operations of recipe upload/download, collection of state variables and metrology data, and handling of events and alarms. EAP can also act as the interface between equipment and various fab CIM systems and functions, such as MES (Manufacturing Execution System), RMS (Recipe Management System), SPC (Statistical Process Control), APC (Advanced Process Control), and so on. However, the integration of EAP to fab CIM systems and functions depends on fab operations requirements and limitations on message response times and communication bandwidth. The essence of EAP is a communication gateway coded with the state machine mechanism of operations logic. Basic computing power without local storage is sufficient to execute an EAP. Also, modern Equipment Engineering Systems (EES) require immense quantities of equipment and process data. The collection of specific data from equipment complies with the SEMI Equipment Data Acquisition (EDA, also known as Interface A) standards.

For the past two decades, most 300mm fabs have adopted the SEMATECH CIM Framework [12,13] to promote computer integration on their planning and operations management through object-oriented technologies. The SEMATECH CIM Framework is a software infrastructure with components that provide functionality common across various applications, which are thus integrated by the CIM Framework. The core of the SEMATECH CIM Framework offers a family of abstractions and services to support fab operations and decision making. Applications that utilize these abstractions and services are deployed and executed on a distributed objective-oriented computer platform. The SEMATECH CIM framework addresses the needs to improve the problems of integrating large, monolithic, centralized legacy systems with small islands of automation in semiconductor manufacturing environments.

With the promotion to *the Factory of the Future* in full swing, semiconductor manufacturing has faced dramatic pressures to reengineer its automation and computer-integrated systems. The use of scalable, distributed architecture to decentralize management and control of manufacturing systems has shown promising as an alternative to classical hierarchical control architectures. As the deployment of distributed computing resources in manufacturing systems increases, many efforts on the design of scalable and distributed manufacturing (SDM) systems [14] have proposed to provide more flexibility, traceability, agility, utilization of manufacturing resources, and timely and dynamic control to production. However, building a sound SDM system still faces fierce challenges. They include (i) the increased system complexity and maintenance efforts, (ii) the difficulty in system synchronization, data consistency and system deployment, (iii) the additional computation and exchange of information, and (iv) the difficulty in security protection and identity verification.

Distributed Ledger Technology (DLT) [15], also better known as blockchain technology (a specific type of DLT) [16], is a disruptive enabling technology that has attracted massive attention and given rise to multiple projects in various industries these years [17,18]. Instead of keeping data centralized in a traditional ledger, DLT stores, shares, and synchronizes transactions in digital ledgers distributed on independent nodes of a network, i.e., a network of distributed ledgers. Especially, each distributed ledger is allowed only to store and organize its data in an append-only mechanism. DLT has features of immutability, transparency, and trustworthiness. It enables transparent, secure, trustworthy, and swift public or private solutions. As a technology establishing a distributed, high-trust data management system, DLT has the potentials for both storing data and increasing the effectiveness of managing the stored data.

DLT classifies and uses three groups of data. They are processing logs, states, and executable code of smart contracts [19]. DLT enables the transitions of state to take place locally and follow the state machine of smart contracts, once some specific criteria met. The decentralized characteristics of DLT enable services or applications (executed via smart contracts) to store and process data close to the place which creates the data. Edge Computing (EC) [20] brings computation and data storage at the edge nodes of a network. Such decentralization of data and their processing at the edge of the network has many advantages of time and cost. Instead of time-consuming operations of forwarding data to the centralized hosts and then processing data in the hosts, EC provides its services and applications closer to end-users with fast processing and quick response time, and with fewer costs required. As processing takes place locally, data are much safer and more private in the EC paradigm.

This paper deals with the design and applications of distributed ledgers and edge computing for automation and computer integration in semiconductor manufacturing. Instead of centralized services provisioning and functions in traditional semiconductor manufacturing systems, we develop a distributed-ledger, edge-computing architecture (DLECA). DLECA utilizes distributed ledger and edge computing technologies to establish a distributed software framework. DLECA stores data in distributed ledgers using a distributed, append-only, time stamped data structure. Data in a distributed ledger are composed of not only its processing requirements and logs; but also complex state variables with smart contracts. State variables are updated dynamically using edge computing of smart contracts once specific criteria met. Such decentralization of data and their processing increases the overall effectiveness and efficiency of planning and operations management in semiconductor manufacturing. We choose the important topic in pioneering semiconductor manufacturing for automation and computer integration of semiconductor research & development (R&D) operations as the study vehicle to illustrate the operational structure and functionality, applications, and feasibility of the proposed DLECA software framework.

The remainder of this paper is as follows. Section 2 gives a brief overview of automation and computer integration in semiconductor manufacturing first, followed by a review on the technologies of distributed ledgers and edge computing and how these technologies can disrupt automation and computer integration for manufacturing systems. Section 3 describes automation and computer integration in semiconductor manufacturing. Section 4 presents the design of DLECA-based cell controllers as a basis for a distributed-ledger, edge-computing framework for automation and computer integration in semiconductor manufacturing detailed in Section 5. Section 6 describes and analyzes the case study of applying the developed DLECA framework for the management of computer-integrated semiconductor R&D operations and their automation. Section 7 concludes this paper.

II. Literature Review

Automation and computer integration in semiconductor manufacturing requires seamless communications, coordination, management, and orchestration among materials, equipment, and automated operations within a semiconductor fab. The Microelectronics Manufacturing Science and Technology (MMST) program [21] first designed the well-known Computer-Integrated Manufacturing (CIM) System Framework to meet manufacturing demands on fully integrated dynamic systems. The MMST CIM framework combines the concepts of lean, flexible, and agile manufacturing to define high-quality manufacturing standards. It provides a disruptive approach to semiconductor manufacturing strongly relied on intelligent and flexible systems. Following the MMST CIM System Framework, SEMATECH proposed the CIM Framework Specification [22], an abstract model for semiconductor manufacturing systems. The SEMATECH CIM framework defines a component-based architecture for the next generation of agile MES and focuses on the integration of fab MES applications. As computing technologies continue to move forward, the coverage of fab MES functionalities has changed significantly. Various hierarchical structures have been developed for

Global Journal of Computer Science and Technology (G) Volume XX Issue III Version I Year 2020

vertical integration of fab automation systems into MES to allow for a seamless flow of control and information. In the last decades, both the academic and industrial communities have devoted to the development of advanced CIM architectures that adopt object-oriented and open approaches to integrate several CIM systems from multiple suppliers [23-25]. Lee [26] reviews automation requirements and technologies for semiconductor manufacturing, including fab integration architectures and fab operations with automated material-handling systems (AMHS), communications and networking, fab control application integration, and fab control and management. Liao [9] deals with the automation and integration problems in semiconductor manufacturing and proposes an intelligent AMHS management framework to optimize the integration of fab operations with AMHS. To our best knowledge, neither papers nor research results have been published so far on the design and applications of distributed ledger and edge computing technologies for automation and computer integration in semiconductor manufacturing.

In a manufacturing system, the functionality of MES supports most of its manufacturing processes, from production order release to delivery of finished goods [27]. The increasing use of sensors and high-speed networks has resulted in the continuous generation of big data. It also triggers renowned models in decentralized and distributed manufacturing systems, including the development of scalable distributed manufacturing (SDM) systems [28]. More and more designs of intelligent, distributed, and collaborative control systems have been proposed and put into practice in semiconductor manufacturing. Holonic and multi-agent control systems [29] have features of intelligence, autonomy, coordination, reconfigurability, and extensibility. Along with the tides in industrial digitalization, both academic and industrial research groups have made a lot of efforts in digital manufacturing [30, 31], which utilize a highly promising set of technologies to reduce the time and cost of product development and to provide mass customization of products in high quality and prompt delivery. Bratukhin and Sauter [32] investigate if and how distribution of existing centralized MES functions is possible and reasonable at the expense of increasing coordination and communication among the entities involved. This paper proposes a Distributed-Ledger, Edge-Computing Architecture (DLECA), where MES functionalities are partially distributed to the edge nodes to make decision-making processes more flexible.

Released in 2005, the IEC 61499 Standard [33, 34] provides a generic model for distributed industrial control and automation systems where programmable logic controllers (PLC), intelligent devices and sensors are integrated. The IEC 61499 architecture adopts an event-driven execution mechanism that allows an explicit specification of the execution order of *function blocks*, the fundamental model of the IEC 61499 Standard. Each function block comprises an Execution Control Charts (ECC), which is a state machine and able to trigger the execution of algorithms as defined in the compliant standards. The network of interconnected function blocks form and define the applications. The IEC 61499 Standard is application-centric. In a system, applications are created for the whole system and then distributed to the available devices accordingly. Therefore, the applications are distributed but maintained together. Interested readers can refer to the up-to-date surveys on the automation technologies and architectures of manufacturing control systems in [35,36].

Distributed Ledger Technology (DLT) relies on a distributed, decentralized, peer-to-peer network that utilizes cryptographic hashes and consensuses mechanisms [37]. In a distributed ledger network, a digital ledger is replicated and shared across multiple peer-to-peer participants. Data stored in a distributed ledger are verifiable and unable to change. Blockchain [38] is a data structure that creates a distributed digital ledger. As a subset of DLT, blockchain technology is the underlying technology of Bitcoin [39] and many digital cryptocurrencies. A distributed ledger is programmable with scripting. A smart contract [40], as a scripting code in DLT, is a program of business logic (or a state machine with a set of state-response rules) that autonomously executes based on the defined rules. The potentials and challenges of using blockchain and smart contracts in developing applications for Industry 4.0 are studied and surveyed in [41]. The research of [42] surveys blockchain technology on its working principles and elements in distributed control and cooperative robotics, which highly demands secure and distributed mechanisms.

Based on a hierarchical control structure, Stanciu [43] presents a blockchain-based, distributed control system for edge computing. A three-tier model for edge computing is adopted where devices are at the bottom, a mesh of edge nodes in the middle, and cloud services on the top of the control hierarchy. There are blockchains deployed on the top level, where smart contracts in a blockchain provided as a cloud service make the strategic decisions. The research of [44] proposes a reference architecture for industrial automation. The architecture combines edge computing and blockchain technologies for flexible, scalable, and reliable configuration and orchestration of automation workflows and distributed data analytics. Based on edge computing and blockchain technologies, Isaja and Soldatos [45] introduce the Reference Architecture (RA) and platform design of the H2020 EC co-funded FAR-EDGE project for developing industrial automation systems. The proposed RA provides functionalities in three complementary domains. They are domains of automation, virtualization of production systems, and data analytics. The RA is composed of four tiers, including Field Tier, Edge Tier, Ledger Tier, and Cloud Tier, from the bottom to the top for describing the structure of a system. This paper adopts a three-layer, hierarchical control model for automation and computer integration in semiconductor manufacturing, detailed in the following sections.

III. Automation and Computer Integration in Semiconductor Manufacturing

Semiconductor manufacturing systems are large-scale complex systems. Industrial automation and control in large-scale complex manufacturing systems usually bases on distributed hardware with the hierarchical design of automation and control functions. Due to the complexity of fab operations, current practice in semiconductor manufacturing automation systems adopts the classical hierarchical control model in implementing their automation and control functions. The model decomposes a large-scale complex system hierarchically into multiple levels of control subsystems. Control subsystems are linked together using hierarchically integrated control mechanisms where the flow of control is strictly vertical and between adjacent levels with data shared across one or more levels of the hierarchy.

The implementation of existing semiconductor manufacturing automation usually involves the integration of three levels of functions [9]. They are *Fab Automation* at the top, *Cell Automation* in the middle, and *Tool Automation* at the bottom. In the top level, Fab Automation covers system integration, manufacturing execution, scheduling and dispatching, activity management, and preventive maintenance. In the middle level, Cell Automation bridges the information exchange in both directions, manages material movement and control, tool connectivity, and equipment communication and control. Also, Cell Automation executes manufacturing processes and technologies like automatic data collection (ADC), overall equipment effectiveness (OEE), and so on. As the foundation level in the automation hierarchy, Tool Automation automates the processing of equipment to minimize or eliminate misoperations caused by human operator errors. Tool Automation also consists of automation of materials handling and metrology tools, wafer sorters, reticle inspection tools, reticle stockers, wafer stockers, and Automated Materials Handling Systems (AMHS). Figure 1 depicts the hierarchical automation in semiconductor manufacturing.

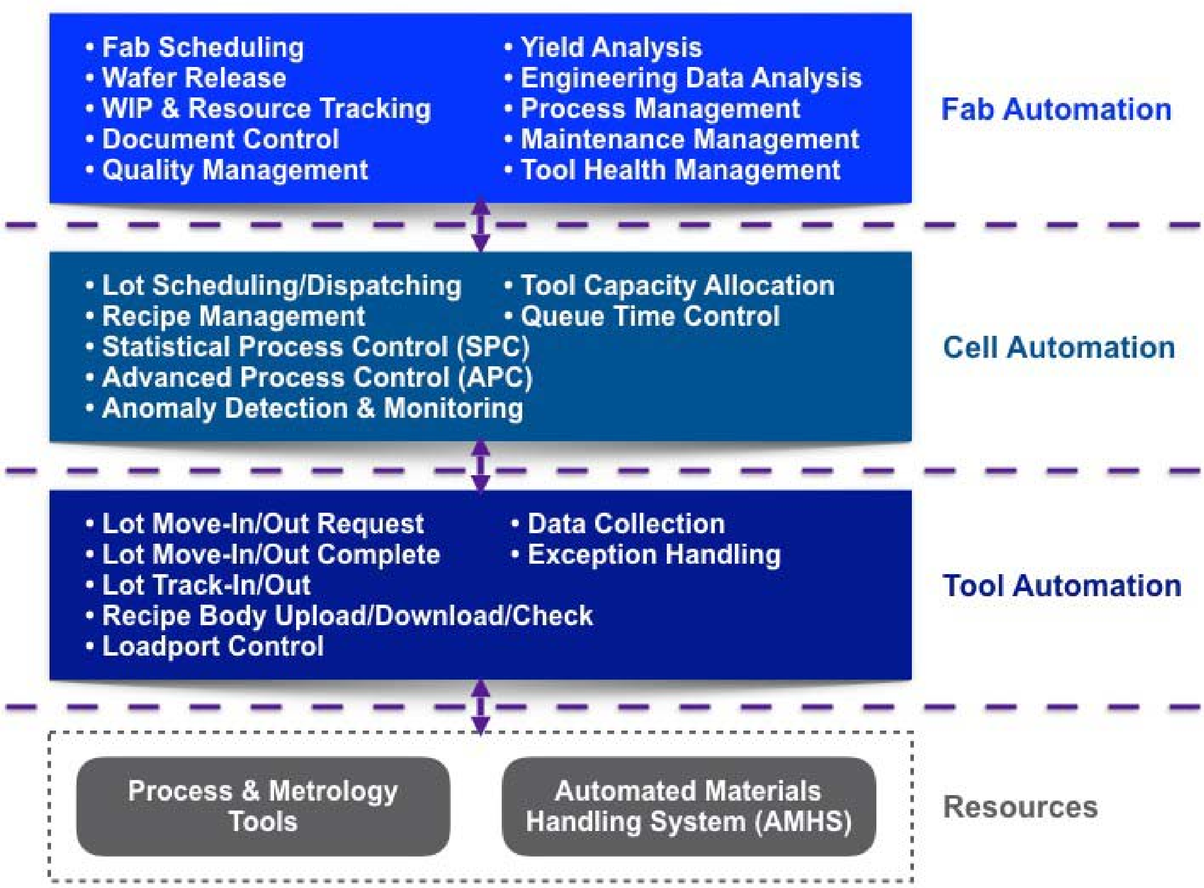

Figure 1: Hierarchical Automation in Semiconductor Manufacturing

Global Journal of Computer Science and Technology (G) Volume XX Issue III Version I Year 2020

Semiconductor manufacturing heavily relies on a broad array of computer systems to satisfy customers' requirements. Computer integration in semiconductor manufacturing involves the integration and coordination among diverse computer systems, applications, and a huge amount of data generated during the production of semiconductor products. The SEMATECH CIM Framework [22] provides a reference architecture that integrates and exploits the capabilities of the hardware, software, and production process concepts to enhance overall business performance in semiconductor manufacturing.

Computer integration for Fab Automation includes typical fab MES applications, including factory services, factory management, specification management, yield enhancement, material management, tool management, reporting, and scheduling. These applications are essential elements for vertical integration between the fab shop floor systems and the enterprise systems like ERP (Enterprise Resource Planning), SCM (Supply Chain Management), PDM (Product Development Management), and so on. In modern fabs, applications of Recipe Management System (RMS), Statistical Process Control (SPC), Predictive Maintenance (PdM), Advanced Process Control (APC), Real-Time Dispatching (RTD), and Fault Detection and Classification (FDC) are usually used for Fab Automation. To facilitate the integration of these applications, customized data models are needed to maintain and automate their Extract, Transform, and Load (ETL) capabilities. In practice, the deployment of Fab Automation systems and applications uses centralized computing power on mainframes or computer servers with large, centralized databases such as relational database management systems (RDMS).

Equipment Automation Program (EAP) plays a dominant role in computer integration for Tool Automation. EAP connects the real world (equipment) to the digital world and allows a host computer system to control and automate the processing. An EAP streamlines the business logic (or code) that interacts with the host system and the equipment to control and automate its processing. Each EAP executes based on a SECS driver that provides control and communication interfaces to the controlled equipment and the host system by SECS-defined timed sequences. A typical code of an EAP implements the automated human operations and acknowledgments such as Lot Move-In/Out Request, Lot Move-In/Out Complete, and Lot Track-In/Out; automated step control, and recipe selection and verification such as Recipe Body Upload/Download/Check. An EAP also deals with automated data collection of engineering and equipment statuses and the integration of automated load ports. In an EAP, its business logic includes automated exception handling of equipment alarm notification, logging, and reporting. The deployment of an EAP can use lightweight computing power on a distributed personal computer (PC) in a bus network topology; or on an instance of virtual machine (VM) for deploying and serving as virtual computers.

Operations in semiconductor manufacturing generally take place in a distributed way. Most semiconductor fab operations and decisions are made locally at the physically separated place. Considering the equipment functionality and efficiency in semiconductor manufacturing, the common fab configuration consists of tens of manufacturing cells. Within each manufacturing cell, computer systems are used for planning, controlling, and executing the production activities in the cell. Such manufacturing cells are autonomous, i.e., with the power to self-government. Each manufacturing cell is capable of managing the fabrication of wafers within the cell. The management of fab operations in a manufacturing cell includes dispatching jobs to all workstations in the cell, monitoring the equipment states, and feeding back to its upper-level supervisor systems. Cell Automation provides functionality and applications of tool dispatching, cell scheduling, tool allocation, overall equipment effectiveness (OEE), recipe management, real-time SPC, anomaly detection and classification (ADC), and tool control. Cell Automation may also act as the fab materials management controller and provides functionality and applications of AMHS management, reticle management, OHT (Overhead Hoist Transport) dispatching, and material control. Cell Automation uses small, rugged computers, called cell controllers. A cell controller provides coordination among individual process and metrology tools and their integration with Automated Materials Handling Systems (AMHS) within a cell.

Figure 2 illustrates the applications and functionality in the three hierarchical levels of Fab Automation, Cell Automation, and Tool Automation, which automates and controls the semiconductor manufacturing resources of process and metrology tools, testers, AMHS, and ARMS (Automated Reticle Management Systems). The CORBA (Common Object Request Broker Architecture) Standard [22] is adopted to facilitate the integration and communications among the diverse systems and applications in both Fab Automation and Cell Automation. Except for some legacy equipment models using serial communications (via RS-232 and SECS I for Tool Automation) only, all levels of fab automation and computer integration can now implement on an Ethernet network (via HSMS in Tool Automation). Figure 3 demonstrates a typical network topology used in a semiconductor fab.

In semiconductor manufacturing, cell controllers enable decentralized and distributed decision making at the edge of the fab network backbone. Manufacturing activities that take place in heterogeneous systems or equipment can be coordinated and controlled by distributed and federated cell controllers to cope with the fast- changing, flexible semiconductor manufacturing environment. The design of distributed cell controllers requires truly distributed workflows and automation logic. This research develops the services and interfaces that implement decentralized, local business logic as smart contracts on top of a distributed ledger for cell controller design. A distributed ledger is a transactional ledger that stores and maintains shared states and data, which are frequently read but infrequently written concurrently by smart contracts. All the smart contracts developed for cell controllers are scenario-specific and able to execute fast and simple logic with associated states and data within the cell scope. Next Section details the structural and functional design of a distributed-ledger, edge-computing architecture. The proposed architecture defines a runtime environment for workflows of edge automation of cell controllers in semiconductor manufacturing.

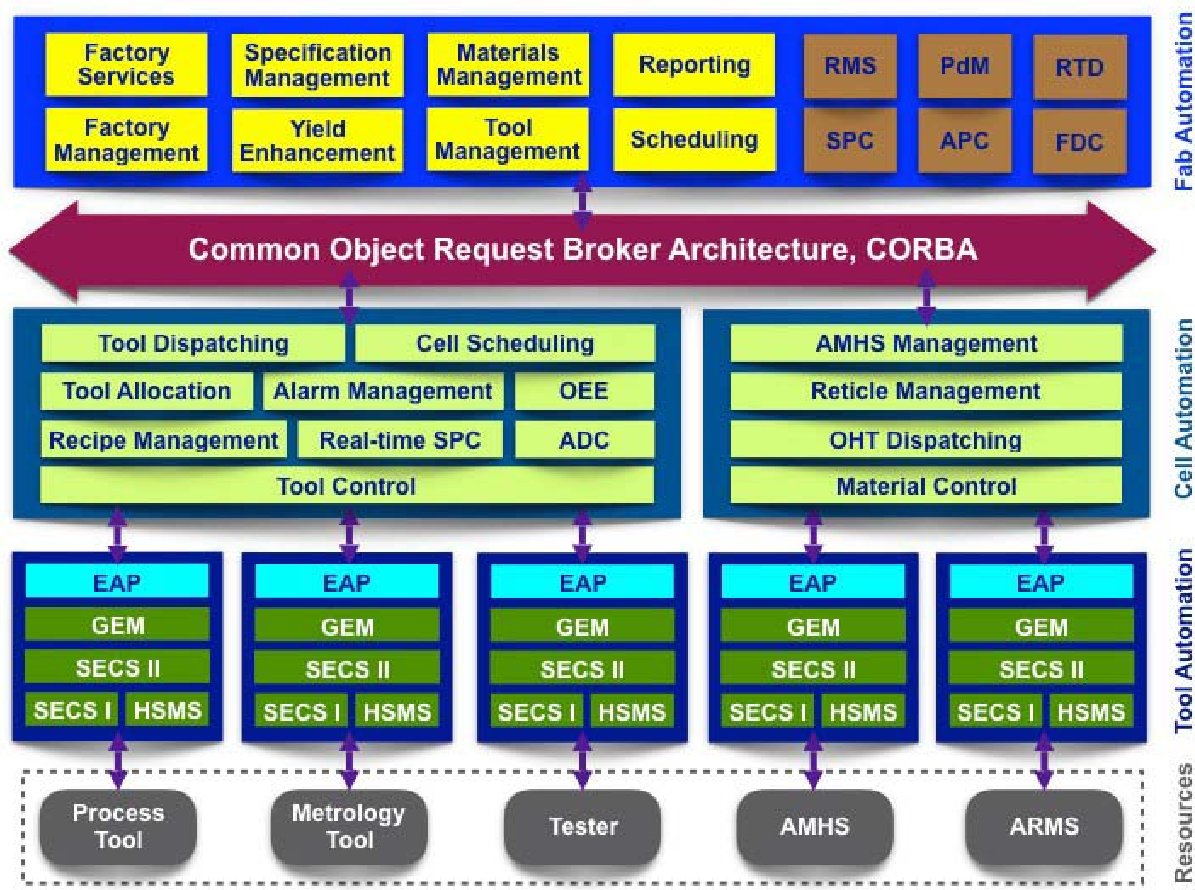

Figure 2: Hierarchical Computer Integration in Semiconductor Manufacturing

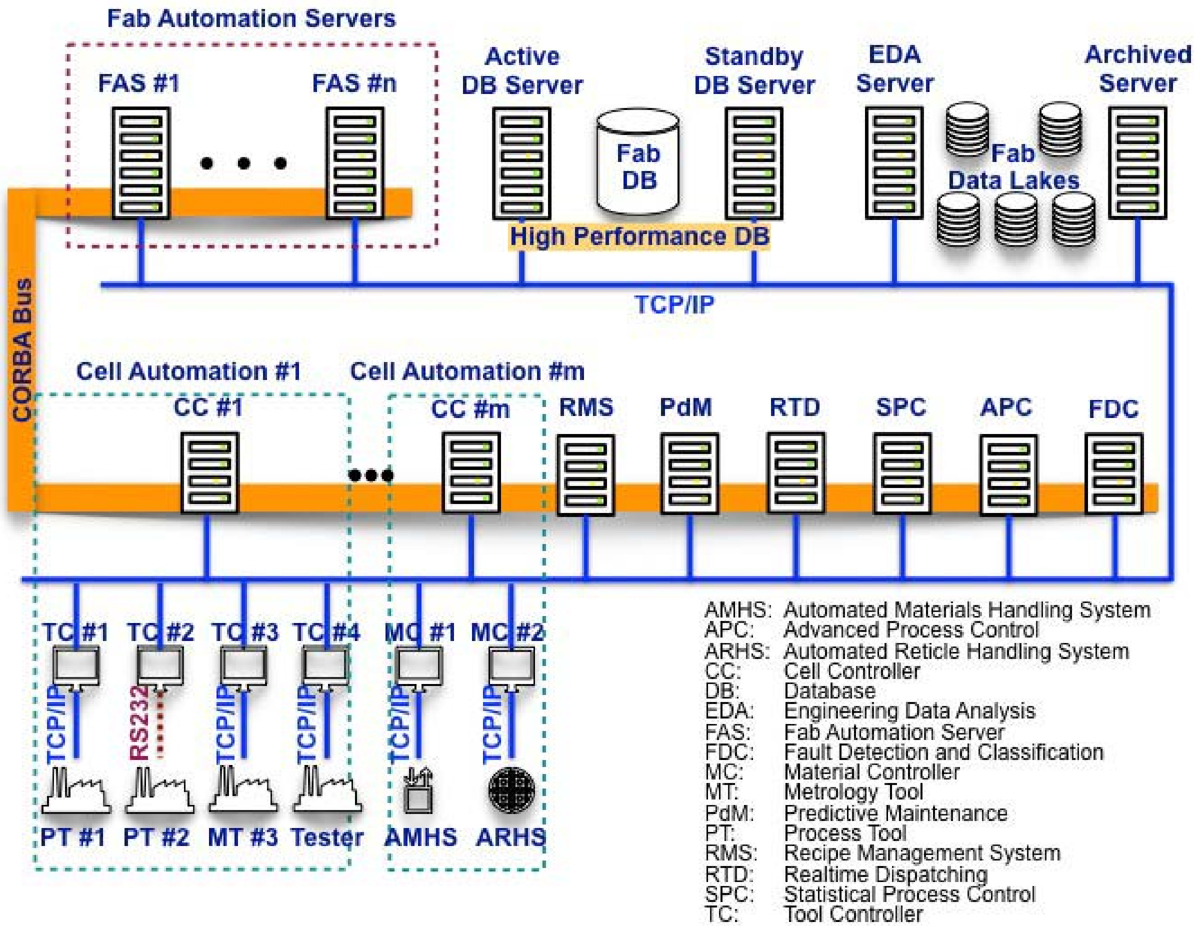

Figure 3: A Typical Network Topology in A Semiconductor Fab

IV. Distributed-ledger, Edge-Computing Architecture (dleca) for Cell Controller Design

This section describes the design of a distributed-ledger, edge-computing architecture (DLECA) for cell controllers, from both structural and functional points of view. DLECA aims to facilitate efficient and effective automation and computer integration in semiconductor manufacturing. Components in DLECA are stacked into a four-tier structure of Distributed Ledgers, Smart Contracts, Interfaces & Events, and Applications, as shown in Figure 4. The following describes the functions of the components in each tier of DLECA.

a) Distributed Ledgers

In this foundational tier, distributed ledgers and their associated services provide the data model, storage mechanism, and their basic CRUD (Create, Read, Update, Delete) operations for cell controllers. A distributed ledger is a shared digital ledger of persistent storage that exists across several locations and among multiple stakeholders of the distributed ledger. Transactions or data updates are only ever stored in the distributed ledger when the stakeholders have reached a consensus. In distributed ledgers, all files are time stamped and given a unique cryptographic signature so that all the data stored are verifiable, auditable, and historically traceable. Different from blockchain as a sequence of blocks of data, distributed ledgers are not required to link into a chain. There are several ways of organizing distributed ledgers. In this paper, we adopt the directed acyclic graph (DAG) data structure in organizing the distributed ledgers in cell controllers. In the DAG data model, all flows of transactions and data updates follow the same direction from earlier to later. This paper defines and uses four classes of distributed ledgers for cell controllers. They are distributed ledgers of *Product*, *Process*, *Resource* and *Service*. The following lists the components and their ingredients in Distributed Ledgers Tier.

- Distributed Ledger Stacks, including
  - core distributed ledger technology
  - storage and database used to store ledgers and data in the distributed ledger network
  - cryptographic processes and signature creation
  - consensus rules
  - peer-to-peer synchronization
  - ability to execute code of smart contracts in the cell controller

  - application program interfaces (APIs) that allow data to be read and appended
- Consensus Service, including
  - processes in which an agreement is made on the state of a distributed ledger
  - consensus rules among the stakeholders in the distributed ledger network
  - validator nodes in the distributed ledger network
- Crypto Service, including
  - processes where all records are timestamped with a given unique cryptographic signature
  - processes where all information is securely and accurately stored using cryptography
  - management of keys and cryptographic signatures used for access to data
- Network Protocol, including
  - a set of rules to ensure the data integrity on the distributed ledger network
  - a set of rules to provide network operations with scalability and low end-to-end latency
  - processes that provide peer-to-peer synchronization among the cell controller and others
- Registry Service, including
  - processes that define the registry of ownership of data in a distributed ledger
  - processes that move the registry of ownership of data between distributed ledgers
- New Ledger Service, including
  - processes that create a new ledger on the distributed ledger network
  - a set of rules that define the specifications on a new distributed ledger
  - a set of rules that validate the feasibility of smart contracts associated with a new ledger

b) *Smart Contracts*

In this Smart Contracts tier, ten groups of smart contracts are defined for the provisioning of a combination of fast and simple code snippets (the functions) and data (the states) for processing distributed ledgers at the cell controller. The ten groups of smart contracts are described in the following.

- Execution Management, including
  - code snippets that execute tool automation workflow of the cell controller
  - code snippets that handle exceptions or unexpected events on the cell controller
  - code snippets that monitor the execution and system status of the cell controller
  - code snippets that share the execution and system status to other cell/fab/tool controllers
- Tracking, including
  - code snippets that provide the communications between the cell controller and its tool controllers
  - code snippets that provide the communications between the cell controller and material controllers
  - code snippets that monitor the execution of automated operations from its tool controllers
  - code snippets that monitor the execution of automated materials handling from material controllers
  - code snippets that track products/lots/WIP (Wafers in Process) in the cell
  - code snippets that track quality of products and processes in the cell
- Definition Management, including
  - code snippets that manage the data object attributes in distributed ledgers
  - code snippets that manage information of process/product/recipe/data collection in distributed ledgers
  - code snippets that perform version control of information of process/product/recipe/data collection
  - code snippets that manage process flow/Q-time/batch operations in the cell
- Detailed Scheduling, including
  - code snippets that schedule the materials transport and wafer processing operations in the cell
  - code snippets that allocate capacity to wafer storage and processing in the cell
  - code snippets that generate sequence and timing for tool automation activities in the cell
- Dispatching, including
  - code snippets that prioritize manufacturing sequences of tool automation activities in the cell
  - code snippets that dispatch lots to a tool or a FOUPs (Front Opening Unified Pod) to AMHS
  - code snippets that dispatch tasks to tool and material controllers
- Data Collection, including
  - code snippets that define process data collection specifications
  - code snippets that define measurement data collection specifications
  - code snippets that collect process or measurement data reported by tool automation
  - code snippets that evaluate data based on different specifications of individual tool characteristic
  - code snippets that calculate data collected by specific pre-/post-measurements operations
- Performance Analysis, including
  - code snippets that monitor real-time product and tool status in the cell
  - code snippets that evaluate yields of products and processes in the cell
  - code snippets that evaluate cycle times of products and lots in the cell

- code snippets that evaluate throughputs and OEE in the cell
- code snippets that evaluate anomaly, fault, alarm conditions in the cell

- Resource Management, including
    - code snippets that manage materials handling and processing capacity in the cell
    - code snippets that manage reticles, dummy/control wafer operations in the cell
    - code snippets that schedule preventive maintenance activities in the cell
    - code snippets that execute tool predictive maintenance in the cell
- Data Compiling & Parsing, including
    - code snippets that compile the operations of process and control jobs and send to the tool controller
    - code snippets that compile the operations of material control job and send to the material controller
    - code snippets that compile recipe info and recipe body and send to the tool controller
    - code snippets that parse tool and lot status info and send to other cell and fab controllers
    - code snippets that parse processing results and send to other cell and fab controllers
    - code snippets that parse data collection results and send to other cell and fab controllers
    - code snippets that parse exception handling info and send to other cell and fab controllers
    - code snippets that parse e-Diagnosis info and send to other cell and fab controllers
- Ledger Lifecycle Management, including
    - code snippets that verify the transactions and signatures of a distributed ledger
    - code snippets that trace the lifecycle history of a distributed ledger, from its creation to archive

c) *Interfaces & Events*

In the Interfaces & Events tier, a series of interfaces provide the access to use of distributed ledgers in the cell controllers. Application program interfaces (APIs) are provided to access data of smart contracts in distributed ledgers. Events and their handling services are defined for operations in cell controllers. Services to peer nodes are provided. The interfaces and event handling services of Interfaces & Events Tier are described in the following.

- Distributed Ledger Gateway
    - interface that connect an application program to a distributed ledger
    - interface that consolidate data from multiple distributed ledgers for application programs
    - interface that access data of smart contracts in distributed ledgers
- Data Service
    - processes that provide data of distributed ledgers to application programs
    - processes that provide aggregated data from multiple distributed ledgers for application programs
    - processes that provide data of smart contracts in distributed ledgers
    - processes that review and apply analytic data assessment of distributed ledgers
- Service & Process Automation
    - processes that automate the manufacturing services of cell controllers
    - processes that automate the manufacturing processes of cell controllers
    - processes that manage the automated smart contracts
- Event Handling Service
    - a set of rules that define the events of cell controllers
    - processes that classify the events taking place in cell controllers
    - processes that handle the expected and unexpected events
- Peers Service
    - processes that access its peer-to-peer copies of a distributed ledger
    - processes that ensure peer-to-peer synchronization in the distributed ledge network
    - processes that manage the peer-to-peer network topology of distributed ledgers

d) *Applications*

In the Applications tier, applications are used to manage operations in Cell Automation. The applications commonly deployed to support cell controller functionality include tool dispatching, cell scheduling, tool allocation, overall equipment effectiveness (OEE), recipe management, real-time statical process control (SPC), anomaly detection and classification (ADC), and tool and material control, listed as the follows.

- Tool Dispatching
    - application programs that determine next lot(s) to be processed by a tool when the tool becomes idle
- Cell Scheduling
    - application programs that determine the processing sequence and timing for tools in cell controllers
- Tool Allocation
    - application programs that allocate tool capacity in cell controllers
- Alarm Management
    - application programs that provide actions to mitigate abnormal situations in cell controllers
- Overall Equipment Effectiveness (OEE)
    - application programs that provide visibility to the manufacturing effectiveness in cell controllers

- Anomaly Detection and Classification (ADC)
  - application programs that identify and analyze anomalies in cell controllers
- Real-time Statistical Process Control (SPC)
  - application programs that collect field data and automatically control the processes in real time
- Tool Control
  - application programs that provide logic orchestration through execution of tool automation scenarios
- Material Control
  - application programs that provide logic orchestration through execution of AMHS scenarios

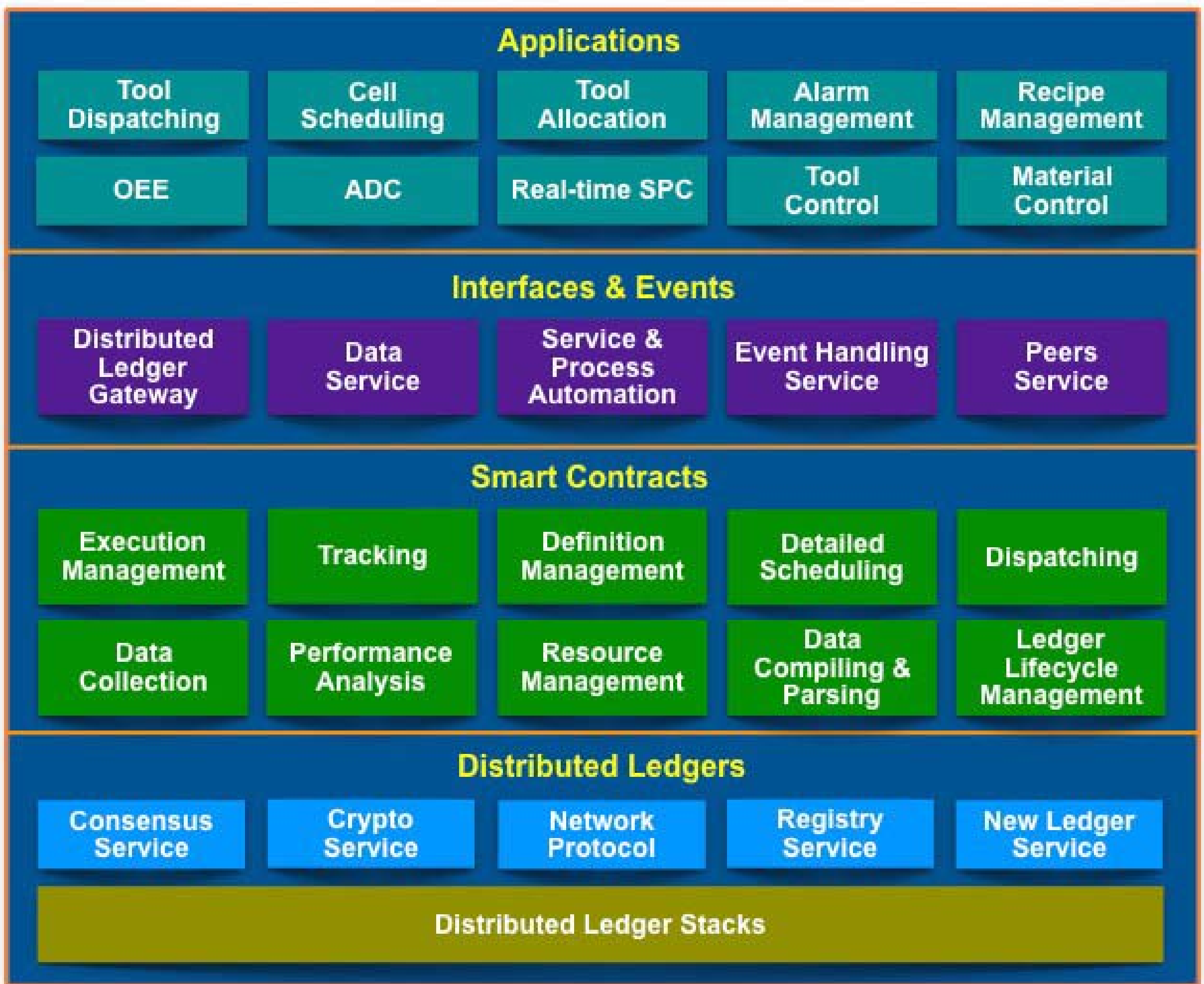

Figure 4: A Distributed-Ledger, Edge-Computing Architecture (DLECA) for Cell Controller Design

V. Distributed-Ledger, Edge-Computing Fab Automation and Computer Integration

The proposed distributed-ledger, edge-computing architecture (DLECA) adopts an enterprise, private, permissioned, peer-to-peer distributed ledger network structure. In DLECA, only certain actions can be performed by permitted participants on the network. As an enterprise, private distributed network, DLECA allows only dedicated nodes in the fab network to participate. The permitted nodes include all nodes in the Cell Automation layer and some in the Fab Automation layer. All the cell controllers form the core nodes in DLECA. Nodes of RMS, PdM, RTD, SPC, APC, and FDC servers and other fab automation servers (FAS) are included in DLECA on demand. DLECA maintains the identity of each participant on its network. DLECA is a peer-to-peer (P2P) network. Communications between any two nodes in DLECA are direct and reachable with reachability of one. The upper part of Figure 5 illustrates a complete graph representation of the DLECA peer-to-peer communications topology.

DLECA adopts the Directed Acyclic Graph (DAG) model as the data structure of distributed ledgers. The consensus mechanism under DAG demands newly added data to reference and validate the last two updated data. Such a consensus mechanism allows multiple data to be verified simultaneously. It is simpler and more flexible than other

Global Journal of Computer Science and Technology (G) Volume XX Issue III Version I Year 2020

classic techniques that validate data one at a time. In DLECA, nodes can create data at their discretions at any time. Data inconsistency is possible. Once a conflict occurs, stakeholders or preselected representatives vote to resolve.

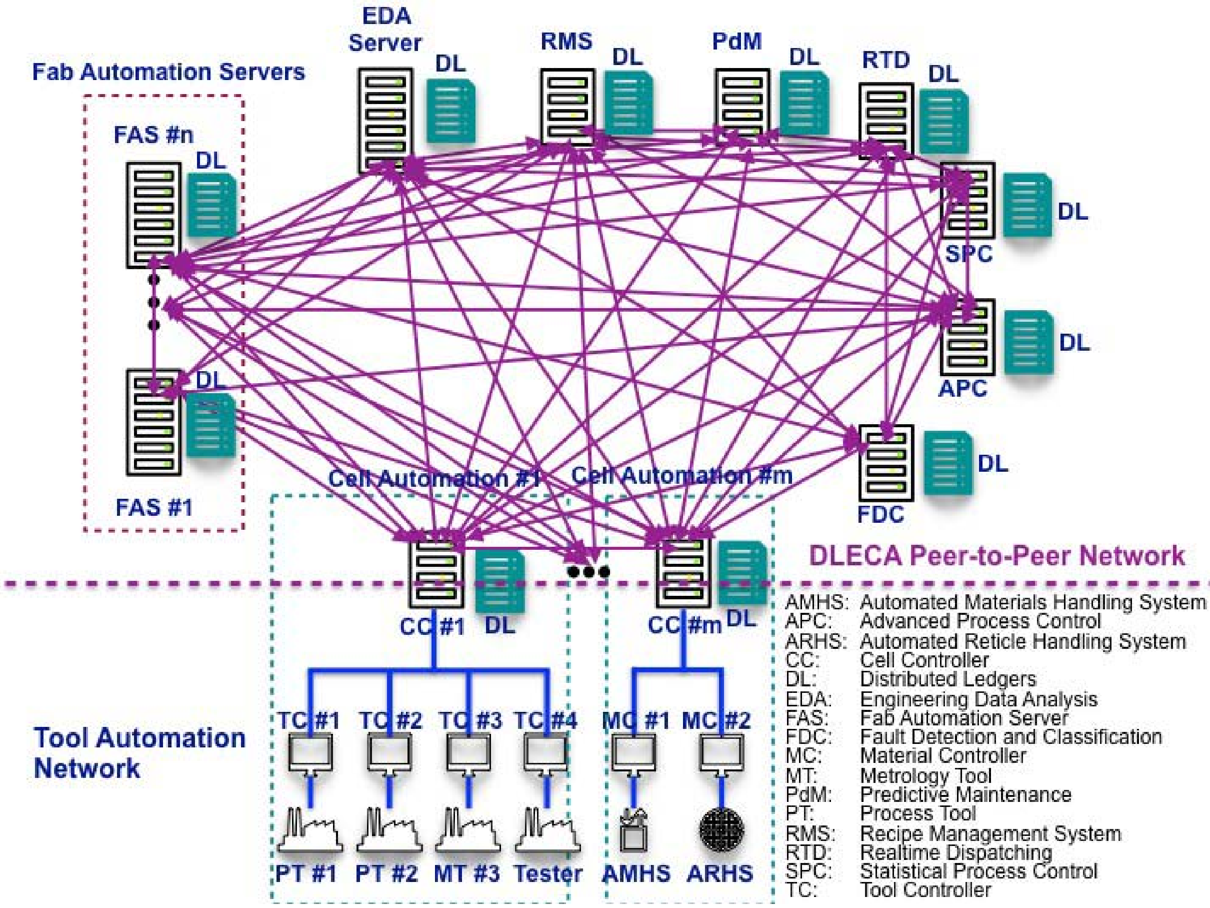

Figure 5: The DLECA Peer-to-Peer Communication Network Topology

VI. Case Study: Automation and Computer Integration for Smart Semiconductor R&D Operations

a) Semiconductor R&D Operations

Semiconductor technology research & development (R&D) is stagnating at the time close to the end of Moore's law. In the era of pushing Moore's law, both materials and equipment are lacking far behind, and product performance gets poor when new complex semiconductor structures increase. On the other hand, in the era post Moore's law, cutting edge research is still seeking opportunities to enable a new era of creativity. As customer and time-to-market pressures become escalating, expedited R&D cycle times, and enhanced learning cycles in the semiconductor industry become more and more crucial. Existing semiconductor fab environments are neither friendly nor flexible enough for R&D operations. Most designs of semiconductor fabs are for production. Even though many of them may share some space to allocate R&D-purposed equipment, their manufacturing systems are production-oriented, which are too rigid and inflexible to cope with R&D experiments and their complexity. New manufacturing systems and automated operations for semiconductor R&D are thus needed.

Compared with semiconductor production operations, activities in semiconductor R&D are more diverse and experiment-oriented. Semiconductor R&D operations have the following characteristics and requirements [46]:

- Wafers of small volume but wide variety.
- Many dedicated machines without backup.
- High production uncertainties such as frequent machine failures and tune-ups.
- Intensive engineering experiments, lot holdings and releases, process changes, inspections, and reworks.
- Complicated lot split/merge operations for the Design of Experiments (DoE) needs.
- Dynamic, on-demand, and floating data items during data collection.
- Few historical data for a new process to reference.
- Lack of baseline and baseline management.

b) *DLECA-based Semiconductor R&D Automation and Computer Integration*

We propose a design for automation and computer integration in semiconductor R&D operations, which bases on the developed distributed-ledger, edge-computer architecture (DLECA). To automate the tedious semiconductor R&D operations and integrate with computer systems, our design proposes three prominent systems. They are the systems of *R&D Workflow Management*, *R&D Data Engineering*, and *R&D Data Engineering.* The *R&D Workflow Management* system rationalizes, streamlines, and automates semiconductor repeated, tedious R&D procedures and operations. The *R&D Data Engineering* system aims to integrate and automate the processes that extract R&D data from various, different sources, then transform and finally load the data into the *Smart R&D Analytics* system that uses AI (artificial intelligence), big data and analytics techniques to accelerate R&D cycles and R&D learning cycles.

Semiconductor R&D operations generate and use massive amount of heterogeneous data distributed in different locations and computer systems. Technology data, parameters, and specifications, such as test vehicles, mask information, process/route/recipe data, and so on, are fundamental to semiconductor R&D operations and execution of R&D workflows. Metrology tools collect the metrology data at different locations, then extract, transform, and load into engineering data lakes for further analysis. The semiconductor R&D process uses a data sheet (also called *runcard*) to detail the processing parameters. We have developed four semiconductor R&D- specific applications for semiconductor R&D Cell Automation, including *Runcard Automation*, *Pilot Run Automation*, *R&D Recipe Management*, and *R&D Route Management*. To support R&D data collection, Tool Automation demands the development of R&D-specific EAPs to provide flexibility and scalability of automated data collection.

Figure 6 shows the design for DLECA-based semiconductor R&D automation and computer integration.

The developed DLECA-based semiconductor R&D automation and computer integration uses two types of R&D distributed ledgers—*R&D Specifications Ledger* and *Metrology/Engineering Ledger*. The R&D specifications ledgers store the primitive data of test vehicles, reticle information, and process/route/recipe data. The Metrology/Engineering ledgers store the generated data produced and collected from the process and metrology tools. We have developed five semiconductor R&D-specific functions for DLECA-based cell controllers. They are *R&D Route Management*, *R&D Recipe Management*, *DoE Template Management*, *Auto Split*, *R&D Runcard Management*. The R&D Route Management function deals with the split/merge operations and data collection of R&D lots. The R&D Recipe Management function controls and validates the feasibility of recipes for processing of R&D operations and experiments. The DoE Template Management function provides DoE templates and transforms templates into processing data for specifications ledgers. The Auto Split function automates inline lot split operations to replace existing engineers' operations. The R&D Runcard Management function manages R&D-specific runcards of process/route/recipe data from their creation to archive.

Figure 7 illustrates the semiconductor R&D-specific design of DLECA-based cell controllers.

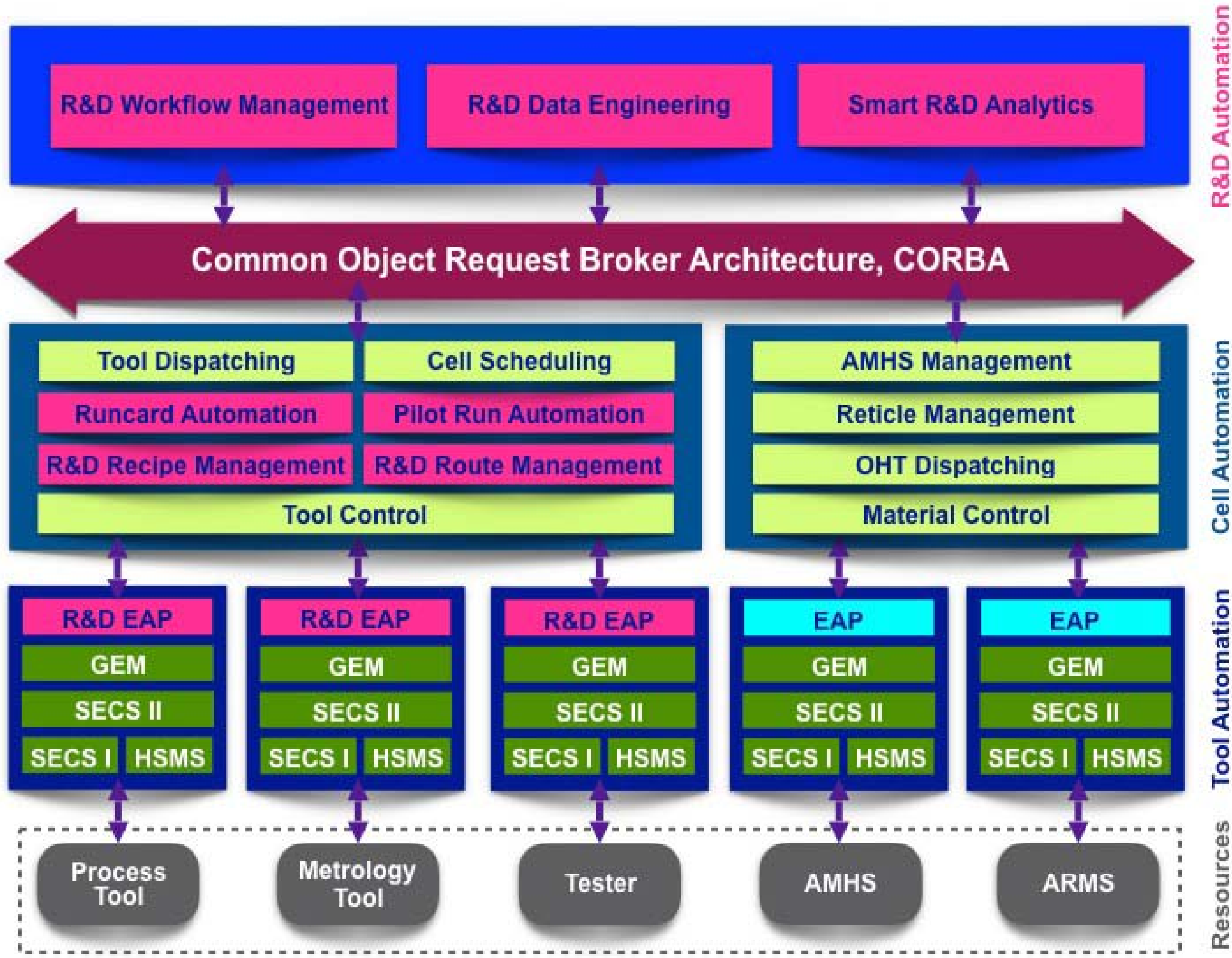

Figure 6: DLECA-based Semiconductor R&D Automation and Computer Integration

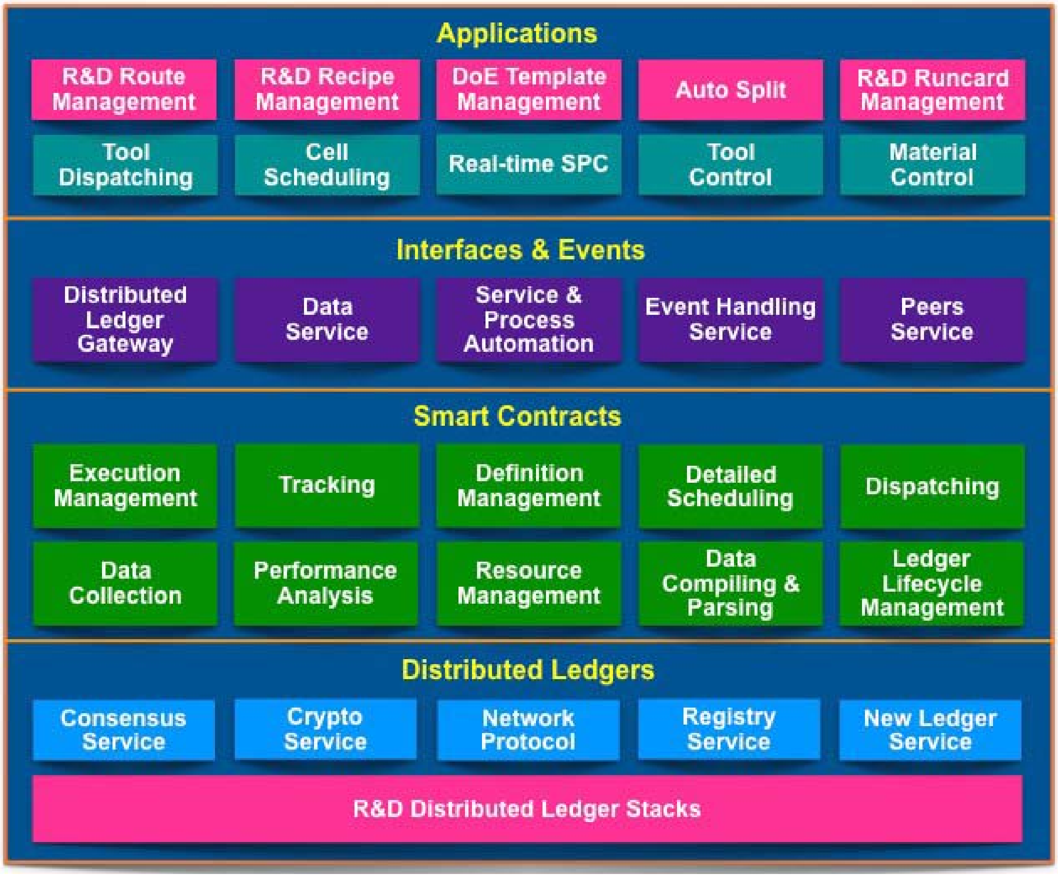

Figure 7: DLECA-based Semiconductor R&D Cell Controller Design

VII. Conclusions

The effectiveness of modern semiconductor manufacturing comes from a high level of automation and computer integration. This paper adopts the disruptive technologies of distributed ledger and edge computing to design a Distributed-Ledger, Edge-Computing Architecture (DLECA) for automation and computer integration in semiconductor manufacturing. DLECA utilizes distributed ledgers to manage the data of processing specifications, requirements, logs, states, and executable smart contracts. When a specified condition happens, smart contrasts of the distributed ledger trigger their edge computing to update state variables accordingly. The decentralization and distribution of data and their processing facilitates the overall effectiveness and efficiency of fab planning and operations management in semiconductor manufacturing. We adopt the important topic in pioneering semiconductor manufacturing for automation and computer integration of semiconductor research & development (R&D) operations as our study vehicle to demonstrate the operational structure and functionality, applications, and feasibility of the proposed DLECA software framework.

The proposed DLECA adopts a hierarchical control architecture stacked by three layers of *Tool Automation*, *Cell Automation* and *Fab Automation*. In DLECA, all the cell controllers in Cell Automation and some fab automation servers in Fab Automation form together as the distributed ledger network and perform edge computing at the edge of the fab backbone. Tool controllers in Tool Automation play a data gateway and control to one equipment only. Such hierarchical decomposition allows one cell controller to coordinate the automation of several associated tool controllers and collaborate with its distributed peers at the same time. As the computing power gets more powerful, in practice, the configuration where a cell controller and its associated tool controllers reside in a single computer is possible and cost-effective in both capital investment and system management. Compared with the traditional two-tier fab automation solutions, the three-layer architecture has more benefits on overall fab effectiveness and efficiency. The distribution and decentralization of fab CIM functionality and applications make fab automation and computer integration more flexible and more agile. Our observations show the increased flexibility and agility in the study case of automated semiconductor R&D operations, which have been considered very challenging and almost impossible. DLECA provides distributed and decentralized processing capability that can collect and process data close to the place which generates these data, and analyze and react to the analytics of the data timely. All the above expedite the R&D cycle and R&D learning.

We are implementing the proposed DLECA-based framework as part of the Manufacturing IT Architecture for Smart Semiconductor R&D Automation Program in a pioneer 300mm production fab where some of its capacity is allocated for novel technology research and development. The preliminary goal of the Program has three folds: (1) to achieve 95% of automated data collection in R&D activities; (2) to automate all the first-level data analysis; and (3) to automate all the small data analysis with the help of artificial intelligence (AI).

Instead of pushing all the legacy MES functions from Fab Automation to the edge nodes of cell controllers, DLECA only decomposes and moves the Cell-Automation-related data and applications to edge computing, such as cell scheduling and dispatching. Note that the distribution of functionalities in Fab Automation is not always feasible due to their centralized nature. For example, the execution of fab planning in MES or ERP (Enterprise Resource Planning) systems is better in a centralized approach. Further research may consider the problems of determining the optimal or reasonable degree of distribution in decomposing and integrating functionalities of semiconductor legacy centralized computer and automation systems into the DLECA-based automation and computer integration framework.

Funding: This research received no external funding.

Conflicts of Interest: The author declares no conflict of interest.

References Références Referencias

1. Hermann, M., Pentek, T., & Otto, B. (2016, January). Design principles for industrie 4.0 scenarios. In *2016 49th Hawaii International Conference on System Sciences (HICSS),* 3928-3937. IEEE.
2. Thoben, K. D., Wiesner, S., & Wuest, T. (2017). “Industrie 4.0” and smart manufacturing-a review of research issues and application examples. *International Journal of Automation Technology*, *11*(1), 4-16.
3. Kusiak, A. (2018). Smart manufacturing. *International Journal of Production Research*, 56 (1-2), 508-517.
4. Van Zant, P. (2013). *Microchip Fabrication: A Practical Guide to Semiconductor Processing: A Practical Guide to Semiconductor Processing*. McGraw Hill Professional. ISBN: 0071821015.
5. May, G. S., & Spanos, C. J. (2006). *Fundamentals of Semiconductor Manufacturing and Process Control*. John Wiley & Sons. ISBN: 0471784060.
6. Gartland, K. & Kono, S. (1999). *Automated Material Handling System (AMHS) Framework User Requirements Document*: Version 1.0. International SEMATECH.
7. Liao, D. Y., & Fu, H. S. (2004). Speedy delivery-dynamic OHT allocation and dispatching in large-scale, 300-mm AMHS management. *IEEE Robotics & Automation Magazine*, *11*(3), 22-32.
8. Liao, D. Y., & Wang, C. N. (2004). Neural-network-based delivery time estimates for prioritized 300-mm automatic material handling operations. *IEEE Transactions on Semiconductor Manufacturing*, *17*(3), 324-332.
9. Liao, D. Y. (2010, April). Automation and integration in semiconductor manufacturing. In *Semiconductor Technologies*, 39-56. InTech.
10. ITRS (2015). *International Technology Roadmap for Semiconductors 2.0*. http://www.itrs2.net/itrs-reports.html/
11. SEMI. https://www.semi.org/
12. Hawker, J. S. (1998, May). CIM framework architecture and application models. In *International Working Conference on the Design of Information Infrastructure Systems for Manufacturing*, 201-214. Springer, Boston, MA.
13. Doscher, D. (1998). Computer integrated manufacturing (CIM) framework specification version 2.0. *Technology Transfer*. Technology Transfer # 93061697J-ENG. SEMATECH.
14. Hedberg, T., Helu, M., & Sprock, T. (2018, June). A standards and technology roadmap for scalable distributed manufacturing systems. In *ASME 2018 13th International Manufacturing Science and Engineering Conference*. American Society of Mechanical Engineers Digital Collection. [CrossRef]
15. El Ioini, N., & Pahl, C. (2018, October). A review of distributed ledger technologies. In *OTM Confederated International Conferences on the Move to Meaningful Internet Systems,* 277-288. Springer, Cham.
16. Swan, M. (2015) *Blockchain: Blueprint for A New Economy*; O’Reilly: Sebastopol, CA, USA. ISBN 978-1-491-92049-7.
17. Nofer, M., Gomber, P., Hinz, O., & Schiereck, D. (2017). Blockchain. *Business & Information Systems Engineering*, *59*(3), 183-187.
18. Al-Jaroodi, J., & Mohamed, N. (2019). Blockchain in industries: A survey. *IEEE Access*, *7*, 36500-36515.
19. Szabo, N. (1997). Formalizing and Securing Relationships on Public Network. *First Monday*, *2*(2).
20. Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: vision and challenges. *IEEE Internet of Things Journal*, *3*(5), 637-646.
21. McGehee, J., Hebley, J., & Mahaffey, J. (1994). The MMST computer-integrated manufacturing system framework. *IEEE Transactions on Semiconductor Manufacturing*, 7(2), 107-116.

22. SEMATECH (1997) *Computer Integrated Manufacturing (CIM) Framework Specification*, Version 1.5, Austin, Texas.
23. Cheng, F. T., Shen, E., Deng, J. Y., & Nguyen, K. (1999). Development of a system framework for the computer-integrated manufacturing execution system: a distributed object-oriented approach. *International Journal of Computer Integrated Manufacturing*, *12*(5), 384-402.
24. Devedžić, V., & Radović, D. (1999). A framework for building intelligent manufacturing systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C-Applications and Reviews*, *29*(3), 402-419.
25. Lin, C. P., & Jeng, M. (2005). An expanded SEMATECH CIM framework for heterogeneous applications integration. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *36*(1), 76-90.
26. Lee, T. E. (2009). Semiconductor manufacturing automation. In *Springer Handbook of Automation*, 911-926. Springer, Berlin, Heidelberg.
27. Saenz de Ugarte, B., Artiba, A., & Pellerin, R. (2009). Manufacturing execution system–a literature review. *Production planning and control*, *20*(6), 525-539.
28. Sprock, T., Sharp, M., Bernstein, W. Z., Brundage, M. P., Helu, M., & Hedberg, T. (2019). Integrated Operations Management for Distributed Manufacturing. *IFAC-Papers OnLine*, *52*(13), 1820-1824.
29. Vrba, P., Tichý, P., Mařík, V., Hall, K. H., Staron, R. J., Maturana, F. P., & Kadera, P. (2010). Rockwell automation's holonic and multiagent control systems compendium. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *41*(1), 14-30.
30. Chryssolouris, G., Mavrikios, D., Papakostas, N., Mourtzis, D., Michalos, G., & Georgoulias, K. (2009). Digital manufacturing: history, perspectives, and outlook. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, *223*(5), 451-462.
31. Balta, E. C., Lin, Y., Barton, K., Tilbury, D. M., & Mao, Z. M. (2018). Production as a service: A digital manufacturing framework for optimizing utilization. *IEEE Transactions on Automation Science and Engineering*, *15*(4), 1483-1493.
32. Bratukhin, A., & Sauter, T. (2011). Functional analysis of manufacturing execution system distribution. *IEEE Transactions on Industrial Informatics*, *7*(4), 740-749.
33. Vyatkin, V. (2007). IEC 61499 Function Blocks For Embedded and Distributed Control Systems Design. *ISA/O3neida, USA*.
34. Zoitl, A., & Lewis, R. (2014). *Modelling Control Systems Using IEC 61499* (95). IET. ISBN 978-1-84919-760-1.
35. Foehr, M., Vollmar, J., Calà, A., Leitão, P., Karnouskos, S., & Colombo, A. W. (2017). Engineering of next generation cyber-physical automation system architectures. In *Multi-Disciplinary Engineering for Cyber-Physical Production Systems* (pp. 185-206). Springer, Cham.
36. Dotoli, M., Fay, A., Miśkowicz, M., & Seatzu, C. (2019). An overview of current technologies and emerging trends in factory automation. *International Journal of Production Research*, *57*(15-16), 5047-5067.
37. El Ioini, N., & Pahl, C. (2018, October). A review of distributed ledger technologies. In *OTM Confederated International Conferences on the Move to Meaningful Internet Systems*, 277-288. Springer, Cham.
38. Iansiti, M.; Lakhani, K. R. (2017) The truth about blockchain. *Harvard Business Review*, *95*(1), 118-127.
39. Nakamoto, S. (2008). A peer-to-peer electronic cash system. *Bitcoin*. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 12 February, 2020).
40. Buterin, V. (2014). A next-generation smart contract and decentralized application platform. *white paper*. Available online: https://cryptorating.eu/whitepapers/Ethereum/Ethereum_white_paper.pdf (accessed on 12 February, 2020)
41. Fernández-Caramés, T. M., & Fraga-Lamas, P. (2019). A review on the application of blockchain to the next generation of cybersecure industry 4.0 smart factories. *IEEE Access*, *7*, 45201-45218.
42. Khan, A. T., Cao, X., Li, S., & Milosevic, Z. (2019). Blockchain Technology with Applications to Distributed Control and Cooperative Robotics: A Survey. *International Journal of Robotics and Control*, 2(1), 36-48.
43. Stanciu, A. (2017, May). Blockchain based distributed control system for edge computing. In *2017 21st International Conference on Control Systems and Computer Science (CSCS)*, 667-671. IEEE.
44. Isaja, M., Soldatos, J., & Gezer, V. (2017, November). Combining edge computing and blockchains for flexibility and performance in industrial automation. In *International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*.
45. Isaja, M., & Soldatos, J. K. (2018). Distributed ledger architecture for automation, analytics and simulation in industrial environments. *IFAC-PapersOnLine*, *51*(11), 370-375.
46. Liao, D. Y., Chang, S. C., Pei, K. W., & Chang, C. M. (1996). Daily scheduling for R&D semiconductor fabrication. *IEEE transactions on semiconductor manufacturing*, *9*(4), 550-561.