# Ready configuration of machines into an existing manufacturing system

Covanich, W. , McFarlane, D. , Brusey, J. and Farid, A.M.

# READY CONFIGURATION OF MACHINES INTO AN EXISTING MANUFACTURING SYSTEM

**W. Covanich*, D. McFarlane*, J. Brusey*, A.M. Farid***

*Institute for Manufacturing, University of Cambridge, UK, CB2 1RX. contact email: wc226@cam.ac.uk

**Keywords:** Flexible manufacturing systems, manufacturing system integration.

## Abstract

This paper focuses on simplifying and easing the integration of a new machine into an existing conventional hierarchical manufacturing system. Based on a distributed manufacturing paradigm, it proposes the functions and interfaces that a new machine and an existing manufacturing system should possess so that ready and simple configuration of additional machines can be achieved. The configuration process is intended to include not only mechanical and electrical interfaces but also decision system interfaces too (such as planning, scheduling and shop floor control). The preliminary experiments to compare the reconfigurability resulting from a conventional integration method and the proposed distributed method are presented. The results are then discussed.

## 1 Introduction

Manufacturing practices in the future will have to cope with customers demanding low cost products whose needs are likely to change quickly. Hence, manufacturing operations will have to be organized differently and be more effective in responding. As a consequence, in the last 10 years, many designs and trial-implementations of distributed manufacturing systems have been reported in the literature [12]. One of the key properties of the manufacturing system which can react to changes rapidly and cost effectively is reconfigurability [11]. The term reconfigurability can be defined as the ability of a manufacturing system to be simply altered in a timely and cost effective manner [9]. Although, it is believed that by applying distributed manufacturing system solutions, the reconfigurability, the responsiveness, and the performance of the manufacturing systems can be improved, only a few of the proposed distributed manufacturing systems have resulted in any industrial take up. The lack of the adoption may be because of a shortage of evaluations and comparisons of the resulting designs to the conventional approaches [4]. The migration strategy to enable existing manufacturing systems which use conventional controller technology to progressively incorporate distributed manufacturing concepts is also required [10].

To address these problems, an approach based on a distributed manufacturing system paradigm is proposed for integrating new conventional machines into an existing conventional manufacturing system. It is expected that this approach should be able to simplify and ease the process of integrating new machines into an existing manufacturing system. In addition, the method should be able to be used to incrementally convert existing conventional manufacturing systems with conventional controller technology into distributed manufacturing systems.

The reconfiguration of a manufacturing system can be categorized into three types of operations: addition of new components, removal of the existing components, or modifying the existing components. Note that manufacturing components can be physical components, such as machines, or logical components such as control software. This paper only focuses on the addition of new machines into an existing manufacturing system. The other two cases of reconfiguration are not considered here.

The paper begins in section 2 by reviewing: the existing integration approaches, distributed manufacturing paradigms, and the approaches that can be used to evaluate reconfigurability of the manufacturing systems. Section 3 presents the proposed method. The implementation of the proposed method in Cambridge Distributed Information and Automation Laboratory (DIAL) is described in section 4. The results are discussed in section 5. Finally section 6 summarizes the paper.

## 2 Background

This section gives a brief review if the previous works relating to the integration of new machines into an existing manufacturing system. The evaluation methods can be used to evaluate the reconfigurability of the manufacturing systems are also discussed in this section. This section begins by giving examples of the integration solutions used in the computer domain. The approaches that can be used to solve the problem of integrating new machines into a conventional manufacturing system are discussed next. Finally, the existing reconfigurability evaluation tools are discussed.

### 2.1 Integration in computer system domain

This section presents reconfiguration problems in the computer domain. The solutions for the reconfiguration problems are also discussed. An example of the reconfiguration of an individual computer is to attach a new peripheral device (such as a printer or a scanner) to the computer. In some cases, an existing peripheral may need to be removed. Since this paper only focuses on adding new machines into an existing conventional manufacturing system, the problem of modifying or removing the existing components will not be discussed here.

Reconfiguration in the computer domain can occur in both individual computers and computer networks. An example for the first case is to attach a new device to a computer. In this case, each computer peripheral is usually different and the way each peripheral can be controlled is normally dissimilar. To be able to integrate the new device into a computer, Operating Systems (OS) essentially dictate how every computer peripherals should be controlled [16]. The "device driver" is used to create a standard interface between OS and a specific computer peripheral. This device driver translates the generalized command from the OS into a specific command used to control a particular device [16]. Thus, the OS only need to communicate with the device driver. The driver can be considered as a wrapper that provides a standard interface between the OS and a particular device.

For the computer network case, a new computer may need to be added to the existing network. This new computer may have a different communication interface. Thus, it is not possible to connect the computer directly into the computer network. In this case, a network bridge can be used to connect the new computer into the network [20]. The network bridge can be used to connect two different networks with different interfaces at the data link layer (layer 2) of the OSI model. In summary, a network bridge can be considered as an intermediate component which acts as an interface between two computer networks.

### 2.2 Integration in conventional manufacturing systems

Examples of the problems needed to be solved in order to be able to integrate a new machine into an existing conventional manufacturing system are: to create a communication link between the centralized controller and the machine, and to define how the new machine would be controlled by the centralized controller. The problem of establishing a communication link between the centralized controller and the new machine can be solved by using a standard communication interface and communication protocol. OLE for Process Control (OPC) [17], and Manufacturing Automation Protocol (MAP) [19] can be used as the standard communication interface and the communication protocol respectively. ISA S88 [1] provides guideline for designing the manufacturing systems. It can be used to define how the centralized controller controls the subordinate machines. Although it is intended to be used in batch manufacturing systems, it can also be used in discrete and continuous manufacturing systems [2]. One of the most important features of the ISA S88 standard is the separation of the production plan (recipe) from equipment control. This dramatically reduces the time required to modify the control program [2].

### 2.3 Distributed manufacturing paradigm

To improve the reconfigurability and responsiveness of the manufacturing systems, many distributed control architectures are proposed [15]. It is expected that the components in distributed manufacturing systems (such as machines) should be able to be removed or added easily. This follows from the changeability property (modularity + decentralized) of the distributed system [18]. This property is shown in Figure 1.

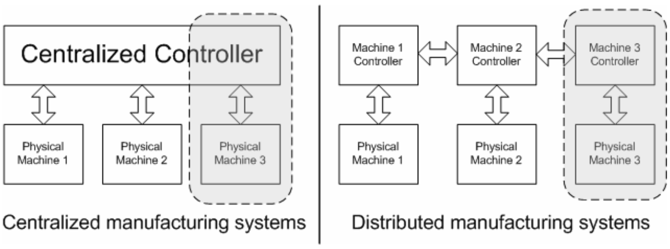


Figure 1: Centralized manufacturing systems and distributed manufacturing systems

From Figure 1, if machine 3 is to be added or removed from the centralized manufacturing system, the centralized controller must be modified. If many machines are controlled by one centralized controller, the control program would be very complex. In this case, a lot of time and effort would be required if the manufacturing system is to be reconfigured [18]. On the other hand, if machine 3 is to be added or removed from the distributed manufacturing system, ideally there will be no need to modify the other machines at all, since all the functions required to control machine 3 are embedded within the machine 3 controller. This eases the process of reconfiguring the manufacturing system [18]. Although it is expected that distributed manufacturing systems should be more reconfigurable, creating the whole new distributed control system to replace an existing centralized control system would require too much time and effort. Thus, a migration approach that minimizes the modification require to be made to the existing centralized manufacturing system is required.

### 2.4 Evaluating reconfigurability of a manufacturing system

As mentioned in section 1, although the distributed manufacturing systems are expected to be more reconfigurable and responsive, only few of the proposed distributed manufacturing systems have resulted in any industrial take up. One of the reasons for the lack of the adoption may be because of a shortage in the evaluations and comparisons between the resulting designs and the conventional approaches [4]. Thus, in order to provide rigid evidence to show that distributed manufacturing systems are more reconfigurable than centralized manufacturing systems, reconfigurability of both types of manufacturing systems should be evaluated and compared. Chirn [4] evaluated the reconfigurability of the manufacturing systems by calculating strategic complexity of the control system software, operational complexity of the control system software, extension rate, and reuse rate. The strategic complexity of the control system software indicates the level of complexity of the control system in design phase, while the operational complexity of the control system software is used to evaluate the level of complexity of the control system in the implementation phase. The extension rate represents the growth rate of the scale or complexity of new scenario compared with that of the existing scenario. Reuse rate is defined as the percentage of the existing design or codes used in a new scenario. The method mainly focuses on evaluating the complexity of the internal structure of the controller but does not focus on interfaces or the capability of the manufacturing system. Structural complexity of the software can also be evaluated using cyclomatic complexity [13].

Recently, Farid [7;8] proposed an approach for evaluating reconfigurability of the manufacturing systems. The Design Structure Matrix (DSM) is used to evaluate the ease of reconfiguration of the manufacturing systems [7]. The DSM is used to capture the interfaces between modules within the manufacturing systems. The ease of the reconfiguration is then evaluated based on the interfaces captured by the DSM. The potential of reconfiguration is measured by the use of Production Degrees of Freedom for manufacturing systems (DOF) [8]. The production DOF captures all the capabilities that can be physically provided by the manufacturing systems and the constraints that make the number of the capabilities less than ideal. The production DOF can be categorized into two classes. The first class, scleronomic production DOF, is independent of the sequence of the production operation. This class of DOF can further be classified into two subclasses; transformation scleronomic DOF and transportation scleronomic DOF. The transportation scleronomic DOF can be calculated by simply counting the available transformation processes. The transportation rheonomic DOF can also be calculated by counting the number of the available transportation processes. The scleronomic production DOF can be used to capture all the production processes the manufacturing system can physically perform. The constraints that make some production processes unperformable can also be captured using the constraints matrix.

The second class of the production DOF, rheonomic production DOF, is used to capture all the feasible two concatenated production operations. It can be calculated by counting all the possible two concatenated production operations. The reheonomic production DOF can be further divided into four subclasses, which are defined, based on the class of the two concatenated production operation. The four subclasses are; two successive transformation processes (Type I), a transformation process followed by a transportation process (Type II), a transportation process followed by a transformation process (Type III), and two successive transportation processes (Type IV). The constraint that prohibits a particular sequence of operations can be captured using the constraint matrix. This matrix can be used to identify the limitations of the control system of the manufacturing system. Thus, it gives a guideline to how the control system of the manufacturing system can be improved. However, the complexity or effort required to configure, create or modify the manufacturing components or module is not considered. The combination of all the reviewed evaluation methods should be able to capture most of the effort required to reconfigure a manufacturing system.

## 3 The proposed method

As mentioned in section 2.3, to promote the use of the distributed manufacturing systems in real industrial factory, a migration approach, which minimizes the modification required to be made to the existing centralized manufacturing system is required. By using the distributed manufacturing solutions, it is expected that the benefits of the distributed control structure, such as improving changeability (and thus reconfigurability) should be achieved. Before presenting the proposed method, a fundamental concept used in the proposed method will be discussed. From the functionality point of view a machine in a fully distributed manufacturing system can be considered as a miniaturized factory as shown in Figure 2. A conventional manufacturing control hierarchy is shown in fig. 2a. Each function (planning, scheduling, manufacturing order release, manufacturing control, and device operation) is performed at a different level (and thus different component) in the factory (see fig. 2a). However, in a complete distributed manufacturing system, a machine should possess all these functions [14]. Thus it can be considered as a miniaturized factory. The only difference is: a factory can perform many physical production operations and produce complete products. However, a machine may be able to perform only a small number of production operations and produce part of a product.

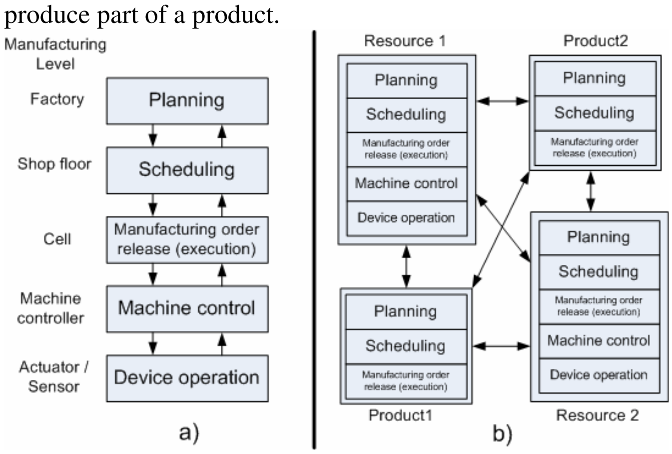


Figure 2: a) Conventional centralized control approach and b) Distributed solutions [14]

Based on the wrapper concept (see section 2) and the distributed manufacturing paradigm (see section 2), an approach for integrating a new machine into an existing conventional hierarchical manufacturing system is proposed. This approach can be used as a first step to convert a conventional manufacturing system into a distributed manufacturing system. The proposed method comprises of three main steps: (i) convert an existing conventional manufacturing system into an intelligent machine, (ii) convert a new machine into an intelligent machine, and (iii) system integration.

Since many distributed manufacturing systems have been proposed in the literature, candidate architectures to be used in the proposed method for this project should be selected from one of the available architecture. In this project the HCBA (Holonic Component Based Architecture) [3] was selected for the following reasons. First, it has been implemented in a real physical manufacturing system. Second, it specifically focused on improving the reconfigurability of the manufacturing systems. However, it is expected that any distributed architecture should be able to be used. Having defined the distributed architecture to be used, the three main steps for the integration process are described next.

### 3.1 Step 1: Convert an existing manufacturing system into a HCBA resource

HCBA is comprised of two fundamental classes of manufacturing objects; resources and products. The resources are machines in the manufacturing systems. They are to perform production operations required in order to produce products. The resources are comprised of physical machine part, which perform the physical production operations, and intelligent software part, which control the behaviour of the machine. HCBA products represent the products to be produced by the manufacturing system. They are comprised of physical parts, which are the parts to be produced, and the intelligent software part. The intelligent software part is used to monitor the status of the physical parts to be produced. It is also used to assign tasks to the associated resources via negotiation. As mentioned earlier in this section, a machine in a distributed manufacturing system possesses all the required functions in order to be able to perform its tasks and can be considered in some sense as a miniaturized factory. Thus, a conventional manufacturing system, which possesses all the functions required to produce a product, can also be considered as a miniaturized factory and should be able to be converted into a single intelligent resource. If some of the required functions are missing (such as scheduling) then the missing functions must be added to the conventional manufacturing system before it can be considered as a miniaturized factory and converted into an intelligent resource.

Thus in order to convert an existing conventional manufacturing system into an intelligent machine, all the functions the distributed intelligent machine (or a miniaturized factory) should possess must be defined first. The functions the intelligent machine should possess are planning, scheduling, manufacturing release, machine control, and device operation as shown in Figure 2a. In addition, it should also possess the ability to communicate with other manufacturing entities. In the real world, a factory must also be able to communicate with other factories or other functional units within the enterprise, since it must receive raw material, production orders, etc. from other entities and send the complete products, production statuses, etc. to other entities.

Having described the concept of how an existing manufacturing system can be converted into an intelligent machine, the conversion process will be described next. To convert an existing manufacturing system into an intelligent machine, first the existing manufacturing system must be analyzed and the part of the existing system to be wrapped must be defined. It may be possible to create a wrapper for the whole existing factory but this would lead to a very complex intelligent machine. The part of the existing system to be wrapped can be found by identifying the associated centralized controller. This controller will be used to control the new machine to be added, if the existing hierarchical architecture is used. In addition, the capabilities of the manufacturing system to be wrapped may need to be modified so that it can coordinate with the new machine to be added (see section 4). It must be ensured that the redefined capabilities can be provided solely by the existing manufacturing system itself (see section 4). The capabilities should be able to be matched with the language used to describe the production plans defined in the intelligent software part of the intelligent product. The next step is to compare the functions of the existing manufacturing system to be wrapped with those of the HCBA resource (intelligent machine) and identify the missing functions. Note that the wrapped existing manufacturing system will be considered as a single aggregated HCBA resource. After all the missing functions have been identified, intra-resource interfaces and inter-resource interfaces can be defined. Finally, a wrapper, which provides the missing functions and interfaces, is created and wrapped around the existing manufacturing system as shown in Figure 3.

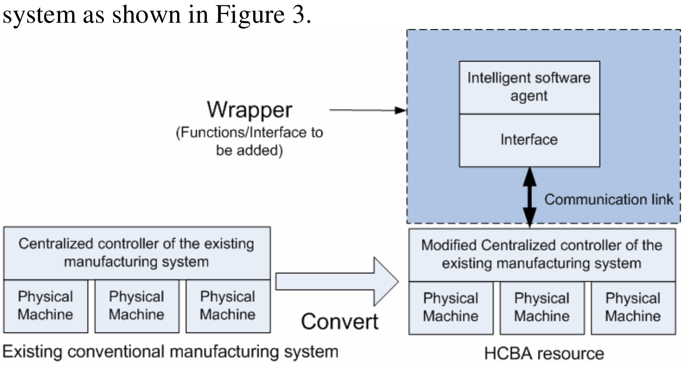


Figure 3: Converting an existing cell into a HCBA resource

### 3.2 Step 2: Convert a new machine into a HCBA resource

The second step is to convert a machine into a HCBA resource. This can be done by: first, comparing the functions of the machine to those of the HCBA resource and identifying the missing functions. After all the missing functions have been identified, intra-resource interfaces and inter-resource interfaces can be defined. The next step is to define the set of the capabilities will be provided by the machine. These capabilities should be able to be executed individually by this machine. The capabilities should also be able to be matched with the language used to describe the production plan defined in the intelligent product. Finally, a wrapper, which provides the missing functions and interfaces, is created and wrapped around the machine as shown in fig. 4.

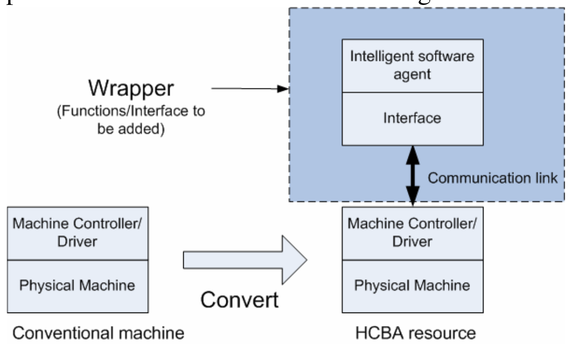


Figure 4: Converting a machine into HCBA resource

### 3.3 Step 3: System integration

This step includes creating an intelligent product (one for each type of the product to be produced) and an interface to the higher level of the manufacturing hierarchy. The intelligent product is used to coordinate the production operations provided by the resources. It has all the necessary information required to produce a product. The product is used to perform horizontal integration between resources. An interface for interfacing the new manufacturing system (the combination of a HCBA resource of the new machine, a HCBA resource of the existing manufacturing system and the intelligent product) to the higher level of the manufacturing hierarchy is also created. This interface is used to receive order/command to produce products from the higher level and send request to the associated intelligent product. The resulting integrated manufacturing system is shown in Figure 5.

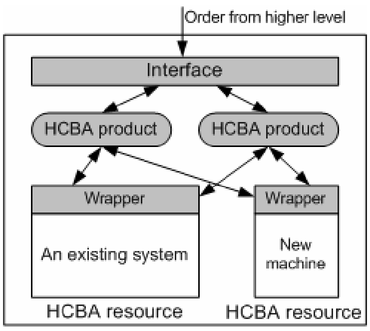


Figure 5: The integrated manufacturing system

## 4 Experimental studies

To evaluate and compare the reconfigurability between the proposed method, the centralized architecture, and the full HCBA architecture, all the architecture is implemented in a real physical manufacturing system in the Cambridge Distributed Information and Automation Laboratory. The experimental manufacturing cell is used to pack gift boxes. The picture, the layout of the laboratory, and the gift box are shown in Figure 6, 7 and 8 respectively.

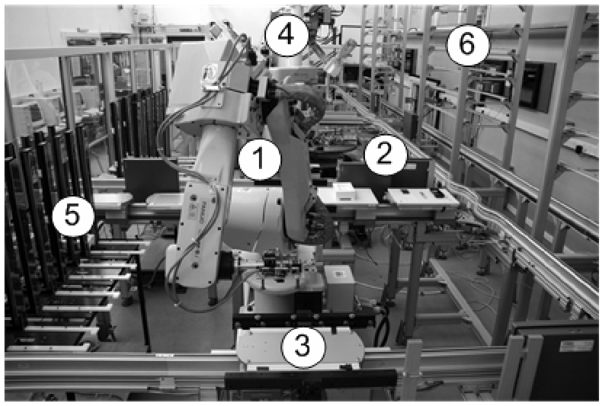


Figure 6: Picture of Cambridge Distributed Information and Automation Laboratory showing robot3 (1) shuttles (2), docking station1 (3), robot4 (4), buffer3 (5), and shelf 2 (6)

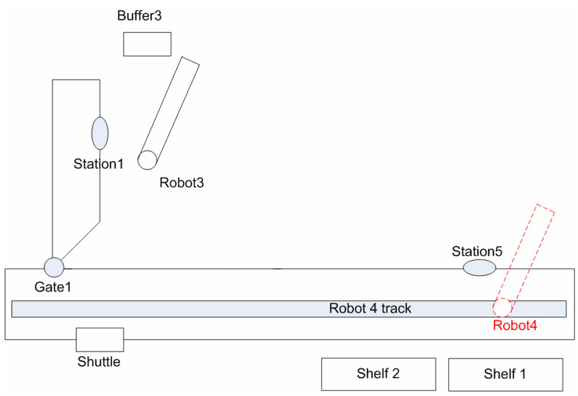


Figure 7: Lay out of the experimental test bed

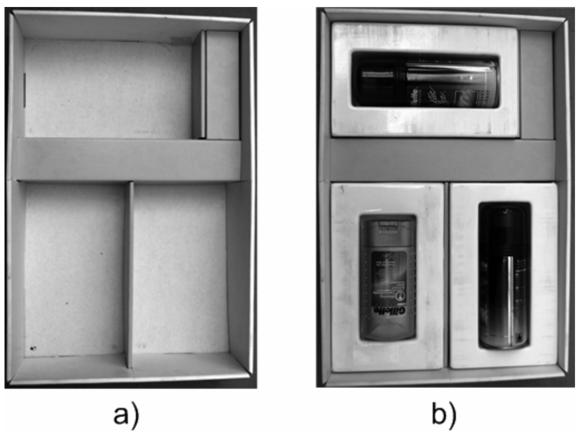


Figure 8a) an empty box
8b) a packed gift box

## 4.1 Test scenarios

This project focuses on a specific type of reconfiguration, adding new machines to an existing manufacturing system. Thus, the test scenarios are designed so that the reconfigurability (effort required to add the new machines), if the different architecture is used, can be compared. The experiments comprises of two test scenarios. The first test scenario is the initial configuration of the manufacturing system. The layout of the test bed for this test scenario is that as shown in Figure 7. The only difference is that it is assumed that there is no robot 4. In test scenario 2, robot 4 is added so that gift boxes can be automatically moved between docking station 5, shelf 1, and shelf 2. Thus, in this experiment, robot 4 is the new machine to be integrated into the existing manufacturing system. Test scenario 1 serves as the existing manufacturing system, while test scenario 2 is the manufacturing system after the reconfiguration. The process plan for each test scenario1 is described below.

*Process plan for test scenario1*
Initial condition: A shuttle with an empty box is in docking station 5.
Sequence of operation

1. Docking station 5 releases the shuttle
2. When the shuttle arrives at gate 1, gate 1 directs the shuttle to docking station 1.
3. When the shuttle arrives at docking station 1, docking station 1 clamps.
4. Robot 3 picks part A and places in slot 1.
5. Robot 3 picks part B and places in slot 2.
6. Robot 3 picks part C and places in slot 3.
7. Docking station 1 unclamps.
8. Docking station 1 releases the shuttle.
9. When the shuttle is detected at gate 1, gate 1 directs the shuttle to docking station 5.
10. The shuttle arrives at docking station 5.

*Process plan for test scenario 2*
Initial condition: An empty box is on shelf1 and an empty shuttle is in docking station 5.
Sequence of operation

1. Robot 4 picks an empty box from shelf 1 and places the box on the shuttle in docking station 5.
2. Docking station 5 releases the shuttle
3. When the shuttle arrives at gate 1, gate 1 directs the shuttle to docking station 1.
4. When the shuttle arrives at docking station 1, docking station 1 clamps.
5. Robot 3 picks part A and places in slot 1.
6. Robot 3 picks part B and places in slot 2.
7. Robot 3 picks part C and places in slot 3.
8. Docking station 1 unclamps.
9. Docking station 1 releases the shuttle.
10. When the shuttle is detected at gate 1, gate 1 directs the shuttle to docking station 5.
11. The shuttle arrives at docking station 5.
12. Robot 4 picks the packs box from the shuttle in docking station 5 and places the box on shelf 2.

Note that because of the addition of robot 4, two new steps (step 1 and step 12) are added to the process plan of test scenario 2.

## 4.2 Common modules implementation

From Figure 1, it can be seen that the major difference between a centralized manufacturing system and a distributed manufacturing system is that in the former system, there exists a central controller, which dictates the actions of its subordinates. However, at the machine real-time controller level and the physical machine level, the two systems are almost the same. It is common nowadays that a machine together with its real-time controller is provided directly by machine providers. Thus, the implementation of each architecture (centralized, the proposed method, and HCBA) at the real-time controller and physical machine level in this project will be the same. The major difference is how the machines are coordinated (either by having a centralized controller to dictate the actions of all machines or the machines cooperating with other machines in order to produce a product). The machine real-time controller, physical machines, and its interfaces are designed for each class of machine. The combination of the machine real-time controller, physical machine and its interface will be referred to as a machine module. All classes of the machine modules for the test bed are shown below.

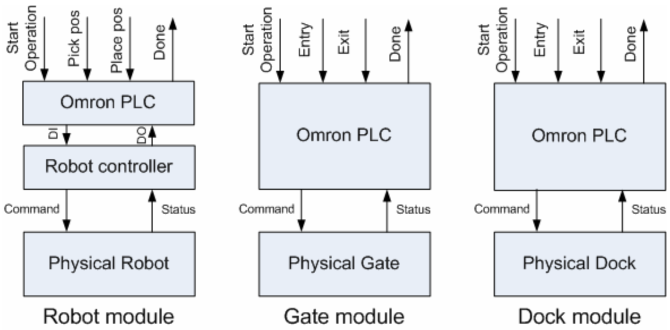


Figure 9: Robot module, gate module, and dock module

From Figure 9, there are three classes of the machine module; robot module, gate module and dock module. The robot module is comprised of Omron PLC, robot controller and physical robot. The Omron PLC is used to standardize the interface of the robot module. The robot controller is used to control and synchronize the movement of the links of the robot in real time. The gate module and dock module are comprised of Omron PLC and the physical machines. The Omron PLC is used as a machine controller. It also acts as an interface to the higher-level controller. A Petri-net is used to model the control logic in Omron PLC for all modules. The Petri-net is then converted into ladder logic program and uploaded to the Omron PLC. Note that the modules will be used in all of the architectures.

### 4.3 Implementation of the centralized (conventional) manufacturing system

As mention in section 4.2, the major difference between a centralized manufacturing system and a distributed manufacturing system is how the machines in the manufacturing system are coordinated. In the case of centralized control architecture, there will be only one single decision node that performs all planning and information processing functions. In this test scenario, Omron PLC is used as a centralized controller, which coordinates and dictates all the operations of its subordinate machines. A Petri-net is used to represent the model of the control logic for the centralized controller. The model is converted into ladder logic program and uploaded to the PLC. All the modules are connected to the centralized PLC as shown in Figure 10 and Figure 11.

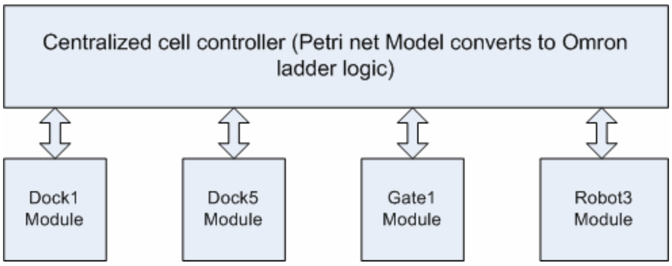


Figure 10: The implementation of the centralized manufacturing system for test scenario1

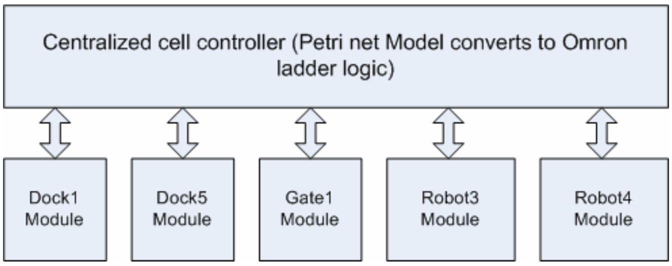


Figure 11: The implementation of the centralized manufacturing system for test scenario2

### 4.4 Implementation of the distributed manufacturing system (HCBA)

The process of developing a HCBA based manufacturing systems is comprised of four main steps; infrastructure design, resource design (intelligent machine), product design (intelligent product), and system integration [3]. In the first step, infrastructure design, the interfaces (both intra-machine and inter-machine), and the internal structure of the machine and product are defined. The interfaces and the internal structure are based on those defined in the original HCBA [3]. However, the blackboard interface is improved so that the occurred event will be pushed to the intelligent software. The internal structure for the intelligent machine and product is the same except that the real-time controller module is not included in the product. The structure of the HCBA machine is shown below.

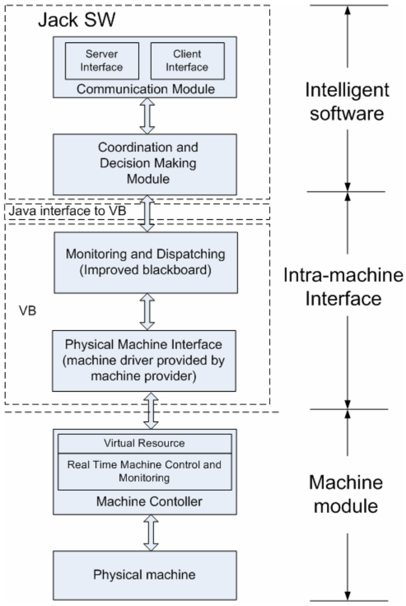


Figure 12: The internal structure of the HCBA machine

The HCBA machine (and product) is comprised of three main components, which are physical machine module, intra-machine interface, and intelligent software part. The physical machine module is the combination of the physical machine and its real-time controller. The common module described in section 4.2 can be used as a physical machine module. The intra-machine interface acts as an interface between intelligent software module and physical machine module. Visual Basic is used to create this module. The intelligent software part makes a decision about what the machine will do. It creates its own schedule by negotiating with the HCBA product. In this experiment, the HCBA machine will provide service to the first product sending a request to it provided that it is available. The intelligent software is created by the using JACK agent platform [6].

The next step is to define the functionality of the HCBA resource (machine). In order to define the functionality of the HCBA resources, the resources within the manufacturing system must first be identified. For this experiment, the resources are gates, robots, docking stations, and shuttles. After all resources have been identified, the functionality of each resource can be identified by defining the production operations that can be done by each resource. Note that the defined production operations must be able to be matched to the process plan. An example of the capability table of the robot module is shown below.

| Resource | Capability | Parameters |
|---|---|---|
| Robot | Pick and Place | Part type, Pick location, Place location |

Table 1: Capability table of the robot

The third step is to define the intelligent product and its process plans. The process plans are based on those defined in section 4.1. The final step is to perform the integration and operate the manufacturing system.

### 4.5 Implementation of the proposed method

The proposed method will only be implementable, if there already is a complete conventional manufacturing system. Because of this, the centralized manufacturing system implemented in test scenario 1 will be used as an existing manufacturing system. The proposed method is only implemented in test scenario 2. Robot 4 is a new machine to be added to the existing centralized manufacturing system. Three main steps for adding robot 4 using the proposed method are; (i) convert an existing manufacturing system into a HCBA resource, (ii) convert a machine into a HCBA resource, and (iii) system integration.

To convert an existing manufacturing system into a HCBA resource, first the existing system must be analyzed. The objectives of the analyzing process are to identify the part of the manufacturing system to be wrapped, to check whether the existing process plan needs to be modified, and to define the capabilities of the converted machine. It can be seen from the process plans of test scenario 1 and test scenario 2 that in order to integrate robot 4 into the manufacturing system, two new production steps must be added. These two new production steps are step 1 and step 12 of the process plan of test scenario 2. There is no need to split, rearrange, or modify the existing process plan (process plan of test scenario 1). The two new production steps can be added in the beginning and in the end of the existing plan as show in Figure 13.

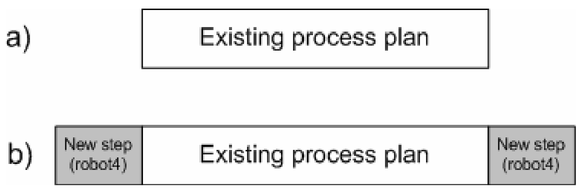


Figure 13a) the old process plan
13b) the new process plan

The part of the existing system to be wrapped can be found by identifying the associated centralized controller. This controller will be used to control the new machine if the existing centralized architecture is used. Next, the functions of the existing system are compared to those of the HCBA machine and the missing functions are added. Finally, the capabilities of the existing manufacturing system are defined. The whole existing process plan (the test scenario 1 process plan) can be defined as a capability “pack the box”. This can be the only capability of the converted HCBA machine. However, splitting the capability “pack the box” into smaller capabilities may ease the reconfiguration process in the future.

The second step is to convert robot 4 into a HCBA resource. This process is the same as that described in section 4.4. Finally, the intelligent product is introduced. The product has the information about the process plan. However, since the old process plan is considered as one complex production step, the new process plan can be considered as three steps production process as shown below.

| Step | Operation | Equivalent to operation in the old process plan |
|---|---|---|
| 1 | Robot 4:Pick box from shelf1/place on the shuttle in dock5 | |
| 2.1 | | Dock 5:Release |
| 2.2 | | Gate 1:Connect path from main loop to station1 |
| 2.3 | | Dock 1:Clamp |
| 2.4 | | Robot 3:Picks part A and place in slot 1 |
| 2.5 | | Robot 3:Picks part B and place in slot 2 |
| 2.6 | | Robot 3:Picks part C and place in slot 3 |
| 2.7 | | Dock 1:Unclamp |
| 2.8 | | Dock 1:Release |
| 2.9 | Existing cell : Pack the box | Gate 1:Connect path from station1 to mainloop |
| 3 | Robt 4:Picks a box and place on shelf 2 | |

Table 2: the process plan for the proposed method in test scenario 2

## 5 Evaluation and results

This section evaluates and compares the reconfigurability between the proposed method, the centralized architecture, and the full HCBA architecture. The reconfigurability of all architecture is compared by comparing their production degrees of freedom [8], cyclomatic complexity [13], strategic complexity [4], and extension rate [4]. It is expected that the DSM will be used in future work.

### 5.1 Degrees of Freedom Comparison

The production DOF can be used to capture the number of all processes the manufacturing system can physically perform. It can also be used to capture the constraints that make some production operations infeasible. The DOF of the experimental manufacturing systems is shown below.

| Test | Architecture | Possible transf. DOF | Constraints | transf. DOF |
|---|---|---|---|---|
| 1 | All | 1 | 0 | 1 |
| 2 | All | 2 | 0 | 2 |

Table 3: Sceleronomic production transportation DOF

| Test | Architecture | Possible transp. DOF | Constraints | transp. DOF |
|---|---|---|---|---|
| 1 | All | 4 | 0 | 4 |
| 2 | All | 10 | 2 | 8 |

Table 4: Sceleronomic production transportation DOF

| Test | Architecture | Possible Rh. DOF | Constraints | Rh. DOF |
|---|---|---|---|---|
| 1 | All | 13 | 10 | 3 |
| 2 | All | 21 | 15 | 6 |

Table 5: rheonomic production DOF

From table 3, 4, and 5 it can be seen that the number of the physically possible production DOF for all architecture in the same test scenario is the same. This is because this number represents the physical capabilities of the manufacturing system. Since the physical machines used in all architecture in the same test scenario are the same, the number of the possible production DOF is the same. However, it is expected that the distributed manufacturing systems should have less constraints, but from the table, the constraints for all architecture are the same. This is because the production plan in the same test scenario is the same and it prohibits some particular sequence of operations. For example, the operation "robot 4 picks the box from shelf" cannot be followed by the operation "robot 1 picks the box from the buffer" because of the constraints imposed by the production process. However, in the centralized control architecture used in the experiment, the production process is mixed with the control logic. Thus if the constraints are to be removed, not only the production process need to be modified but also the control logic may need to be modified. In the case of HCBA, the production process is separated from the control logic. Because of this, modifying process plan in HCBA based manufacturing systems should be simpler [2]. Thus, the constraints in HCBA based manufacturing systems should be easier to be removed. The DOF can be used to evaluate the available capabilities of the manufacturing systems. However, the cost of design, testing, and maintenance of the control system are not considered. These issues can be evaluated by calculating the complexity and the extension rate of the experimental manufacturing systems [5].

### 5.2 Complexity and extension rate comparison

In order to evaluate the complexity and extension rate, the components to be compared must first be identified. The next step is to create the model of the associated components or identify the relevant piece of control program. The extension rate can be calculated by counting the line of code of the program. However, the model of the control system must be created, if the complexity of the system is to be evaluated. In this project, the component to be evaluated is the centralized controller for centralized manufacturing system. For HCBA and the proposed method, a combination of the intelligent software module of all resources and products will be evaluated. The common modules (machine modules) are not considered here because all architecture uses the same modules. Thus, including or excluding these modules in the evaluation should make no difference.

The next step is to create models for all of the architectures. The Petri net models for all architecture have already been created in the implementation phase. The complexity and the extension rate will be calculated based on these Petri net models. An example of a Petri net model for the test scenario1 centralized architecture is shown in Figure 14.

The strategic complexity (SC), cyclomatic complexity (CC), and extension rate (ER) can be calculated using the equations below.

$$SC = N_{OP} + N_{OT} + N_{OA} \qquad (1)$$

Where $N_{OP}$ is the number of places, $N_{OT}$ is the number of transitions, and $N_{OA}$ is the number of arcs. The SC reflects the size of the control system.

$$CC = E - N + P \qquad (2)$$

Where E is the number of edges (arc), N is the number of nodes (places + transition), and P is the number of connected components. The CC reflects the complexity of the structure of the control system.

$$ER = \frac{SC_{i+1}}{SC_i} \qquad (3)$$

Where i is the number of scenario.

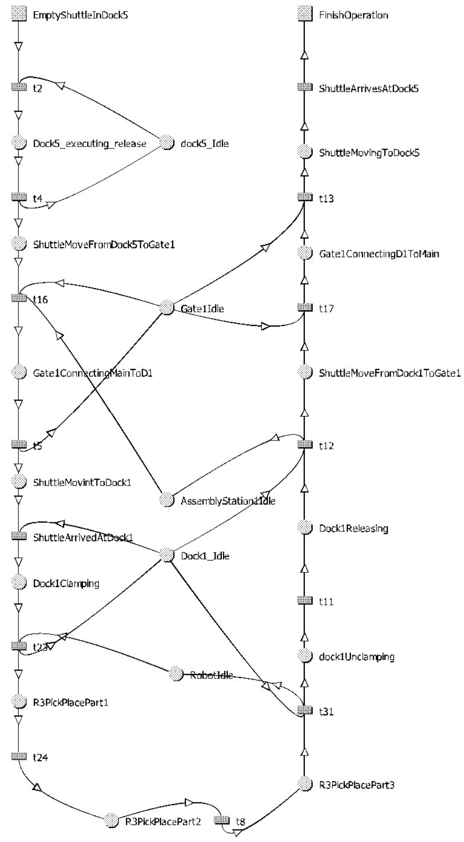


Figure 14: Petri net model for the test scenario1 centralized architecture

The complexity and extension rate of the test scenarios are shown in table 6.

| Test | Architecture | With infrastructure | | | Without infrastructure | | |
|---|---|---|---|---|---|---|---|
| | | SC | CC | ER | SC | CC | ER |
| 1 | Centralized | 76 | 9 | - | 76 | 9 | - |
| | HCBA | 177 | 17 | - | 40 | 1 | - |
| 2 | Centralized | 89 | 12 | 1.171 | 89 | 12 | 1.171 |
| | HCBA | 205 | 20 | 1.158 | 48 | 1 | 1.2 |
| | Proposed | 168 | 15 | 2.211 | 92 | 10 | 1.21 |

Table 6: Complexity and extension rate comparison

The complexity and extension rate are calculated two times. The "with infrastructure" columns use the whole Petri net model to calculate SC, CC, and ER. The "without infrastructure" columns do not include HCBA infrastructure in the calculation. Since, the infrastructure will be the same for all type of machine (or product), it can be created and provided in advance. If this is the case, the system integrator will not have to create the infrastructure. The existing HCBA template of the infrastructure can be used. However, for the centralized PLC based case, the control program is likely to be created from scratch. Since the control program for the centralized PLC architecture is created specifically for a particular application, the program tends to be smaller and less complex than those of the proposed method and the HCBA when the infrastructure is included. On the other hand, when the infrastructure is excluded from the calculation, it can be seen that the control program need to be created manually for the HCBA case and the proposed method is smaller and less complex than the centralized PLC case.

Table 6 shows that when the infrastructure is included, CC for both HCBA and the proposed method is higher than the centralized architecture. This is because the complexity of the infrastructure. When the infrastructure is excluded from the calculation, the complexity of both HCBA and the proposed methods are reduced. This is because the only part needs to be manually created is the process plan, which is very simple in these experiments.

When the infrastructure is considered, the extension rate for the proposed method is much higher than those of HCBA and the centralized architecture. This is because the control program for the proposed method comprises of both the old control program of the centralized architecture and the HCBA infrastructure, which wrapped around the old control program. However, if the infrastructure can be provided in advance, the extension rate for all architecture is almost the same. The benefits of the HCBA architecture should become more obvious when there is more than one type of product to be produced [3]. It is expected that some experiments for the other cases will be done in the future.

## 6 Conclusion

The method for integrating a new machine based on distributed paradigm is presented. Although, the benefits of the proposed method are not obvious when comparing with the centralized PLC architecture in the current scenarios, it is expected that the benefits of the proposed method will be significant when a number of new machines are to be added which is the subject of future evaluations. The benefits should also be more noticeable when there are redundancies in the system and there is more than one product type to be produced. This is an on going work and it is expected that the DSM will be included as one of the evaluation tool in the future. It is also expected that the ISA S88 architecture will be included in the list of the architecture to be compared.

## References


[1] ANSI/ISA-S88.01, *Batch Control Part1:Models and Terminology* Instrument Society of America, 2000.

[2] D. Brandl, *Design pattern for flexible manufacturing* ISA - Instrumentation, Systems and Automation Society, 2006.

[3] J.-L. Chirn, "Developing a Reconfigurable Manuacturing Control System -- A Holonic Component Based Approach." University of Cambridge, 2001.

[4] J.-L. Chirn and C. McFarlane, "Evaluating holonic control systems," 2007.

[5] J.-L. Chirn and D. C. McFarlane, "Evaluating Holonic Control Systems: A Case study," 2005.

[6] R. Evertsz, M. Fletcher, R. Jones, J. Jarvis, J. Brusey, and S. Dance, "Implementing industrial multi-agent systems using JACK," in *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science),* 3067 ed Melbourne, Australia: Springer Verlag, Heidelberg, D-69121, Germany, 2004, pp. 18-48.

[7] A. Farid, "An approach to the application of the Design Structure Matrix of Assessing Reconfigurability of Distributed Manufacturing Systems," 2006.

[8] A. Farid and D. C. McFarlane, "A developement of Degrees of Freedom for Manufacturing Systems," in *IMS'2006: 5th INTERNATIONAL SYMPOSIUM ON INTELLIGENT MANUFACTURING SYSTEMS Agents and Virtual Worlds* 2006.

[9] E. Garcia-Herreros, J. Christensen, J. M. Prado, and S. Tamura, "IMS - Holonic Manufacturing Systems: System Components of Autonomous Modules and their Distributed Control," HMS Consortium,HMS -- Strategies, Vol.1, 1994.

[10] J. Jarvis, D. Jarvis, and D. McFarlane, "Achieving holonic control--an incremental approach," *Computers in Industry*, vol. 51, no. 2, pp. 211-223, June2003.

[11] Y. Koren, U. Heisel, F. Jovane, T. Moriwaki, G. Pritschow, G. Ulsoy, and H. Van Brussel, "Reconfigurable manufacturing systems," *CIRP Annals - Manufacturing Technology*, vol. 48, no. 2, pp. 527-540, 1999.

[12] V. Marik and D. McFarlane, "Industrial adoption of agent-based technologies," *IEEE Intelligent Systems*, vol. 20, no. 1, pp. 27-35, 2005.

[13] T. J. McCabe and C. W. Butler, "Design complexity measurement and testing,", 32 ed 1989, pp. 1415-1425.

[14] D. McFarlane and S. Bussmann, "Developments in holonic production planning and control.," *Production Planning and Control*, vol. **11**, no. **6**, pp. 522-536, 2000.

[15] Monostori L, Va'ncza J, and Kumara S, "Agent based systems for manufacturing,", 55 ed 2006, pp. 697-720.

[16] W. Oney, *Programming the Microsoft Windows Driver Model*. Washington: Microsoft Press, 1999.

[17] OPC Task force, "OPC overview,"1998.

[18] H. Parunak, "Industrial and Practical Applications of DAI," in *Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence*. G. Weiss, Ed. Cambridge, MA: MIT Press, 1999, pp. 377-424.

[19] U. Rembold, B. O. Nnaji, and A. Storr, *Computer Integrated Manufacturing and Engineering*. Wokingham, England: Addison-Wesley, 1993.

[20] A. S. Tanenbaum, *Computer networks*, 4 ed. New Jersey: Prentice Hall, 2003.