Quick Simulation Model Development Method
For Assembly Systems

By

Phoenix M. F. CHEUNG, B.Eng

A Thesis Presented to
The Hong Kong University of Science and Technology
in Partial Fulfilment
of the Requirements for
the Master of Philosophy
in Industrial Engineering and Engineering Management.

Hong Kong, February, 1997

Copyright © by Phoenix M.F. Cheung 1997
Authorisation

I hereby declare that I am the sole author of the thesis.

I authorise the University of Science and Technology to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorise the University of Science and Technology to reproduce the thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.
Quick Simulation Model Development Method
For Assembly Systems

By

Phoenix M. F. CHEUNG, B.Eng

APPROVED:

Prof. Mitchell M. TSENG, SUPERVISOR, HEAD OF DEPARTMENT

Dr. Ka-ching CHAN, CHAIRMAN

Dr. Yat-Wah WAN, MEMBER

Dr. Ajay JONEJA, MEMBER

Department of Industrial Engineering and Engineering Management.
February 1997
Acknowledgments

I have to say thanks to my supervisor Prof. Mitchell M. Tseng, Dr. Yat-Wah Wan and Dr. Ajay Joneja since they guided me to finish the thesis with their great patient during the last two years. My ideas have been greatly shaped by them. Also, I have to thank Dr. Y.K. Law since this thesis has benefited from lots of his comments.

In addition, special thanks go to Mr. Andrew Lupton who is Business Development Manager of WOMEX, and Mr. Feng An Wei who is English instructor of language center of HKUST, since they spent lots of time to correct the spelling and grammatical mistakes in this thesis.

Finally, I want to express my appreciation to Andy and Prudence, who are the best friends of mine, for their support during all those days spent working on this thesis.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title Page</td>
<td>i</td>
</tr>
<tr>
<td>Authorisation Page</td>
<td>ii</td>
</tr>
<tr>
<td>Signature Page</td>
<td>iii</td>
</tr>
<tr>
<td>Acknowledgement</td>
<td>iv</td>
</tr>
<tr>
<td>Table of Contents</td>
<td>v</td>
</tr>
<tr>
<td>Abstract</td>
<td>vi</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Assembly System</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Possible techniques used in analysis for assembly system design</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Applying Simulation For Assembly System Design</td>
<td>4</td>
</tr>
<tr>
<td>1.4 Limitations of Simulation.</td>
<td>6</td>
</tr>
<tr>
<td>1.5 The Objective of the Master Thesis</td>
<td>7</td>
</tr>
<tr>
<td>2. LITERATURE REVIEW</td>
<td>8</td>
</tr>
<tr>
<td>2.1 Simulation Generator</td>
<td>8</td>
</tr>
<tr>
<td>2.2 Simulation generator for assembly line systems</td>
<td>12</td>
</tr>
<tr>
<td>2.2.1 Requirements to build specific simulation generator</td>
<td>12</td>
</tr>
<tr>
<td>2.2.2 Limitation of the existing generator for assembly line systems</td>
<td>12</td>
</tr>
</tbody>
</table>
3. QUICK SIMULATION MODEL DEVELOPING METHOD FOR ASSEMBLY SYSTEMS

3.1 Introduction

3.2 Classification of all types of assembly system

3.3 Assembly systems model constructing process

3.4 Simulation code generation

3.5 Simulation

4. SELECTION OF ASSEMBLY LINE

4.1 Introduction

4.2 Five types of assembly lines

4.2.1 Continuous Assembly Line

4.2.2 Intermittent Assembly Line

4.2.3 Manual Assembly Line

4.2.4 Flexible Assembly Flow Line

4.2.5 Flexible Assembly Cell System

(a) Multi-Stage cell

(b) Multi-Station cell

(c) Single Station cell
4.3 Capabilities and limitations of Assembly Systems 31

4.3.1 Flexibility 31

(a) Operational flexibility 31

(b) Routing flexibility 32

(c) Changeover flexibility 33

4.3.2 Throughput Rate 33

4.3.3 Operation Uniformity 34

4.3.4 Investment Cost 34

4.4 Questionnaire for selecting assembly systems 36

5. ASSEMBLY LINE MODELS CONSTRUCTING PROCESS 41

5.1 Introduction 41

5.2 Building an Assembly Line Model 43

5.3 Activities performed by workpieces 49

5.4 Resources involved in the system 52

5.5 Activities performed by resources 58

5.6 Example 62
6. AN EVALUATION EXERCISE

6.1 Introduction 71
6.2 The Experiment 73
6.3 The Participants 74
6.4 The Results 76
   6.4.1 Total time taken to finish the exercise 76
   6.4.2 Comments about Exercise A 77
   6.4.3 Comments about Exercise B 78
6.5 Discussions 79
   6.5.1 About Exercise A 79
   6.5.2 About Exercise B 80
6.6 Conclusions 80

7. CONCLUSIONS AND POTENTIAL WORKS 81

7.1 Conclusions 81
7.2 Potential works 82
   7.2.1 Real-Time Control 84
   7.2.2 Design For a Flexible Assembly System. 84
   7.2.3 Design of Simulation Experiments 85
APPENDICES

A. Quick Simulation Modeling System, Exercise A. 86
B. Quick Simulation Modeling System, Exercise B 122
C. Raw Data collected from the participants 148
   1. Questionnaire collected 148
   2. SIMAN's Model and Experiment file 166
   3. Statistical results collected from the Exercises 174
D. Flowchart for the Quick Simulation Modeling System 177
E. ACD of the 5 assembly systems 180

REFERENCES 185
Quick Simulation Model Development Method
For Assembly Systems

By

Phoenix M. F. CHEUNG, B.Eng

A Thesis Presented to
The Hong Kong University of Science and Technology
in Partial Fulfilment
of the Requirements for
the Master of Philosophy
in Industrial Engineering and Engineering Management.

Hong Kong, Feb., 1997

ABSTRACT
ABSTRACT

Simulation is an effective method for solving assembly system problems because of its wide applicability and because it provides a laboratory to study systems without the costs of building or modifying the real systems. However, it also has some disadvantages which limit its utilisation. Firstly, developing the simulation model may be time consuming and requires a simulation specialist, it does take training and practice before a beginner can perform moderately complicated simulation with a commercially available simulation language. Even for an experienced analyst, modeling and programming of a complicated system can be a laborious process and it usually takes longer than most companies can tolerate. Some simple facility can take a minimum of two weeks of modeling, programming and validation.

Quick modeling methods for modeling assembly systems have drawn interests from industry and academics. Some earlier attempts have proposed various approaches with limited scopes. For example, R. Bruce Taylor and Hamdly A. Taha developed a quick modeling method with one specific type of assembly line. In this thesis, a Quick Simulation Modeling Method is proposed with five types of most commonly used assembly systems. The proposed method starts with selecting the appropriate assembly systems according to product and production requirements. They then mapped into a set of generic models for simulation. These generic models are represented in Activity Cycle Diagram (ACD). The users can then based on generic models to input resources
in each activity and configuration of the activities. Hence, simulation models can be
generated with the input parameters.

In addition, an experiment was conducted to show, to what extent, the Quick Modeling
Approach is better than the traditional simulation modeling software. This experiment
requires some participants to build simulation models with Quick Model Building
Program (which was written by the author) and some participants to build models with
Promodel. Assuming all the participants are manufacturing engineers who know their
assembly line well, we want to see how quickly the participants can finish the model
building and simulation process with and without the Quick Model Building program.
At the ends, participants using Quick Model building program spent a shorter time to
finish the exercise than those using Promodel.
CHAPTER 1

INTRODUCTION

1.1 Assembly System

Generally, an assembly system consists of several highly independent resources that exhibit high degree of stochastic behaviour, such as machines, robots, AGV's, tool, etc. An assembly system primarily consists of these resources and a number of parts that flow through, or are processed by, these resources. The performance of each of these resources and the flow of individual parts through the system usually depend on a number of variables, such as scheduling, part routing and process planning, system configuration, system capacity, etc. These variables are themselves strongly correlated. So, the design process of an assembly system is quite complicated and the complexity of this decision making process is further increased by the system flexibility which allows for several competing designs and control solutions.

Even if all the necessary information for operating the system were available, it is still subject to another problem, that is uncertainty. We can not assume that all the available information would remain unchanged during the line-design process. Two major problem areas are production demand and engineering change. The design concept of an assembly line is heavily influenced by the desired production volume. Unfortunately, production demand is usually based on the market situation which is difficult to understand and predict and is well beyond the control of the manufacturer. On the other hand, engineering change is quite common when dealing with a new product. Any changes in product design, manufacturing process, tooling, fixtures, or even work methods may result in a need for line adjustment.
Although a line can be modified by adding, deleting or rearranging workstations and altering material-handling methods, it can be a very costly and time-consuming approach. Therefore line designer must face up to two major challenges in order to have an optimal design of an assembly system:

1. Process a large amount of information and knowledge about the system's components and control variables, as well as about the interdependency structure between these individual elements.

2. Based on this information, generate a set of feasible designs and operations control decisions that would optimise the system's performance in view of the management's objectives.

Unfortunately, the decision making process stated above is a time-consuming process and decision makers have to put a great deal of effort for comprehensive consideration so as to generate optimal results. Therefore, a system examination tool which is capable of efficiently and effectively handling many of the system complexities, functionalities and interaction characteristics of an assembly system is needed.

1.2 Possible techniques used in analysis for assembly system design

The techniques used for system examination may broadly be divided into three categories: Mathematical Optimisation approach, Hardware Prototyping and Simulation.
Because of the size of some problems, mathematical optimisation approach is extremely difficult to implement without a great deal of either simplification or segmentation. Simplification could result in unrealistic models that yield little insight into the actual system, while segmentation could result in overall counter productive decisions. So, the primary reason for using simulation is that many models cannot be adequately analysed by standard mathematical techniques. This is usually the case when the interactions between variables are non-linear or when random effects are inherent in the system. Very few real-world systems are free from the influence of random or unpredictable variables in the environment or in its own components. Systems with waiting lines, or queues, are examples of this situation. Simulation can capture this stochastic behaviour. Although there are mathematical methods to solve queuing-type problems, the solutions are available only for limited cases and do not provide complete information about system behaviour.

The alternatives of Hardware Prototyping and implementation are not only costly but also prohibitive from lead time perspective. Since quick and extensive system parameter changes are not possible using physical models, the results would be costly in terms of time and money.

Simulation is a very powerful method for solving system problems because of its wide applicability and because it provides a laboratory to study systems without the costs of building or modifying the real systems. It is a process of designing a real system model and conducting experiments with this model for the purpose of understanding the behaviour of the system and/or evaluating various strategies for the operation of the system.
In other words, it allows the construction of complex and realistic representations of actual systems. Through statistics collection or real-time observation, great insight can be gained into the operation of the system. Therefore, simulation is a tool to observe performance of different system design alternatives and to discover all potential problems of an assembly system before real implementation, thus major modification to a large capital investment after installation can be avoided.

1.3 Applying Simulation For Assembly System Design

A simulation model can be built for many reasons: to describe a current system, explore a hypothetical system, or design an improved system. Typical parameters studies in simulation would be the following:

1. Hardware Design Parameters
   - number and type of machines
   - number, type, and physical arrangement of material-handling equipment
   - location and size of inventory buffers and storage
   - effect of a new piece of equipment on an existing line

2. Operational Parameters
   - manpower requirements
   - change in product mix
   - production scheduling policies
   - reliability analysis (the effect of planned versus unplanned maintenance)
After we determine the issue which is of interest, the next logical step is to determine what kind of data must be known in order to give us directions or idea to resolve the issue. Following are some examples of measures of performance typically used in simulation:

- Throughput: The number of jobs of each type produced per time period.
- Cycle time: the amount of time it takes to get an workpiece through the system.
- Queue time: the amount of time that jobs are delayed.
- WIP: the size of work-in-process inventories.
- Downtime: the percentage of time that a machine is down or blocked.
- Utilisation: the percentage of time that people and machines are busy.

Then, extensive model parameters can be changed for different kind of experiments. By obtaining particular values from the specified performance measurements, we know how to improve the system being studied with respect to the issue previously determined. Potential problems due to the imperfection created in the product design or process design stage can be identified before real production, thereby, saving unnecessary time and cost expense. Furthermore, any interested system performance in the whole life of the system can be examined as simulation can be seen as a closed loop process. Different levels of the real system can also be modeled and examined depending on different examination objectives. Figure 1.1 illustrates a cycle of conducting simulation.
1.4 Limitations of Simulation

The previous sections has been demonstrated that simulation is very useful in production start-up and optimisation, however, it also has some disadvantages which limit its utilisation.
Firstly, developing the simulation model may be time consuming and requires a simulation specialist. The more experience the person doing the modeling has, the faster the models can be built. Even though it does not require a very long model building process like in the case of building a physical model, it does take training and practice before a beginner can perform moderately complicated simulation with a commercially available simulation language. Simulation expertise can be obtained only after months of intensive training and practice in simulation languages, such as SLAM or SIMAN.

Even for an experienced analyst, modeling and programming of a complicated system can be a laborious process. Modeling, programming and debugging in simulation analysis usually take longer than most companies can tolerate. A simple facility would probably take a minimum of two weeks of modeling, programming and validation. The time consuming part of developing the model is to establish the operation logic.

1.5 The Objective of the Thesis

The objective of this thesis is to propose a Quick Simulation Model Development method for Assembly systems. By using this approach, engineers or managers who are not simulation experts can model an assembly system and conduct simulation studies quickly and effectively.
CHAPTER 2

LITERATURE REVIEW

2.1 Quick model development method

Quick model development methods are designed to speed up the modeling and programming stages of simulation. They can generate error-free simulation code automatically, thus enabling people with almost no knowledge of simulation language to conduct studies. Moreover, time spent in modeling, programming and debugging can be reduced by using a simulation generator, especially when the system to be simulated is large.

Development of this method has been an interesting area for researchers for several years, mainly because the spreading application of computer simulation has created a relative shortage of simulation analysts. Moreover, simulation expertise can be obtained only after months of intensive training and practice in a simulation language, such as GPSS, SLAM, SIMAN, etc. Modeling, programming and debugging in simulation usually take longer than most companies can tolerate. Some of the significant work about Quick Model Development methods is the following.
Hannock and Davis [1] constructed a simulation generator for design and control of flexible manufacturing systems. Written in BASIC, the simulation generator converts input data into SIMAN simulation codes and also runs the simulation program. Brazier and Shannon [2] have reported a simulation generator developed at Texas A&M University (SIMTOOL). It generates simulation codes in SIMAN for the analysis of Automated Guided Vehicle (AGV) systems.

Montazeri and Wassehove [3] have reported a modular generator with some 20 scheduling rules for flexible manufacturing systems. Mathewson [4] reported a modular simulation generator which uses an interactive dialogue with the users to receive input data and generates simulation code in SIMSCRIPT II.5 simulation language.

A discrete-event simulation generator for operational system (SGOS) was presented in [5]. The simulation generator is based on a network called "Operations Network". This network is used to model the structure of the system to be simulated. A set of expressions termed "Operation Equations" are introduced to describe the structure of the operations network. These equations and other system specification are used by the simulation generator to create the model and experimental frame in SIMAN simulation language. It generates simulation codes according to operation equations, rather than generating a group of predefined constructs.
The advanced Assembly Line System Simulator ALSS II is a special purpose simulator used to model and analyse assembly line systems [6]. Productive system developed ALSS II to quickly model and accurately simulate a broad class of assembly line systems. It is based on System Modeling Corporation's SIMAN IV/Cinema simulation language. Seppanen claimed that ALSS II's user can skip the tedious simulation language detail and concentrate on the critical logic and timing which makes their assembly system function. A five symbol diagram which includes Assembly Line, Operation, Transfer, Return and Infinite Capacity Source/Sink, is used to model the assembly line system graphically. The graphical model is then entered into the computer using an interactive model development program. Model and Experiment files written in SIMAN will automatically be generated.

In [7], a quick simulation model generator was proposed by R. Bruce Taylor and Hamdly A. Taha. They mentioned the potential of using a higher level language, such as BASIC, to automatically generate syntactically correct simulation models from functional database. The resulting models may then be executed directly by the general purpose simulation of choice. This procedure is particularly useful in modeling assembly line systems or any network-based situation due to:

1. The model logic in this type of system is repeatable, since only input-output relationships at each node are involved.
2. A set of consistent parameters can be ascribed to each repeated logic of the model.

3. Since the input-output relationship at each node is specified by the precedence relationships which govern the operation of the system, this makes the model highly input-data dependent.

4. The basic difference in modeling at each model components centre around making changes to accommodate the precedence relationship associated with them.

This approach can directly convert the functional data of the situation under consideration into a model that can be executed by a general purpose simulation language of choice. Thereby, user will be dealing with databases with familiar formats rather than with special formats that satisfy the syntax of the general purpose simulation language itself. Each time when input data are changed for testing different situations in the same model, there is no need to debug the model since the simulation generator can provide an error free simulation model by the predefined built-in constructs.
2.2 Quick model development methods for assembly line systems

2.2.1 Requirements

It is important to note that the simulation generators mentioned in the last section are all application-oriented. Predefined constructs built inside the generator are used to define the operation logic of the system being modeled. This is done in order to free the users from building the model by themselves and users can obtain the model by simply preparing input data. Therefore, the programmer of the generator should first identify the type of system wanted to be modeled as well as its operation logic, before starting to develop the simulation generator. The generator can be written only after knowing the details of the operation logic of the system to be modeled.

2.2.2 Limitation of the existing methods for assembly line systems

[6,7] should be the two we are looking for since they are generators specially written to quickly generate simulation models for assembly line systems. However, they have some limitations.

It is a fact that there are still some varieties in assembly systems in terms of operation logic, and they should be considered carefully and included into the generator as some predefined constructs.
R. Bruce Taylor and Hamdly A. Taha did not mention this point and say nothing about what are the differences of operation logic between various kinds of assembly line system and how we can put them into the simulation generator. The line they considered was only a general assembly line. Other types of assembly system may not be suitable, such as flexible assembly lines, since their operational logic is definitely more complicated. The more careful and thorough consideration of the system's operation logic, the better performance of the generator will be. Even the generator in [7] which can generate error free simulation codes automatically and very fast, the performance and usefulness of the generator cannot be maximised since not all types of assembly system's operation logic have been considered. The objective of this thesis is to propose an overall framework of a simulation generator with all different assembly system operation networks included.

The time factor nowadays becomes more and more important. Competition in the international market makes it necessary to be quick in presenting new products on the market since if the length of time from product conception to market introduction is shorter, higher market share can be gained. Flexible assembly is becoming more frequently used since it is the solution to react to this changing environment.
R. Bruce Taylor and Hamdly A. Taha in [7] developed a generator using a higher level language to automatically generate syntactically correct simulation models from functional databases. Functional data of the situation under consideration can be directly converted into a model that can be executed by some simulation language. But the model generated for this generator only limited to general assembly line.

In the next chapter, I will introduce a Quick Simulation Model Development method for all types of assembly systems which is certainly an increment of both of the previous works. So, non-simulation experts can build models of any kind of assembly systems quickly and efficiently.
CHAPTER 3

Quick Simulation Model Developing Method for Assembly Systems

3.1 Introduction

The objective of this thesis is to propose a Quick Simulation Model Developing Method (QSMD) for all types of assembly lines. Therefore, QSMD's users, even if they are not simulation experts, can model various assembly lines very quickly simply by preparing the input data only. The time and effort required for simulation modeling will reduce. High quality of the model operation logic of various types of assembly system can be ensured and statistical results can be obtained rapidly for making management decisions. The following sections will show the steps employed to establish QSMD.

3.2 Classification of all types of assembly system

There are a number of assembly systems are existing in the market. Most of the time, production requirements and product specifications are the criteria which drive engineers or managers to select an assembly system for production. Many factors have to be reviewed and those are based on thorough understanding on the user's own application as well as the various characteristics and performance of different types of assembly lines. Factors affecting the selection depend on product variety, product complexity, product quality or throughput, etc..
So, the first thing to establish a QSMD is to classify all types of assembly system by some product and production specification. Totally, there are 5 types of assembly systems and the selecting strategy among the 5 type of assembly lines will be discussed in Chapter 4.

3.3 Assembly system models construction process

Not every area and scenario in an assembly system is of interest to the modeler. So, based on the purpose of doing simulation, the simulation model would only contain the relevant system components and its information. Moreover, each type of assembly system has its own unique specific operation logic. Therefore, we have to provide a quick and accurate method for capturing the necessary details from the selected assembly system and putting them into the simulation model. Once users have selected the type of assembly line, QSMD knows which framework should be selected and the user will be prompted to key in the corresponding input data accordingly.

The quick and accurate model development method will be discussed in Chapter 5. This idea mainly comes from the concept of one well-known modeling method: Activity Cycle Diagram (ACD). Activity cycle diagram is one of the most powerful methods used to describe the system being modeled to assist in the model-building process. Every entity is considered to perform some cycle of activities in its life, and to pass through queues between one activity and another. In other words, the life cycle of workpiece in different type of assembly line will vary. The set of parameters used to be ascribed each operation logic of the assembly lines will also be discussed.
3.4 Simulation code generation

Once the simulation model has been developed, we have to translate them into a simulation code by which the simulation can be run in the respective simulation software. In Chapter 6, a simulation translation program will demonstrate how the collected user input data can be translated to a readable simulation code. An higher level language, PASCAL, is chosen to represent different types of operation logic of the assembly lines to translate the input data into SIMAN code which is a readable simulation model of the assembly lines in ARENA.

It should be known that the simulation generator can be written by any kind of higher level language. The fact that PASCAL is used here to represent QSMD has no special meaning. The output of the QSMD would be a text file contains SIMAN simulation program which comprises both a MODEL” file and a corresponding EXPERIMENT” file. The MODEL” file is a functional description of the system's components and their interactions. The EXPERIMENT” file, on the other hand, defines the experiment conditions such as the length of time of each run and initial conditions under which the model is exercised to generate specific output data.

3.5 Simulation

After we have created SIMAN Model and Experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued.
Basically, the exact commands used to accomplish this task vary with the host computer's operating system. Simulation model generated in Chapter 6 will be run in an MS-DOS operating system and the corresponding commands for executing simulation in MS-DOS will be shown also. Figure 3.1 shows the whole Quick Simulation Model Developing Method graphically.
Classification of assembly system

Map into assembly Models

Build model

simulation code generation

simulation

Analysis

Report

Figure 3.1: Quick Simulation Model Developing Method
CHAPTER 4

SELECTION OF ASSEMBLY LINE

4.1 Introduction

Normally, the selection of an assembly system would be based on the product and production requirements such as product variety, product complexity, product stability, production volume, throughput rate and product quality. After the product and production requirements are made known, decision maker starts to look for an assembly line system which meets and can accomplish all the product and production requirements. Before we can tell which type of assembly line system is the most suitable for our product and production requirements, we have to know the capabilities and limitations of each type of assembly line system. It must be ensured that the selected assembly line system can perfectly meet the requirements. Therefore, the selection of an assembly line system requires us to understand not only our product and production requirements but also the system capabilities and limitations of different assembly line systems.

Based on literature review [8-16], 5 types of assembly line can be classified into which are the most commonly used in the market. They are applicable to different product and production purposes. The 5 types of assembly line are:
1. Continuous Motion Assembly Line

2. Intermittent Assembly Line
   - rotary
   - in-line

3. Manual Assembly Line

4. Flexible Assembly Flow Line

5. Flexible Assembly Cell System, which is a system with any combination of the following assembly cell:
   - multi-station
   - multi-stage
   - single station

---

**Figure 4.1 : Classification of 5 types of Assembly System**
4.2 Five types of assembly lines

Now let us look more closely at each of the five assembly lines.

4.2.1 Continuous Assembly Line

In a continuous motion system, the product is processed while moving. Typically, product nests move through the system in a turret or on chains at constant speed. All product components must remain at that speed, precisely positioned relative to each other during assembly operations.

<table>
<thead>
<tr>
<th></th>
<th>advantages/disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>Extremely high speed- much faster than any indexing or pallet-based system</td>
</tr>
<tr>
<td>Operation</td>
<td>Typically all-mechanical operation</td>
</tr>
<tr>
<td></td>
<td>Excessively complex for multi-component assembly</td>
</tr>
<tr>
<td>Investment cost</td>
<td>Very expensive as first machine, less so for repeats</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Difficult maintenance in multi-component assembly</td>
</tr>
<tr>
<td>Changeover</td>
<td>Very difficult, relatively inflexible in accommodating product change</td>
</tr>
</tbody>
</table>

Table 4.1: Advantages and disadvantages of Continuous Assembly Line

So, Continuous Motion Assembly Line systems are a preferred choice when:

- Production rate is extremely high

- Product design is very stable
- Product complexity is simple, only containing a few components
- No product variety

4.2.2 Intermittent Assembly Line

In this method the workpieces are transferred with an intermittent of discontinuous motion. The workstations are fixed in position and the parts are moved between stations and then registered at the proper location for assembling. All workpieces are transferred at the same time. Examples of applications of the intermittent transfer of workpieces can be found in mechanised assembly.

1. Rotary Indexing Dial Machine.

Typically, the dial sits on a cam-operated indexing drive, in a simple, rugged construction. A dial usually carries from 4 to 24 production nests. Operating stations are mounted around the dial.

2. Linear Indexing System

In this system, product nests are attached to a linear transfer means which is driven by an indexer, generally cam-operated. The transfer means can move in a horizontal plane (carousel type) or vertical plane (over-and-under type).
The system can accommodate more operations than a rotary dial, since the linear transfer means can have space for more nests and operations.

<table>
<thead>
<tr>
<th>advantages/ disadvantages</th>
<th>Rotary Indexing Dial Machine</th>
<th>Linear Indexing System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>high throughput, but <em>machine speed determined by slowest operation</em></td>
<td></td>
</tr>
<tr>
<td>Operation</td>
<td>accurate, reliable but <em>limited number of operation, limited access to product, articulately on inboard side</em></td>
<td>accurate, reliable, capability for many successive operations, good access to product and operating station when idle stations are added between operations</td>
</tr>
<tr>
<td>Investment cost</td>
<td>low cost</td>
<td><em>more expensive than rotary dial</em></td>
</tr>
<tr>
<td>Maintenance</td>
<td>durability, <em>operation halted by problem at any station</em></td>
<td></td>
</tr>
<tr>
<td>Changeover</td>
<td><em>difficulty in performing field modifications</em></td>
<td></td>
</tr>
</tbody>
</table>

Table 4.2: Advantages and disadvantages of Intermittent Line

Indexing systems are a preferred choice when:

<table>
<thead>
<tr>
<th></th>
<th>Rotary Indexing Dial Machine</th>
<th>Linear Indexing System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Production rate</td>
<td>high</td>
<td>high</td>
</tr>
<tr>
<td>Product complexity</td>
<td>relatively few operation</td>
<td>- more operations than will fit around a dial.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- operation are performed on both sides of the product</td>
</tr>
<tr>
<td>Product variety</td>
<td>none</td>
<td>none</td>
</tr>
</tbody>
</table>
4.2.3 Manual Assembly Line

Manual free transfer lines consist of multiple workplaces in which the assembly work is accomplished as the product or subassembly is passed from station to station along the line. At each workstation, one or more human workers perform a portion of the total assembly work on the product by adding one or more components to the existing subassembly. When the product comes off the final station, the work has been completed.

This system of transfer allows each workpiece to move to the next station when processing at the current station has been completed. Each part moves independently of other parts. Hence, some parts are being processed on the line at the same time that others are being transported between stations. Asynchronous transfer system offer the opportunity for great advantage in certain circumstances. In-process storage of workpiece can be incorporated into the asynchronous systems with relative ease.

4.2.4 Flexible Assembly Flow Line

Flexible Assembly System can be divided into two main groups, flow lines where the product moves along the line and the components are incrementally added and flexible cells where in-cell movement is restricted or non-existent. In the following sections, flexible assembly flow line is considered first.
The flexible flow line is a more recent development of the flow line which utilises workstations equipped with a number of identical assembly machines. These machines are identical in term if speed and capability, and can perform a range of operation on a variety of products.

![Diagram of Flexible Assembly Flow Line](#)

**Figure 4.2 : Flexible Assembly Flow Line**

These workstations are connected by a non-synchronous materials handling system that permits some degree of routing flexibility between the assembly machines at each workstation, as shown in Figure 4.2. The routing flexibility is mainly accomplished by using three different types of automated asynchronous material handling system. They are Pallet-based Flexible Automation system, Reconfigurable Conveyor system and Automated Guided Vehicle systems (AGVs). Details of these three types of Material Handling system can be found in [8,23,24].
In more complex versions, the routings allow a particular workstation to be skipped or revisited. The non-synchronous transfer necessitates workpiece buffering between workstations to maintain a satisfactorily high degree of machine utilisation.

The need to balance the work content at each workstation is an important requirement of the assembly flow line. A workstation can only release work if the next input buffer has a free position available, if this is not the case, the work must be held. This phenomenon is known as "Blocking". The converse, when a downstream workcentre is idle because of lack of work from upstream, is know as "Starvation". Additional equipment and expense is required for the control logic necessary for such sequencing.

4.2.5 Flexible Assembly Cell Systems

The words Assembly cell" can have two different meanings. The first meaning is discrete small assembly system like a Machining cell". It is usually designed to assemble a family of products. Less than four assembly robots are commonly installed in a cell.

Another meaning is a standardised component of an assembly line. It consists of a basic machine with transfer mechanism, part feeding unit and some assembling process unit, e.g. robots, pick and place mechanisms, human operators.
So, the Flexible Assembly Cell System contains a number of assembly cell. There are three types of assembly cell commonly used as an assembly cell: Multi-stage cell, Multi-station cell and Single station cell.

(a) Multi-stage cell

Figure 4.3 is a multi-stage cell in which each station is with fixed tasks but with operational flexibility matched to product range. A common buffer is used with workstations arranged so that the pick-up and deposit areas are in reach of a single transfer robots.

![Diagram of a Multi-Stage cell]

**Figure 4.3 : A Multi-Stage cell**
(b) Multi-Station cell

Figure 4.4 is a multi-station cell. The assembly workstations are linked by automated material transfer system and each assembly workstation has some product and operational flexibility.

Figure 4.4: A Multi-Station cell
(c) Single Station cell

Figure 4.5 is a single station cell with tool changing capability, so complete product and operational flexibility can be accomplished. End effector exchange also increases flexibility considerably allowing more complex tasks.

Figure 4.5 : A Single Station cell
4.3 Capabilities and Limitations of Assembly Systems

Different characteristics of assembly systems have been mentioned in the previous section. The capabilities and limitations of assembly systems are commonly measured on performance as follows:

- Flexibility, including Operational, Routing and Changeover
- Throughput
- Operation uniformity
- Investment cost

4.3.1 Flexibility

According to the author knowledge, there are three meanings of flexibility from the system point of view:

- Operational
- Routing
- Changeover

(a) Operational flexibility

Operation flexibility is the capability of any assembly station to handle a number of assembling operations.
A dedicated machine can only handle a very simple assembling task, people can handle a large number of assembling operations. A robot is considered as somewhere between a person and a machine. It can handle a little more variety of work than conventional assembly machine.

(b) Routing flexibility

Routing flexibility is the capability of the transfer system to route the different type of workpieces to the right assembling location respectively in a quick and accurate manner. The key disadvantage of the conventional conveyor system is that it is hard, take a very long time and can be expensive to reconfigure the system routing, speed and control rules, etc., once they are installed. AGVs provides a more flexible routing solution to replace the conventional conveyor. Some flexible conveyor systems with “Traffic cop” system have capability for families of products, to direct pallets to appropriate stations. Recently a more advanced “Open and Reconfigurable conveyor system” has been introduced. It is a flexible, modular conveyor system, consisting of various liner and rotational 2D and 3D elements, that can be fitted into an open CIM system architecture. It provides an extendible and reconfigurable solution, meaning that different vendors’ products can be integrated into the system, for both planned, as well as unplanned changes. This is true in terms of system configuration as well as control without necessitating any major overall system redesign.
(c) Changeover flexibility

Changeover flexibility is a capability of the assembly system to change the system configuration physically to response to planned or even unplanned events due to change of product requirement. In a market characterised by a shorter and shorter product life cycles and by a quick obsolescence of the models, it is neither difficult nor new to automate operations typical of the assembly such as storing, feeding with parts, moving parts, screwing, etc.. What can be difficult is to guarantee enough flexibility to the plant designs in such a way as to allow an automated and quick reconfiguration of the equipment necessary to assemble new products.

The Reconfigurable Material Transfer system mentioned in the previous section and a “Flexible assembly cell” are designed specially for this purpose. They can be used for any planned or unplanned physical change in the system configuration to adapt to the changing environment.

4.3.2 Throughput Rate

It is expressed as number of products or subassemblies produced per unit time.
4.3.3 Operation Uniformity

It is expressed as the ability of the assembly system to produce product in a consistent way. The smaller variation among the products produced, the greater operation uniformity the assembly system possesses.

4.3.4 Investment Cost

The cost of the equipment paid at the very beginning, some equipment may have a high initial cost but with favourable equipment running cost per unit product later due to high throughput.

Comparison of the five types of assembly lines in terms of these performance measures are shown in Table 4.3. "*" means the importance of these performance measures with respect to each type of assembly system. Therefore more "*" in a cell means the importance of the corresponding performance measurements with respect to the assembly system is greater.
<table>
<thead>
<tr>
<th>Type of Assembly System</th>
<th>Flexibility</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Operational</td>
<td>Routing</td>
<td>Change-over</td>
<td>Throughput</td>
<td>Operation uniformity</td>
<td>Investment Cost</td>
</tr>
<tr>
<td>Continuous Motion Assembly Line</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*******</td>
<td>*******</td>
<td>*******</td>
</tr>
<tr>
<td>Intermittent Rotary</td>
<td>**</td>
<td>**</td>
<td>*</td>
<td>******</td>
<td>*******</td>
<td>******</td>
</tr>
<tr>
<td>Line</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manual Assembly Line</td>
<td>*******</td>
<td>***</td>
<td>***</td>
<td>***</td>
<td>**</td>
<td>**</td>
</tr>
<tr>
<td>Flexible Flow Line System</td>
<td>*******</td>
<td>***</td>
<td>**</td>
<td>******</td>
<td>*******</td>
<td>*******</td>
</tr>
<tr>
<td>Flexible Assembly Cell System</td>
<td>*******</td>
<td>******</td>
<td>******</td>
<td>******</td>
<td>*******</td>
<td>*******</td>
</tr>
</tbody>
</table>

Table 4.3 Capabilities and Limitations of different Assembly Lines and their system features.
4.4 Questionnaire for selecting assembly systems

One factor of the product or production requirements may suggest one type of assembly line while other factors may suggest another type of assembly line. Since some of them are conflicting, they cannot all concurrently be achieved. Following is a proposed method which is introduced to guide users to select an appropriate assembly line mainly based on:

- product variety
- product complexity
- throughput
- changeover efficiency due to model changing

By answering the questions below, users will be able to figure out the type of assembly line required.

**Question 1.**

Within the planning horizon, which products is the system going to assemble?

1. Single-model product
2. Mixed-model product, but they has not been grouped into family
3. Mixed-model product which has been grouped into family
If the answer is 1, ⇒ Question 2

If the answer is 2, "Manual assembly line is selected"

If the answer is 3, ⇒ Question 4

**Question 2.**

Is high throughput required?

1. Yes

2. No

If the answer is Yes, ⇒ Question 3

If the answer is No, "Manual assembly line is selected"

**Question 3.**

How many operations are needed for the product to be produced (OP)?

OP ≤ 4 ⇒ "Continuous assembly line is suggested"

4 ≤ OP ≤ 24 ⇒ "Intermittent line (rotary type) is suggested"

OP ≥ 24 ⇒ "Intermittent line (in-line type) is suggested"
**Question 4.**

Do you want to include a Flexible Assembly Cell in your system?

1. Yes
2. No

If the answer is Yes, \(\Rightarrow\) **Question 5**

If the answer is No, "Flexible Assembly Flow Line is selected"

**Question 5.**

You can include any three types of the following cells in your system.

1. Multi-stage cell

Figure 1 is a multi-stage cell in which each station has fixed tasks but with operational flexibility matched to product range. Common buffer is used with workstations arranged so that the pick-up and deposit areas are in reach of a single transfer robot.

![Diagram of a Multi-Stage Cell](image)

**Figure 1: A Multi-Stage cell**
2. Multi-Station cell

Figure 2 is a multi-station cell. The assembly workstations are linked by automated material transfer system and each assembly workstation has some product and operational flexibility.

![Diagram of a Multi-Station cell]

**Figure 2: A Multi-Station cell**
3. Single Station cell

Figure 3 is a single station cell with tool changing capability, so complete product and operational flexibility can be accomplished. End effector exchange also increases flexibility considerably allowing more complex tasks to be handled.

![Diagram of Single Station cell]

**Figure 3: A Single Station cell**

What type of cells are required in your system? How many are required?

Put an ✓ in the □ to indicate if it is needed.

- Multi-stages Cell : ______________________
- Multi-Stations Cell : ______________________
- Single Station Cell : ______________________
CHAPTER 5

ASSEMBLY LINE MODELS CONSTRUCTING PROCESS

5.1 Introduction

We have discussed the physical and functional characteristics of the 5 types of assembly lines in the last chapter. In fact, more or less they involve the same line components, such as workpieces, conveyor, operator or pallets etc., and these components will interact with each other and perform different activities during the time of simulation.

This chapter is to introduce a method to build a high-quality assembly line model in an organised way. Following the previous chapter, 5 types of assembly line could be further developed into a model for simulation studies by an organised method. Section 5.2 will briefly introduce all the steps employed in the proposed model building method. Later in each of the following sections, details of the model building process will be discussed accordingly. An example will be given in the last section to show how the proposed model-building method can be used. This example will show the capability of this method of capturing the required information step by step. The output model will be represented by Activity Cycle Diagram (ACD).
Almost all simulation languages use some form of diagram or flow-chart to describe the system being modeled, either to assist in the model-building process, or in implementing the model on a computer, or just to help understanding and communication. They may be represented in any of the following forms:

- Structured, computer-assisted graphs
- Flowcharts
- Structured English and Pseudocode
- Condition specification
- Diagramming Techniques

One of the most powerful methods is to use Activity Cycle Diagram (ACD). Every entity is considered to perform some cycle of activities in its life, and to pass through queues between one activity and another. Details about ACD can be found in [34].
5.2 Building an Assembly Line Model

Let us first define what are the meanings of Entities, Activities and Queues, since they will be frequently used later in this chapter.

- **Entities**

Entities are the components of the system, such as the machines, workpieces, handling equipment and so on. Entities are generally of two types: permanent entities and temporary entities. Permanent entities are those which are in the model for the entire duration of simulation experiment. Temporary entities are ones which enter the model at some time, pass through it, and leave at some later time. In the assembly system model, the resources are the permanent entities and the workpieces are temporary entities.

- **Activities**

Activities are the things that entities do or have done to them. In virtually every activity more than one type of entity is involved, and hence an activity could be defined as the coming together of two or more entities for a period of time. It is an essential aspect of activities that we know how long they will last, so that when they start we can specify when they will end. Even if we are uncertain of the duration of
the activity we sample a value from a distribution, as in stochastic simulation. Activities are assumed to begin or end instantaneously.

- **Queues**

Queues are passive states of an entity, while it waits for conditions to change so that another activity can begin. In the assembly system model, workpieces are waiting for machines to finish operations on other parts. When a machine became available, the next activity could begin. There were also queues of machines waiting for workpieces to arrive at them so that they could start an operation. Although the machines not physically line up behind one another, logically there is no difference between the queues of workpieces waiting for machines and the queues of machines waiting for workpieces. Below are steps employed in the proposed model-building method. Figure 5.1 shows the steps graphically.

1. **Determine activities performed by workpieces.**

After selecting the type of assembly system, the first thing to do is to determine activities involved in the system which all workpieces will go through. According to the type of assembly system selected, some workpiece activities are compulsory to be chosen. However, there are also some optional activities which can be selected.
up to the modeler preference. For details about compulsory and optional workpiece activities in each type of assembly system, please refer to Table 5.1.

2. **Determine the resources involved in each activity performed by workpieces**

Other resources may also involve in any activities performed by workpieces. Workpieces are commonly named as temporary entities, and other resources in the assembly system can be called permanent entities. So, a modeler may include both temporary and permanent entities in each workpiece activity selected. Different assembly system may involve different resources, so by using Table 5.2.1 to Table 5.2.4, resources in each activity will be figured out.

3. **Determine the activities performed by resources.**

Each resource, i.e. permanent entity, may have its own activities inside the model. Users can determine those by using Table 5.3, which is a table listing all possible activities of each resource in 5 types of assembly line.

4. **Experimental data.**

- Data for workpieces
- Data for resources
- Simulation data

The value of the workpiece and resource data depend on the type of assembly line selected, the activities for workpiece selected, and other permanent resources selected in the previous steps. They may be data about the queue disciplines or rules of entities or duration of activities.

Rules determining how entities are selected from queues, when activities are being initiated, must be defined. Frequently the first-in-first-out (FIFO) rule will apply. However, it is often necessary to test the entities in the queue to see whether some condition holds. So queue disciplines have to be stated for each activity. For more details of the most commonly used rules, please refer to [18]. Normally in simulation activity duration are randomly sampled using the Monte Carlo method. The duration of each activity should be known so that we know when the activity is finished. We have to arrange for data to be collected for any results to be obtained. There is no point building a model and then not obtaining any results from it. The statistics are usually of two kinds:

- Time functions, such as the number of machines which are busy,
- Observed values, such as the time each workpiece is in the system.
Regarding to the simulation data, all are determined by the statistics wanted to be collected and how the experiment is designed. Please refer to [18] for all types of model parameters for each type of entity. Table 5.4 are some examples of the 3 categories of data type.

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Workpiece</td>
</tr>
<tr>
<td></td>
<td>- arrival rate</td>
</tr>
<tr>
<td></td>
<td>- workpiece variety</td>
</tr>
<tr>
<td></td>
<td>- occurring probability for each workpiece</td>
</tr>
<tr>
<td></td>
<td>- batch size</td>
</tr>
<tr>
<td></td>
<td>- station visitation sequence</td>
</tr>
<tr>
<td></td>
<td>- processing duration at each station</td>
</tr>
<tr>
<td></td>
<td>- any queuing rules applied on workpiece</td>
</tr>
<tr>
<td>2</td>
<td>Resources</td>
</tr>
<tr>
<td></td>
<td>- parameters of each specific type of resource</td>
</tr>
<tr>
<td></td>
<td>e.g. Conveyor speed, capacity of free path transporter</td>
</tr>
<tr>
<td></td>
<td>- relationship between resource</td>
</tr>
<tr>
<td></td>
<td>e.g. distance between stations, same 2nd resource shared by more than 1 primary resources, etc..</td>
</tr>
<tr>
<td></td>
<td>- breakdown and maintenance data of the resources</td>
</tr>
<tr>
<td></td>
<td>e.g. breakdown frequency, repair duration</td>
</tr>
<tr>
<td>3</td>
<td>Simulation</td>
</tr>
<tr>
<td></td>
<td>- simulation duration</td>
</tr>
<tr>
<td></td>
<td>- warm-up period</td>
</tr>
<tr>
<td></td>
<td>- no. of replication</td>
</tr>
<tr>
<td></td>
<td>- statistics wanted to be collected</td>
</tr>
</tbody>
</table>

Table 5.4: Examples of the 3 categories of data
Select an Assembly System out of the Five Types (use Questionnaire in Ch.4)

Continuous Line  Intermittent Line  Manual Line  Flexible Flow Line  Flexible Cell system

Select the optional activities for workpiece according to the assembly system selected (use Table 5.1)

Determine the resources involved in each activities selected (use Table 5.2.1-Table 5.2.4)

Determine the activities of each resource (use Table 5.3)

Give Simulation Data

Workpiece  Resources  Simulation

Figure 5.1: Assembly Line Models Building Process
5.3 Activities performed by workpieces

Next, we need to determine the activities workpieces will go through. According to the type of assembly system selected, some activities are compulsory, and they are represented ACD in Appendix E. However, there are also some optional activities which can be selected up to the modeler's preference. For details about compulsory and optional workpiece activities in each type of assembly system, please refer to Table 5.2.

According to some modeling related literature, the most basic structure of an assembly line model is Arrival-Move-Process-Exit. Figure 5.1 is an ACD of this basic model structure. ARRIVAL activity means workpieces arrive at the system from the outside world. MOVE activity means workpieces are moved between stations. Assembling activities for workpieces would be carried out in PROCESS. EXIT means workpieces leave the system and go back to outside world. It should be noted that there must be a QUEUE before activities MOVE, PROCESS and EXIT. Workpieces have to wait in QUEUE when any activity is not yet ready to happen for any reasons.
<table>
<thead>
<tr>
<th>Workpiece activities in different Assembly systems</th>
<th>C</th>
<th>I</th>
<th>M</th>
<th>FL</th>
<th>FC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrival</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Moving between stations/ cells</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Processing</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Exit</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Inspection</td>
<td></td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Loading and Unloading of fixture</td>
<td></td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Move to and from Central Storage</td>
<td></td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

**Notes:**

A.) **Symbols for the type of Assembly Line:**

- **C** = Continuous Line
- **I** = Intermittent Line
- **M** = Manual Line
- **F** = Flexible Flow Line
- **FS** = Flexible Cell System

B.) **The Meaning of 1,2 and 3.**

- **1** = Compulsory Activity
- **2** = Optional Activities selected by the modeler

**Table 5.1: Activities involved for the Temporary Entity - Workpiece**
Figure 5.2: The most basic structure of Assembly System
ARRIVAL, MOVE, PROCESS, EXIT are the compulsory activities that workpieces have to take part in all types of assembly systems. When those activities have not yet happened, workpieces have to wait in QUEUE for local storage. Obviously, workpieces will perform differently in different assembly system. For example, workpieces in a continuous line will be moved between stations within a definite length of time but workpieces in a manual assembly line will be moved according to normal distribution. INSPECTION, FIXTURE LOADING & UNLOADING and MOVE TO CENTRAL STORAGE may be selected depending on the preference of the modeler and the type of assembly system selected. For example, if a manual assembly line is selected, INSPECTION, FIXTURE LOADING & UNLOADING are the optional activities which the modeler can select according to his preference. If a continuous assembly line is selected, INSPECTION is not applicable because inspection stations are seldom used in that kind of line.

5.4 Resources involved in the system

Entities in an assembly system model include workpieces and resources. Obviously, workpieces should be inside in all types of assembly systems but different resources may be chosen based on the type of assembly system selected. Please select the permanent entities, i.e. the resources, by using Table 5.2.1 to Table 5.2.4. So, if your selection of the type of Assembly Line is:
Continuous and Intermittent Line : use Table 5.2.1
Manual Assembly Line : use Table 5.2.2
Flexible Flow Line : use Table 5.2.3
Flexible Assembly Cell System : use Table 5.2.4

<table>
<thead>
<tr>
<th>Possible Permanent Resource involved :</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrival</td>
</tr>
<tr>
<td>Moving</td>
</tr>
<tr>
<td>Processing</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Exit</td>
</tr>
</tbody>
</table>

Table 5.2.1 : Continuous & Intermittent Line

Different types of assembly line may need different kinds of resources to accomplish the assembling task at the workstation. Each assembly workstation needs at least one primary resource by which the assembling task can be completed. We can say that the primary resource would be the most basic machine, operator or even robot which is already located at the assembly station and used to accomplish the assembling task. Usually they will not be taken away. Besides the primary resources, there may also be one or more secondary resources for each assembly station.
Unlike the primary resources, secondary resources may be human operators, robots and tools such as jigs and fixtures which may be used to setup the workstation environment, load and unload the workpiece to and from the primary resource and process the workpiece. Secondary resources may be dedicated to stations which means that they are used at a particular station only, or shared by multiple assembly workstations of the same type.

| Possible Permanent Resources involved: |  
| Arrival | Not applicable |
| Moving | MHS: 1, Conveyor  
2, Free Path Transporter |
| Processing | 1,  
Primary Resource: Semi-Automated Machine  
Secondary Resource: Operator / Tool / Nil  
2,  
Primary Resource: Operator  
Secondary Resource: Tool / Nil |
| Exit | Not applicable |
| Inspection | 1,  
Primary Resource: Semi-Automated Machine  
Secondary Resource: Operator / Tool / Nil  
2,  
Primary Resource: Operator  
Secondary Resource: Tool / Nil |
| Storage | Local Buffer at each station |
| Loading and Unloading of fixture | Primary Resource: Operator  
Secondary Resource: Tool / Nil  
Other: Fixtures |

Table 5.2.2 : Manual Assembly Line
<table>
<thead>
<tr>
<th>Possible Permanent Resources involved:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrival</td>
</tr>
</tbody>
</table>
| Moving                                | MHS: 1, Conveyor  
                                        2, Free Path Transporter  
                                        3, Guided Vehicle |
| Processing                            | 1, Primary Resource: Semi-Automated Machine  
                                        Secondary Resource: Operator / Tool / Nil  
                                        2, Primary Resource: Operator  
                                        Secondary Resource: Tool / Nil  
                                        3, Primary Resource: Robot  
                                        Secondary Resource: Operator / Tool / Nil |
| Exit                                  | Not applicable |
| Inspection                            | 1, Primary Resource: Semi-Automated Machine  
                                        Secondary Resource: Operator / Tool / Nil  
                                        2, Primary Resource: Operator  
                                        Secondary Resource: Tool / Nil  
                                        3, Primary Resource: Full-automated Machine  
                                        Secondary Resource: Nil |
| Loading and Unloading of fixture      | Primary Resource: Operator  
                                        Secondary Resource: Tool / Nil  
                                        Other: Fixtures |
| Storage                               | 1, Local Buffer at each station  
                                        2, Central Buffer |

Table 5.2.3: Flexible Flow Line
<table>
<thead>
<tr>
<th></th>
<th>Possible Permanent Resources involved</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrival</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Moving</td>
<td>MHS : 1, Conveyor</td>
</tr>
<tr>
<td></td>
<td>2, Free Path Transporter</td>
</tr>
<tr>
<td></td>
<td>3, Guided Vehicle</td>
</tr>
<tr>
<td>Processing:</td>
<td></td>
</tr>
<tr>
<td>Multi-stage Cell</td>
<td>Primary Resource: Semi-Automated Machine</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource : Tool / Nil</td>
</tr>
<tr>
<td></td>
<td>in-cell MHS : MHS : Robot</td>
</tr>
<tr>
<td>Multi-station Cell</td>
<td>1, Primary Resource: Semi-Automated Machine</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource : Operator / Tool / Nil</td>
</tr>
<tr>
<td></td>
<td>2, Primary Resource: Operator</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource: Tool / Nil</td>
</tr>
<tr>
<td></td>
<td>3, Primary Resource: Robot</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource: Operator / Tool / Nil</td>
</tr>
<tr>
<td>Single station Cell</td>
<td>Primary Resource: Robot</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource: Operator / Tool / Nil</td>
</tr>
<tr>
<td>Exit</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Inspection</td>
<td>1, Primary Resource: Semi-Automated Machine</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource : Operator / Tool / Nil</td>
</tr>
<tr>
<td></td>
<td>2, Primary Resource: Operator</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource: Tool / Nil</td>
</tr>
<tr>
<td></td>
<td>3, Primary Resource: Full-automated Machine</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource: Nil</td>
</tr>
<tr>
<td>Loading and Unloading of fixture</td>
<td>Primary Resource: Operator</td>
</tr>
<tr>
<td></td>
<td>Secondary Resource: Tool / Nil</td>
</tr>
<tr>
<td></td>
<td>Other: Fixtures</td>
</tr>
<tr>
<td>Storage</td>
<td>1, Local Buffer at each cell</td>
</tr>
<tr>
<td></td>
<td>2, Central buffer</td>
</tr>
</tbody>
</table>

Table 5.2.4: Flexible Cell System
Usually most of the entities in the model will be obvious from a simple description of the system to be modeled. However for other entities it may not be obvious. For example, if a workpiece is placed in a fixture and placed on a pallet, do all three objects need to be included as entities in the be model, or can the combination be considered as a single entity, perhaps called a job. In general if the different items follow separate path they should be treated as separate entities; if they always act as a team they can be considered as a single entity. This is a fundamental aspect of model building.
5.5 Determine the activities performed by resources.

In fact, resources in an assembly system i.e. permanent entities would carry out their own activities. Table 5.3 has all possible activities that different types of permanent entities would be carried out by 4 types of resource are classified, they are Primary Resource, Secondary Resource, Material Handling Resource and Fixture.

1. Primary Resources

They can be used in activities of fixture loading/unloading, assembling and inspection.

2. Secondary Resources

They can be used in activities of fixture loading/unloading, assembling and inspection.

3. Material Handling Resource

They can be used to move workpieces between stations, in and out the central storage and in and out of the system.

4. Fixture

Workpiece are located onto the fixture for processing.
By referring to the type of resources, i.e. the permanent entities, selected in the previous steps, choose the related activities for permanent entities in Table 5.3.

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Possible Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Primary Resource</td>
<td>Human operator</td>
<td>- Assembling in one station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Idle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Tea break</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Unexpected leave</td>
</tr>
<tr>
<td></td>
<td>- Hard automated m/c</td>
<td>- Assembling</td>
</tr>
<tr>
<td></td>
<td>- Semi-automated m/c</td>
<td>- Idle</td>
</tr>
<tr>
<td></td>
<td>- Robot</td>
<td>- Sudden breakdown</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Repair</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Expected Maintenance</td>
</tr>
<tr>
<td>2 Secondary Resource</td>
<td>- Operator</td>
<td>- Setup the station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- load the workpiece into the station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Assembling</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Serve more than 1 station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Idle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Tea break</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Unexpected leave</td>
</tr>
<tr>
<td></td>
<td>- Tool</td>
<td>- Setup the station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- load the workpiece into the station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Assembling</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Serve more than 1 station</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Idle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sudden breakdown</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Repair</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Expected Maintenance</td>
</tr>
<tr>
<td>3. Material Handling</td>
<td>Conveyor</td>
<td>- workpiece access conveyor</td>
</tr>
<tr>
<td>Resources</td>
<td></td>
<td>- workpiece being moved</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- workpiece exit the conveyor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- moving with no workpiece</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sudden breakdown</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Repair</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Expected Maintenance</td>
</tr>
</tbody>
</table>

Table 5.3: Possible Activities for Permanent Entities
<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Possible Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. Material Handling</td>
<td>Free Path Transporter</td>
<td>- empty transporter move to station to pick up workpiece</td>
</tr>
<tr>
<td>Resources</td>
<td></td>
<td>- workpiece being loaded onto it</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- move workpiece to any destinations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- workpiece being unload from it</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- transporter is idle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- empty transporter move to staging areas</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sudden breakdown</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Repair</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Expected Maintenance</td>
</tr>
<tr>
<td></td>
<td>Guided vehicle</td>
<td>- find a path from the present location of the vehicle to the location of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>requested workpiece</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- empty transporter move to station to pick up workpiece</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- workpiece being loaded onto it</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- move workpiece to any destinations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- deposit the pallet</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- transporter is idle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- empty transporter move to staging areas</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sudden breakdown</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Repair</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Expected Maintenance</td>
</tr>
<tr>
<td></td>
<td>Robot</td>
<td>- place workpieces between machines</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- move workpiece from input buffer to 1st machine</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- move workpiece from last machine to the output buffer</td>
</tr>
</tbody>
</table>

Table 5.3: Possible Activities for Permanent Entities (cont.)
<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Possible Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Other</td>
<td>Fixture</td>
<td>- Empty fixture</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- being loaded with workpiece</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- with workpiece</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- being unloaded from workpiece</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- being sent to central storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- stored inside central storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- being sent out from central storage</td>
</tr>
</tbody>
</table>

Table 5.3: Possible Activities for Permanent Entities (cont.)
5.6 Example

This is an example to show how a model can be constructed by the steps mentioned in the previous section. An ACD is used to show the model logic to provide a clearer idea about how the model can be operated.

**Step 1** : Assuming our model is a *Flexible Assembly Flow Line*.

**Step 2** : Now we start using Table 5.1. Activities have to be included are:

- Arrival
- Moving
- Processing
- Exit
- Inspection (Optional)

Up to now, the model only contains one temporary entity, i.e. Workpiece.

Figure 5.3 is an ACD which can represent entity inside the model.
Figure 5.3 ACD of workpieces
Step 3: Since our assembly line is a Flexible Flow line, we should use Table 4.2.3. It time to tell other details about the permanent resources with respect to each activity selected.

Moving: The MHS used is 2 Free Path Transporter

Processing: There are 4 stations in the cell

Station 1: PR is operator
            : SR is Nil

Station 2: PR is Semi-Automated Machine
           : SR is Operator

Station 3: PR is Robot
           : SR is Nil

Station 4: PR is Operator
           : SR is Tool

Inspection: PR is Semi-Automated machine
           : SR is Operator
Up to now, the model contains not only temporary entity, i.e. Workpiece, but also permanent entities, they are:

- Free Path Transporter,

- 4 primary resources and 2 secondary resources in assembling activity

- 1 primary resource in inspection

- 1 secondary resource in inspection activity

Step 4: We have determined the permanent entity involved inside the model, let us also decide their associated activities by using Table 5.3

1. Free Path Transporter,

- empty transporter move to station to pick up workpiece

- workpiece being loaded onto it

- move workpiece to any destination

- workpiece being unload from it

- transporter is idle

- empty transporter move to staging areas

2. Operator: PR in Station 1

- Assembling

- Idle
3. Semi-Automated M/C: PR in Station 2
   - Assembling
   - Idle

4. Operator: SR in Station 2
   - Assembling
   - Idle

5. Robot: PR in Station 3
   - Assembling
   - Idle

6. Operator: PR in Station 4
   - Assembling
   - Idle

7. Tool: SR in Station 4
   - Assembling
   - Idle
8. Semi-Automated M/C: PR in inspection

- Assembling
- Idle
- Sudden breakdown
- Repair

9. Operator: SR in inspection

- Assembling
- Idle

Figure 5.4 is an ACD which can represent those entities and the associated activities inside the model.
Figure 5.4: ACD represents both temporary and permanent entities in the model
Step 5: Now, the model is not completed yet since experimental data has not been given. According to the activities and resources selected in the previous steps, data required for input are as follows. Those can be referred to [18].

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Examples</th>
</tr>
</thead>
</table>
| Workpiece | 1.1 arrival rate  
            | 1.2 workpiece variety  
            | 1.3 occurring probability for each workpiece  
            | 1.4 batch size  
            | 1.5 station visitation sequence  
            | 1.6 what happens after inspection  
            | 1.7 what happens during inspection breakdown  
            | 1.8 processing duration at each station  
            | 1.9 any queuing rule applied on workpiece  
            | 1.10 probability for good and bad workpiece in inspection |
| All resources | 2.1 parameters of the free path transporter:  
                 | - number of units  
                 | - velocity  
                 | - acceleration  
                 | - deceleration  
                 | - initial position  
                 | - initial status ? active/inactive |
|            | 2.2 relationship between resources  
                 | - distance between the 4 stations, arrival station, the inspection station and the Exit. |
|            | 2.3 breakdown and maintenance data of the resources  
                 | - breakdown frequency of PR in inspection  
                 | - repair duration after breakdown |
| Simulation | 3.1 simulation duration  
            | 3.2 warm-up period  
            | 3.3 no. of replication  
            | 3.4 statistics wanted to be collected |
By going through the above 4 steps, a simulation model of a Flexible Assembly Flow Line can be established.

The example just presented is a modeling building process which was done manually. If the assembly system has many stations or there are a lots of resources needed to be put into the model, it would be very cumbersome. Moreover, the model is not yet for simulation since it is not a readable simulation language of any simulation software. Therefore, a user interface is definitely necessary not only for collecting the information for building a simulation model, but also to translate the information into a readable simulation language.
6.1 Introduction

With respect to the previous chapters, many guidelines have been shown for building simulation models for five types of assembly lines. This illustrative example is to demonstrate how our Quick Modeling Approach can direct users to input the required data so that an assembly line simulation model can be established quickly and accurately. A program, which is written in PASCAL, will be used to translate the information into a simulation model which is in SIMAN language. Since the output of this approach is an error-free SIMAN simulation model and can be run directly inside ARENA environment, statistical results can be collected directly after certain runs of the model without any debugging process.

Chapter 4 has already discussed by what means an appropriate assembly line can be selected in accordance with some product and production requirements. Chapter 5 told the organised modeling development method in each type of assembly line. Having selected the required assembly line, user will be guided to develop the model accordingly.
We have already explained how to construct the assembly line simulation model but the one who decide which method to adopt is user himself. Quick Modeling Approach requires users to input the relevant information step by step in order to generate the output, i.e. the simulation model. A Pascal program is employed as the Quick Modeling system to collect this data and translate it into a SIMAN simulation language. We are not going to establish a full software package about this Quick Modeling Approach for assembly line simulation model, so not all five types of assembly line models will be included inside this Pascal program. The purpose of this PASCAL program is to show, by an example, how a user who knows nothing about simulation modeling can be guided to key in the required information to establish the simulation model very quickly and have the required statistics afterward.

We still want to know how the Quick Modeling Approach is, to what extent, better than the traditional simulation modeling software, so an experiment was designed and conducted.
6.2 The Experiment

The experiment contains two exercises, one is named Exercise A and the other is Exercise B. Exercise A requires the participants to build a simulation model with Quick Model building Program while Exercise B requires participants build models with their most familiar simulation software Promodel. Assuming all the participants are manufacturing engineers who know their assembly line well, we want to see how quickly the participants can finish the model building and simulation process with and without the quick model building process and the respective difficulties encountered. Handouts for Exercise A and Exercise B can be found in Appendix A and Appendix B. For the Quick Model building Program which is written in Pascal, please find in Appendix D.

At the end, the participants were required to answer the questions and give some comments about the exercise by an questionnaire. Raw data collected from the participants can be referred in Appendix C.
6.3 The Participants

During the last few weeks, 6 final year students in IEEM department of HKUST were invited to evaluate the performance of this Quick Model Building program. Three of them have used the demonstrative Quick Model Building program to build 3 assembly line models and the rest of them used their most familiar simulation software, PROMODEL.

The participants are labelled 1, 2, 3… 6. The first 3 participants were grouped to do Exercise A with Quick Modeling Building program. Whereas the other 3 participants were grouped to do Exercise B in which they have to do simulation solely by using Promodel.

<table>
<thead>
<tr>
<th>Participants No.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exercise A</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Quick Model Building Program)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exercise B</td>
<td></td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>(Promodel)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.1: Allocation of the 6 participants
As indicated in the collected questionnaire, all of the participants knew at least one kind of the simulation software before. The most popular simulation software they have learnt is Promodel because it had been introduced in our simulation course and students have to do the course project with Promodel. Moreover, Participant No.2 even knows another simulation software called Witness” quite well since he has experience in building model with Witness last year in his summer job company. Figure 6.2 is a chart to tell the number of weeks they have taken to learn their most familiar simulation software. Light grey represents Promodel” whereas dark grey represents Witness”.

![Time used to learn the simulation software](chart)

**Figure 6.1**: Number of weeks taken to learn their familiar simulation software.
6.4 The Results

6.4.1 Total time taken to finish the exercises

Participants spent different lengths of time to finish the Exercise. Following are the results.

<table>
<thead>
<tr>
<th>Participant</th>
<th>Exercise</th>
<th>Length of time to finish the Exercise</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
<td>1.5</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
<td>1.5</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>1.5</td>
</tr>
<tr>
<td>4</td>
<td>B</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>B</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>B</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 6.2: Total time taken to finish the Exercise A and Exercise B

6.4.2 Comments about Exercise A

Having discussed with participant 1, 2 and 3, we can have the following comments about the Exercise A. Those comments mainly fall into three categories, they are all related to their knowledge on Assembly System, QSMS interface and the QSMS logic.
1. Assembly System Knowledge

The participants do not know very much about Assembly systems, Flexible Manufacturing Concept, Product Family etc. So it was quite hard for them to select the right type of assembly system by themselves.

2. System Interface

The participant often enter the wrong data into QSMS. However QSMS can not provide any way for them to go back the previous steps for amendments. Moreover, QSMS has no self-explanatory means appeared in the screens and as a results they have to ask the author questions frequently for more details or refer to the Handout A back and forth.

3. Program logic

QSMS requires users to select the type of assembly system first, then the activities of the workpieces, the types of resources involved and their activities require input. Lastly simulation data about workpieces, resources and the experiments are requested. All participants thought that it is a clear and convenient way for users to input the necessary data for simulation modeling.
6.4.3 Comments about Exercise B

Likewise, having discussed with participants 4, 5 and 6, we can have the following comments about the Exercise B. Those comments mainly fall into two categories, they are related to their knowledge on Assembly System, and the familiarity of Promodel.

1. Assembly System knowledge

Since Exercise B requires the participants build simulation models with Promodel, they do not need to know assembly systems very well, they were allowed to build model by their own way.

2. Familiarity of Promodel

All the participants have no experience in building models of assembly system, so they found it very difficult to build model with material transfer system like AGV and conveyor.
6.5 Discussion

6.5.1 About Exercise A

Exercise A requires participants know how to select assembly systems well. After selecting the type of assembly systems, QSMS knows the framework of the simulation model and QSMS will prompt users to input the necessary data step by step. So, with the guidelines provided in QSMS, users can build a model and conduct simulations very quickly and effectively.

6.5.2 About Exercise B

It is surprising that the students who took Exercise B have to spend so much time to finish the exercise even they all have learnt Promodel in the simulation course. After discussing with them, following may be the reasons.

Even though students in Exercise B were not required to know very well about how to select assembly system and they have quite a lots of experience in using Promodel, they spent very long time to finish the exercise. Since what they have built before are not manufacturing systems models, but models of Bank, Offices, etc..
They haven’t got enough concept about manufacturing system, they do not know many meanings of some specific terms, such as Accumulating Conveyor, Automated Guided Vehicles (AGV), Automated Storage and Retrieval System (ASRS), Flexible Assembly System, etc. So they have to take time to understand the building construct in Promodel.

6.6 Conclusions

Since guidance is provided in QSMS, participants definitely can spend a shorter time to finish Exercise A. Exercise A requires students know the types of assembly systems well and Exercise B requires students, to a certain extent, familiar with Promodel, especially in building model of assembly system. Participants in Exercise B have no experience in building Assembly system model before. That is the main reason why they took a long time to finish the exercise. Although participants in Exercise A have little concept of selecting assembly systems, Handout A provides enough instructions for them to select an appropriate assembly line and simulation model can be constructed quickly by using QSMS.

To conclude, QSMS is a good means for those who knows assembly system well but have little or even no knowledge about simulation.
CHAPTER 7

CONCLUSIONS AND POTENTIAL PLANS

7.1 Conclusions

So far, we have discussed the method of modeling different kinds of assembly systems in the previous chapters. In fact, it is only a tool to assist decision making in the area of design or control of the system, it has no ability to make decisions. So, if we can include all the related information and knowledge about design, and control of different kinds of assembly systems in this Quick Simulation Model Development Method, not only can the decision be made much more faster but also we can ensure the quality of that decision.

In the previous chapters, five types of assembly system have been classified in accordance with some product and production requirements. We have discussed the physical characteristics of each type of assembly system also. Quick Model Developing Approach can assist users to select the appropriate assembly system and build the simulation model quickly, but the user still need to design the simulation experiments by himself. Moreover, all the data of workpieces or resources should be provided by the user. The quality of the assembly system designs are all rely on the user knowledge about the system but not the Quick Modeling System. Quick Modeling Approach is only a tool which can shorten the time required in modeling.
debugging and verification of the simulation model, so that statistical data can be obtained quickly. Following are some proposals to the further improve this Quick Simulation Model Development Approach.

7.2 Potential works

In this thesis, one of the limitations of using Quick Model Development approach is there are no animation and graphical output results generated. Since the animation module and output module in Arena were not linked to the Quick Modeling system. So, in order to obtain the good graphical results after simulation running, we have to take advantages of those modules in Arena.

Besides, by using this Quick Modeling approach, we can only model any type of the 5 assembly systems separately. If users want to model the mix of the assembly system at the same time, no mean was provided for this purpose. Therefore, Quick Simulation Model Development approach should be improved to provide some means for modeling the mix of any 5 types of assembly systems.
Simulation languages are generally poor in decision making. If we create for instance a manufacturing schedule with simulation, there are several decision points in the process of the simulation. For examples, if there is more than one workpiece in a parallel buffer, the question is which workpiece can leave the buffer first, or if more than one machine tool is suitable for a certain operation, which machine to choose, etc.. In most of the simulation languages these kinds of decisions can be made using the well known if-then-else structure. However, it is rather difficult to write the decision rules if we have a large number of conditions to satisfy at the same time.

The so called knowledge based (KB) systems promise good solutions to the above problems. These systems store the knowledge of a specific field in a separate knowledge base, and a so-called inference engine can make decisions with high accuracy and efficiency. Expert system (ES), Production Systems (PS), Inference System (IS), Rules Based Systems (RBS), etc. belong to this category.
7.2.1 Real-Time Control

Real-time control of a manufacturing cell is an increasingly important area of application for knowledge-based systems and simulation. Within the context of manufacturing, real-time control systems provide decisions for a specific problem associated with part processing, material movement, inspection, storage and distribution, etc. A real-time controller should be capable of reacting to the problem instantaneously, evaluating several alternative policies, providing optimum or near optimum solutions, learning from the previous problems and providing faster and more accurate solutions. Integration of simulation and knowledge bases can solve a variety of control problems in manufacturing cell.

7.2.2 Design For a Flexible Assembly System.

Simulation modeling serves merely a descriptive function, leaving the decision maker with the difficult task of generating, testing and selecting design and operation alternatives. These tasks typically require a great amount of information processing and interpreting. With limitations on time and financial resources, these tasks become prohibitively expensive and decision makers are forced into ad-hoc problem solving and reactive decision making.
Therefore, it has become increasingly important to provide decision makers, especially flexible assembly system decision makers, with not only a descriptive modeling tool, but also a prescriptive aid capable of automatic generation, testing and selection of scenarios for system design and operation. Since it is not effective to implement this function in the form of a procedural program due to the complexity and variability of the associated state space. Therefore, knowledge-based systems are the possible solution.

7.2.3 Design of Simulation Experiments

One of the features of simulation likely to benefit from application of knowledge-based systems is that of the actual design of experiments. Simulation results have to be assessed for validity rather than accepted at face value. Frequently, this means performing several runs to ensure that the observed effect is not due to some random effect of the sampling process. Simulation experiments have to be designed to ensure that sufficient, or preferably, the minimum number of runs are carried out. The conditions in which they are run have to be specified. A knowledge-bases system could be valuable in determining these parameters.
APPENDIX A: EXERCISE A

1. Introduction
2. Objectives
3. Structure of QSMS
4. Background information for the case
   4.1 You are a Manufacturing Engineer......
   4.2 Product Information
   4.3 The 3 Proposed Designs
      4.3.1 Free Path Transporter - Fork Truck
      4.3.2 Guided Vehicles - AGVs
      4.3.3 Accumulating Conveyor - Power Roller Conveyor
5. Details Data for the models
   5.1 Workpiece Related Data
   5.2 Resources Related Data
   5.3 Simulation Related Data
6. SIMAN Program Files & Collect Statistical Results
7. Your Tasks
   7.1 Study this Handout
   7.2 Build 3 simulation models
   7.3 Run the models in Arena
   7.4. Questionnaires
8. Questionnaires
1. Introduction

Nowadays, simulation plays very important role in the area of system design since it provides strong and effective way for managers or engineers to measure the performance of their system design. Its allow the construction of complex and realistic representations of actual system. Through statistics-collection or real-time observation, great insight can be gained into the operation of the system. Therefore, simulation is a tool to observe performance of different system design alternatives and to discover all potential problems of an assembly system before real implementation, thus major modification to a large capital investment after installation can be avoided.

Even though simulation has been demonstrated that it is very useful in production start-up and optimisation, it also has some disadvantages which seriously limits its utilisation. Firstly, developing the simulation model is not quick and requires a simulation specialist. The more experience the person doing the modeling has, the faster the models can be built. Even though it doesn't require a very long model building process like in the case of building a physical model, it does take training and practice before a beginner can perform moderately complicated simulation with a commercially available simulation language. Simulation expertise can be obtained only after months of intensive training and practice in a simulation language, such as SLAM or SIMAN.
Even for an experienced analyst, modeling and programming of a complicated system can be a very time consuming process. Modeling, programming and debugging in simulation analysis usually take longer than most companies can tolerate. A simple facility would probably take a minimum of two weeks of programming, verification, and validation. The time consuming part of developing the model is to establish the operation logic. Surprising, the logic for operating the facility is normally one of the last things defined.

In order to maximise the possible advantages we can gain in the simulation studies, we have to overcome those limitations we have mentioned above. One of the most possible method is to provide a user interface by which user can skip all the programming and debugging process. Therefore, a user interface, named Quick Simulation Modeling System, was specially designed for modeling all type of assembly systems. It includes all type of assembly line's operation logic. QSMS's users, even have no knowledge about simulation, can model various assembly lines very quickly simply by selecting one of the model logic and preparing the input data only. Time and effort required for simulation modeling will become less. High quality of the model operation logic of various type of assembly system can be ensured and statistical results can be obtained rapidly for making management decisions.
2. Objectives

The objective of carrying out this simulation exercise are:

(A). To experience how a Quick Simulation Modeling System (QSMS) can quickly build a simulation model and effectively generate user wanted statistical results for an assembly system.

(B). To evaluate the performance of QSMS by answering the question in questionnaire.
3. Structure of QSMS

QSMS's users, even have no knowledge about simulation, can model various assembly lines very quickly simply by selecting the model provided inside QSMS and preparing necessary data only. Time and effort required for simulation modeling will definitely become less. High quality of the model operation logic of various type of assembly system can be ensured and statistical results can be obtained rapidly for making management decisions. Following are the structure of QSMS:

1. Based on Your Product and production requirements, select the type of assembly lines which are to be modeled.

At the very beginning, QSMS will prompt you to select the type of assembling line wanted to be modeled. In order to choose the right assembly line model from QSMS, many factors have to be reviewed and those are based on thorough understanding on the user's own application as well as the various characteristic and performance of different type of assembly lines. Factors affecting the selection may be depending on product variety, product complexity, product quality or throughput, etc. Step 1 is the procedure used to identify the type of assembly line required to be modeled from studies, we have to overcome those limitations we have mentioned above. One of the most possible method is to provide a user interface by which user can skip all the programming and debugging process. Therefore, a user interface, named Quick Simulation Modeling System, was specially designed for modeling all type of assembly systems. It includes all type of assembly line's operation logic. QSMS's users, even have no knowledge about simulation, can model various assembly lines very quickly simply by selecting one of the model logic and preparing the input data only. Time and effort required for simulation modeling will become less. High quality of the model operation logic of
various type of assembly system can be ensured and statistical results can be obtained rapidly for making management decisions, even have no knowledge about simulation, can model various assembly lines very quickly simply by selecting one of the model logic and preparing the input data only. Time and effort required for simulation modeling will become less. High quality of the model operation logic of various type of assembly system can be ensured and statistical results can be obtained rapidly for simulation modeling will become less. High quality of the model operation logic of various type of assembly system can be ensured and statistical results can be obtained rapidly for making management decisions.
2. Objectives

The objective of carrying out this simulation exercise are:

(A). To experience how a Quick Simulation Modeling System (QSMS) can quickly build a simulation model and effectively generate user wanted statistical results for an assembly system.

(B). To evaluate the performance of QSMS by answering the question in questionnaire.
3. Structure of QSMS

QSMS's users, even have no knowledge about simulation, can model various assembly lines very quickly simply by selecting the model provided inside QSMS and preparing necessary data only. Time and effort required for simulation modeling will definitely become less. High quality of the model operation logic of various type of assembly system can be ensured and statistical results can be obtained rapidly for making management decisions. Following are the structure of QSMS:

1. Based on Your Product and production requirements, select the type of assembly lines which are to be modeled.

At the very beginning, QSMS will prompt you to select the type of assembling line wanted to be modeled. In order to choose the right assembly line model from QSMS, many factors have to be reviewed and those are based on thorough understanding on the user's own application as well as the various characteristic and performance of different type of assembly lines. Factors affecting the selection may be depending on product variety, product complexity, product quality or throughput, etc.. Step 1 is the procedure used to identify the type of assembly line required to be modeled from user so that QSMS know which type of assembly line' operation logic can be selected accordingly. Totally 8 main types of assembly lines will be provided.
2. Determine all the Activities for workpieces in the model.

After selecting the type of assembly line, the next step in QSMS is to determine workpiece activities involved in the system. There are some compulsory activities have to be chosen in order to make the model complete. Also, there are optional activities which can be selected up to the modeler preference.

3. Determine all other resources involved in each activity of workpiece selected in step 2.

Besides workpiece, other resources like fixture, operator or machines could also be involved in a model. So, modeler may include these resources in each activity selected for workpiece in the previous step.

4. Determine all the activities of each resource selected in Step 3.

Each resource may has it own activities inside the model, user can determine those by answering the question in QSMS listed all possible activities for each resource.
5. Modeler gives experimental data for workpieces activities and resource activities selected

There are totally 3 types of experimental data modeler have to provide, they are:

1. Data for workpiece (temporary entity),
2. Data for resources (permanent entity) and
3. Simulation Data.

According to the activities and resources selected in the previous steps, QSMS will ask for the necessary data and modeler have to answer them. Two SIMAN files will be generated after this step. They are model and experiment source files for your problems.

6. Execute Both SIMAN model and Experiment files and then collect the statistical results.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation selected accordingly. Totally 8 main types of assembly lines will be provided.
2. Determine all the Activities for workpieces in the model.

After selecting the type of assembly line, the next step in QSMS is to determine workpiece activities involved in the system. There are some compulsory activities have to be chosen in order to make the model complete. Also, there are optional activities which can be selected up to the modeler preference.

3. Determine all other resources involved in each activity of workpiece selected in step 2.

Besides workpiece, other resources like fixture, operator or machines could also be involved in a model. So, modeler may include these resources in each activity selected for workpiece in the previous step.

4. Determine all the activities of each resource selected in Step 3.

Each resource may has it own activities inside the model, user can determine those by answering the question in QSMS listed all possible activities for each resource.
5. Modeler gives experimental data for workpieces activities and resource activities selected

There are totally 3 types of experimental data modeler have to provide, they are:

1. Data for workpiece (temporary entity),
2. Data for resources (permanent entity) and
3. Simulation Data.

According to the activities and resources selected in the previous steps, QSMS will ask for the necessary data and modeler have to answer them. Two SIMAN files will be generated after this step. They are model and experiment source files for your problems.

6. Execute Both SIMAN model and Experiment files and then collect the statistical results.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.

6. Execute Both SIMAN model and Experiment files and then collect the statistical results.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.

6. Execute Both SIMAN model and Experiment files and then collect the statistical results.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system.
2. Data for resources (permanent entity) and

3. Simulation Data.

According to the activities and resources selected in the previous steps, QSMS will ask for the necessary data and modeler have to answer them. Two SIMAN files will be generated after this step. They are model and experiment source files for your problems

**6. Execute Both SIMAN model and Experiment files and then collect the statistical results.**

After we have created model and experiment source files for the problem, we are ready to take steps necessary to execute our simulation. If you are building the model in the ARENA graphical environment, these steps are completely automated for you. However, we are in the ARENA command line environment, appropriate commands to compile, link and execute the model have to be issued. Basically, the exact commands used to accomplish this task vary with the host computer's operating system. The example model produced in Chapter 5 will be run in an MS-DOS operating system and the corresponding commands for executing simulation in MS-DOS will be shown also.
4. Background information for the case

4.1 You are a Manufacturing Engineer......

Supposing you are a manufacturing engineer in a automobile manufacturing company. Your company has decided to open an assembly line in Mainland China and you are responsible for the design of this new assembly line. So, at the very beginning, you collected all the information concerned and start designing the assembly line accordingly. One month later, 3 possible alternatives designs come out and you and your boss thought that further analysis have to be done in order to decide which should be employed. After having considered several common analytical techniques, you decided to use Simulation.

The primary motivation for the use of simulation models rather than analytical techniques was related to the level of detail required to describe the operations adequately. An analytical approach was not well suited to the solution of this problem because of the stochastic nature of the duration and yields for the some operations inside the assembly system. In addition, you are interested in studying the dynamic behaviour of the new assembly line, which precluded the use of the analytical techniques that yield steady-state results.
4.2 Product Information

The products to be assembled are the housing of the automobile which is a new design and will be appear in the market in the early 1997. Since the product variety for the design of the automobile is great, their manufacturing procedures and the use of material of the housing are not necessary all the same. In order to maximise the production efficiency, all the subassemblies of the automobile have been classified into different families with respect to the use of material and the manufacturing procedures. So, the housing going to be assembled in the new assembly line are belong a family which contain 2 different types of housing with similar assembling procedures.

Since this automobile type is the new design in your company, the production forecast for the next 2 years are not so certain. It is possible that the production volume will be increase very fast once the first batch appear in the market, so you have to ensure that the new line should be capable of expanding the production capacity in a very short period once it is happened. But at this moments no one can say for certain this situation will definitely occur.
4.3 The 3 Proposed Designs

After the first round consideration with your boss, 3 possible designs of the new assembly line come out. The main difference of the 3 designs are in the use of material handling equipment for transferring the parts between stations, they will be discussed one by one. Let first consider the flow of the workpieces among the workstations in the assembly system, i.e. the automobil housing named Part1 and Part2 and it is depicted in Figure 1.

Figure 1. The Flow of Part1 and Part2
Part 1 entering the system are placed at a staging area for transfer to Assembling Station. After the parts have completed processing at the first station, they are transferred to a paint station manned by a second worker, then to a packaging station where they are packed by a third worker, and then to a second staging area where they exit the system.

The time between the arrivals of Part 1 at the system is exponentially distributed with a mean of 28 minutes. The processing time at the Assembling is uniformly distributed between 21 and 25 minutes. The paint time is normally distributed with a mean of 22 minutes and a standard deviation of 4. The packing time follows a triangular distribution with a minimum of 20, mode of 22, and maximum of 26.

The second type of housings is Part 2 and they will be painted in different colour. 30 percent of the arriving housings are randomly designated as type of Part 2. The operation for both parts at the first Assembling station requires the same amount of time. The Paint and Pack time remain the same for Part 1; however, Part 2 required the addition of a different station NewPaint with a painting time that is normally distributed with a mean of 49 minutes and a standard deviation of 7 minutes. After the painting operation, Part 2 is transferred to the packing station and requires a packing time that follows a triangular distribution with a minimum value of 21, a mode of 23, and a maximum value of 26.
4.3.1 Free Path Transporter - Fork Truck

The first alternative for the assembly line is the use of Free Path Transporter, Fork Truck, as a material handling equipment in the assembly line. Part1 and Part2 are transferred by using two fork trucks that travel at an average speed of 150 feet per minute. The distances between the stations are provided in Figure 2. Both the drop-off and pickup points at a station are at the same physical location. Once the truck reached the pickup/drop-off station, it requires a contestant load/unload time of two minutes.

![Diagram](image)

**Figure 2. Assembly line with Free Path Transporters**
Let assume that all requests for transfer are handled on a first-come, first-served basis. This assumption seems realistic for the system being considered because all materials move in a forward direction with no looping or backtracking.

The performance measures of interest are the utilisation and work-in-process (WIP) at the Assembling, Paint, NewPaint, Packaging operations and the 2 Fork Trucks. We also want to collect statistics on the part flowtime, i.e., the total time a part spends in the system.
4.3.2 Guided Vehicles - AGVs

Consider the other alternative in Figure 3. The part flow and production times remain the same as the previous case, the only difference is that in this design AGVs transport the housings within the system. Assume that three AGVs are used, each having a travel velocity of 100 feet per minute.

![Figure 3. Assembly line with Guided Vehicles](image)
The numbers in the diamonds on the network identify intersections, and the connections between these intersections are labeled as links to be included in the system map. Intersection 12 lies on an off-loop, which is used as a vehicles staging area. When a vehicle has completed its task and there are no other requests for transport, the vehicles is sent to the staging area at Intersection 12 to await the next request. If more than one vehicle arrives at the staging area, they automatically accumulate along link 11, behind the vehicles already there. This simple method of control prevents an idle vehicle from blocking another vehicle that is attempting to carry out a transport. Also note that the system map only allows one-way travel, thus preventing vehicle deadlock. The one exception to one-way travel is the spur link for reaching the Enter are at Intersection 6. Temporary blocking can occur if a vehicle is loading or unloading a part at an operation and another vehicle needs to pass that operation. In this case, the second vehicle waits until the first vehicle has completed its task and has moved out of the way.

The performance measures of interest are the utilisation and work-in-process(WIP) at the Assembling, Paint, NewPaint and Packaging operations and the 3 AGVs. We also want to collect statistics on the part flowtime, i.e., the total time a part spends in the system.
4.3.3 Accumulating Conveyor - Power Roller Conveyor

In the last alternative, we replace the fork trucks and AGVs with power roller conveyor, which allows accumulation, to accomplish all part transfers as shown in Figure 4. One power roller conveyer named Main is now used to transfer Part1 through the entire system (Enter-Assembling-Paint-Pack-ExitS). A second power roller conveyer named Loop will be used to transfer Part2 from the Assembling to the NewPaint operation and then to Pack.

![Diagram of Assembling Line with Accumulating Conveyor]

Figure 4. Assembling Line with Accumulating Conveyor
Because this conveyor provides a buffer, we can eliminate the current buffer in front of each operation and use the conveyor until the operation become available. When a part stays on the conveyor waiting for the operation, it creates a local blockage. Both conveyors operate at the same speed, i.e., 50 feet per minute. The amount of space required to place each Part1 on the conveyor is 2 feet of conveyor and 3 feet for each Part2. When the parts accumulate, however, they require 1 foot less space: 1 foot for a Part1 and 2 feet for a Part2. The placement of Parts on the conveyor requires the additional space for clearance.

This time we want to examine more closely what would be happen when the time the conveyors fail, with an exponentially distributed time between failures with a mean of 55 minutes. The time to repair is represented by a normal distribution with a mean of 5 minutes and standard deviation of 1 minutes.

The performance measures of interest this time are changed. The previous statistics on number in queue prior to each operation have been deleted. These statistics represent the WIP at each station, however, in this case the WIP is maintained on the accumulating conveyors. So statistics wanted to be collected are the utilisation at the Assembling, Paint, NewPaint, Packaging operations, both conveyors and the total time a part spends in the system. In addition, the conveyor status, number of accumulating housing and the length of the accumulating housing on both conveyors are also of our interested.
5. Details Data for the models

Regarding to the 3 designs of the new assembly line, the values of all the system parameters have to be known in order to make any simulation model. Following are the systems parameters for the 3 design alternatives, which includes 3 types of information. They are workpiece related data, resources related data and simulation related data.

5.1 Workpiece Related Data

5.1.1. Arrival Rate:

Part1 and Part2 : EXPO (28,1)

5.1.2. Arriving Probability:

Part1 : 0.7
Part2 : 0.3

5.13 Workstations:

Enter
Assembling
Paint
NewPaint
Pack
ExitS
Staging (only for System with AGV)
5.1.4. Routing Sequence:

Part 1: Enter - Assembling - Paint - Pack - ExitS

Part 2: Enter - Assembling - NewPaint - Pack - ExitS

5.1.5. Processing time:

Part 1: Assembling: UNIF (21,25,2),
                 Paint: NORM (22,4,2),
                 Pack: TRIA (20,22,26,2)

Part 2: Assembling: UNIF (21,25,2),
                 NewPaint: NORM (49,7,2),
                 Pack: TRIA (21,23,26,2)

5.2 Resources Related Data

5.2.1. Name of Secondary Resources in:

Assembling: Worker,

Paint: Painter,

NewPaint: NewPainter

Pack: Packer
5.2.2. Name of the queue in each station:

- Assembling: AssemblingQ,
- Paint: PaintQ,
- NewPaint: NewPaintQ,
- Pack: PackQ

5.2.3. Distance between stations in the first alternative (Free transporter):

<table>
<thead>
<tr>
<th>(in feet)</th>
<th>Enter</th>
<th>Assembling</th>
<th>Paint</th>
<th>NewPaint</th>
<th>Pack</th>
<th>ExitS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter</td>
<td>-----</td>
<td>325</td>
<td>445</td>
<td>455</td>
<td>565</td>
<td>815</td>
</tr>
<tr>
<td>Assembling</td>
<td>-----</td>
<td>120</td>
<td>130</td>
<td>240</td>
<td></td>
<td>490</td>
</tr>
<tr>
<td>Paint</td>
<td>-----</td>
<td>250</td>
<td></td>
<td>120</td>
<td></td>
<td>370</td>
</tr>
<tr>
<td>NewPaint</td>
<td></td>
<td>-----</td>
<td>130</td>
<td></td>
<td></td>
<td>380</td>
</tr>
<tr>
<td>Pack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>250</td>
</tr>
<tr>
<td>ExitS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-----</td>
</tr>
</tbody>
</table>

5.2.4. Free Path Transporter:

- Name of transporter: Truck
- No. of transporter: 2
- Average speed of the transporter: 150 feet/minute
- Load/unload time: 2 minutes
- Rules used for request for transfer: first-come first-served
- Name of transporter queue: TruckQ
5.2.5. Guided Transporter:

Name of transporter: AGV
No. of transporter: 3
Average speed of the transporter: 50 feet/minute
Load/unload time: 2 minutes
Rules used for request for transfer: first-come first-served
Name of transporter queue: AGVQue

5.2.6. Intersections

<table>
<thead>
<tr>
<th>Intersection No.</th>
<th>Intersection I.D.</th>
<th>Travel length through intersection</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IntAssembling</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>IntPaint</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>IntPack</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>IntNewPaint</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>IntExit</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>IntEnter</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>Int7</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>Int8</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>Int9</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>Int10</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>Int11</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>IntStaging</td>
<td>1</td>
</tr>
</tbody>
</table>
### 5.2.7. Links

<table>
<thead>
<tr>
<th>No.</th>
<th>Link I.D.</th>
<th>Beginning Int.</th>
<th>Ending Int.</th>
<th>No. of zone</th>
<th>length of zone/feet</th>
<th>Link Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Link1</td>
<td>IntAssembling</td>
<td>Int7</td>
<td>8</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>2</td>
<td>Link2</td>
<td>Int7</td>
<td>IntPaint</td>
<td>4</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>3</td>
<td>Link3</td>
<td>Int7</td>
<td>IntNewPaint</td>
<td>4</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>4</td>
<td>Link4</td>
<td>IntNewPaint</td>
<td>Int8</td>
<td>8</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>5</td>
<td>Link5</td>
<td>IntPaint</td>
<td>Int8</td>
<td>6</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>6</td>
<td>Link6</td>
<td>Int8</td>
<td>IntPack</td>
<td>7</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>7</td>
<td>Link7</td>
<td>IntNewPaint</td>
<td>IntExit</td>
<td>25</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>8</td>
<td>Link8</td>
<td>IntExit</td>
<td>Int9</td>
<td>22</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>9</td>
<td>Link9</td>
<td>Int8</td>
<td>Int9</td>
<td>10</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>10</td>
<td>Link10</td>
<td>Int9</td>
<td>Int10</td>
<td>11</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>11</td>
<td>Link11</td>
<td>Int9</td>
<td>IntStaging</td>
<td>10</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>12</td>
<td>Link12</td>
<td>IntStaging</td>
<td>Int10</td>
<td>3</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>13</td>
<td>Link13</td>
<td>Int10</td>
<td>Int7</td>
<td>8</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>14</td>
<td>Link14</td>
<td>Int10</td>
<td>Int11</td>
<td>4</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>15</td>
<td>Link15</td>
<td>Int11</td>
<td>IntEnter</td>
<td>23</td>
<td>10</td>
<td>Spur</td>
</tr>
<tr>
<td>16</td>
<td>Link16</td>
<td>Int11</td>
<td>IntAssembling</td>
<td>9</td>
<td>10</td>
<td>unidirectional</td>
</tr>
</tbody>
</table>
### 5.2.8. Conveyor

<table>
<thead>
<tr>
<th></th>
<th>Conveyor 1</th>
<th>Conveyor 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Name</strong></td>
<td>Main</td>
<td>Loop</td>
</tr>
<tr>
<td><strong>2. Segments</strong></td>
<td>Enter-Assembling- 325</td>
<td>Assembling-NewPaint-130</td>
</tr>
<tr>
<td></td>
<td>Assembling-Paint-120</td>
<td>NewPaint-Exit-130</td>
</tr>
<tr>
<td></td>
<td>Paint-Pack-120</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pack-Exit-250</td>
<td></td>
</tr>
<tr>
<td><strong>3. Velocity</strong></td>
<td>50 feet / min</td>
<td>50 feet / min</td>
</tr>
<tr>
<td><strong>4. Length of each conveyor cell</strong></td>
<td>1 foot</td>
<td>1 foot</td>
</tr>
<tr>
<td><strong>5. Initial Status</strong></td>
<td>Active</td>
<td>Active</td>
</tr>
<tr>
<td><strong>6. Max no.of cells occupied by any Part</strong></td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td><strong>7. Mean Time between Failure</strong></td>
<td>EXPO(55,3)</td>
<td>EXPO(55,3)</td>
</tr>
<tr>
<td><strong>8. Repair time</strong></td>
<td>NORM(5,1,4)</td>
<td>NORM(5,1,4)</td>
</tr>
<tr>
<td><strong>9. Type of conveyor</strong></td>
<td>Accumulating</td>
<td>Accumulating</td>
</tr>
</tbody>
</table>
5.3 Simulation Related Data

5.3.1 Performance Measures of 1st Design

1. Utilisation of Worker, Painter, NewPainter, Packer.

2. Utilisation of 2 Fork Trucks.

3. Work-in-process(WIP) at the queue for Assembling, Paint, Newpaint, Packaging station.

4. Work-in-process(WIP) at the queue for 2 Fork Trucks.

5. The total time a part spends in the system.

5.3.2. Performance Measures of 2nd Design

1. Utilisation of Worker, Painter, NewPainter, Packer.

2. Utilisation of 3 AGVs.

3. Work-in-process(WIP) at the queue for Assembling, Paint, Newpaint, Packaging station.

4. Work-in-process(WIP) at the queue for the 3 AGVs.

5. The total time a part spends in the system.
5.3.3. Performance Measures of 3rd Design

1. Utilisation of Worker, Painter, NewPainter, Packer.
2. Utilisation of the 2 accumulating conveyors.
3. Number of parts on both conveyors.
4. Length of parts on both conveyors.
5. Both conveyor status. (active / inactive / idle / busy)
6. The total time a part spends in the system.

5.3.4. Experimentation

1. Number of replication : 1
2. Start Time : 0
3. End Time : 700000
4. Warm-up Period : 20000
6. SIMAN Program Files & Collect Statistical Results

Once the output files have been generated from QSMS, the next step is to "Compile" the two source files into object files. Then we "Link" the two resulting object files into a Program File. The program file contains model- and experiment- related information in a form that can be read and executed by the ARENA program.

Suppose your model file is named abc.mod and experiment file named abc.exp, followings are the steps by which the program file can be generated:

1. Make sure ARENA have been installed in your computer.
2. Go to MS-DOS operating system
3. Type MODEL abc.mod abc.m"
4. Type EXPMT abc.exp abc.e”
5. Type “ LINKER abc.m abc.e abc.p”
6. Type SIMAN abc.p”

Then you will discovered that a statistical result appear in the screen. Do you remember that you have specified a file, which is named with extension at”, for output statistics ? Collect the output statistics from that file.
7. Your Tasks

7.1 Study this Handout

Please spend about 30 minutes to study this handout carefully. Make sure you understand what you are going to do.

7.2 Build 3 simulation models

You are provided with a Disk named "QSMS.pas". Put it into A drive of your computer and go to Turbo Pascal. Retrieve the file and press Ctrl F9. Now answer all questions carefully, *you can obtain most of the modeling data in this handout Ch4 and Ch5*. Based on the background information in Ch.4 and details data for the designs in Ch.5, build three simulation models by QSMS:

Design 1: Assembly line with Free Path Transporter - Fork Truck,

Design 2: Assembly line with Guided Vehicles - AGVs,

Design 3: Assembly line with Accumulating Conveyor - Power Roller Conveyor

7.3 Run the models in Arena

According the instruction in Ch.6, run the corresponding model and experiment source files of each of the design alternatives in ARENA. Then collect and you statistical results.

7.4. Questionnaires

Do the Questionnaires in CH. 8 as a evaluation of this exercise.
8. Questionnaires

Section I : Personal Information

Name : ________________

Student ID : ________________

Dept. / Year : ________________

Section II : Background about your skills in Simulation

1. Have you learnt simulation before ? (Please “ ✓”)

    Yes ( )    No ( )

*If yes,*

1.1 Which simulation software you have learnt ? Please fill in one or two simulation software you are the most familiar.

    1. ____________________________

    2. ____________________________

1.2 How many hours you have taken to learn this software ?

    1. ____________________________

    2. ____________________________
1.3 Do you know these software well? (Please “✓”)

For 1, excellent ( ) good ( ) fair ( ) very little ( )
For 2, excellent ( ) good ( ) fair ( ) very little ( )

Section III: Comments about using QSMS.

1. The total time it takes to finish this exercise. (Please “✓”)

( ) Less than 1 hour
( ) 1-2 hours
( ) 2-3 hours
( ) 3-4 hours
( ) 4-6 hours
( ) more than 6 hours
2. Do you think it is easy to you QSMS? (Please "✔")

( ) Very Easy
( ) Easy
( ) Fair
( ) Quite Difficult
( ) Difficult

3. Please advise what else can be improved about QSMS?

Comment: _____________________________________________
_________________________ ____________________________
_________________________ ____________________________
_________________________ ____________________________
_________________________ ____________________________
_________________________ ____________________________
_________________________ ____________________________
Appendix B: Exercise B

1. Introduction

2. Objectives

3. Background information for the case

   3.1 You are a Manufacturing Engineer......

   3.2 Product Information

   3.3 The 3 Proposed Designs

      3.3.1 Free Path Transporter - Fork Truck

      3.3.2 Guided Vehicles - AGVs

      3.3.3 Accumulating Conveyor - Power Roller Conveyor

4. Details Data for the models

   4.1 Workpiece Related Data

   4.2 Resources Related Data

   4.3 Simulation Related Data

5. Your Tasks

   5.1 Study this Handout

   5.2 Build 3 simulation models

   5.3 Run the models in Arena

   5.4. Questionnaires

6. Questionnaires
1. Introduction

Nowadays, simulation plays very important role in the area of system design since it provides strong and effective way for managers or engineers to measure the performance of their system design. Its allow the construction of complex and realistic representations of actual system. Through statistics-collection or real-time observation, great insight can be gained into the operation of the system. Therefore, simulation is a tool to observe performance of different system design alternatives and to discover all potential problems of an assembly system before real implementation, thus major modification to a large capital investment after installation can be avoided.

Even though simulation has been demonstrated that it is very useful in production start-up and optimisation, it also has some disadvantages which seriously limits its utilisation. Firstly, developing the simulation model is not quick and requires a simulation specialist. The more experience the person doing the modeling has, the faster the models can be built. Even though it doesn't require a very long model building process like in the case of building a physical model, it does take training and practice before a beginner can perform moderately complicated simulation with a commercially available simulation language. Simulation expertise can be obtained only after months of intensive training and practice in a simulation language, such as SLAM or SIMAN.
Even for an experienced analyst, modeling and programming of a complicated system can be a very time consuming process. Modeling, programming and debugging in simulation analysis usually take longer than most companies can tolerate. A simple facility would probably take a minimum of two weeks of programming, verification, and validation. The time consuming part of developing the model is to establish the operation logic. Surprising, the logic for operating the facility is normally one of the last things defined.

In order to maximise the possible advantages we can gain in the simulation studies, we have to overcome those limitations we have mentioned above. One of the most possible method is to provide a user interface by which user can skip all the programming and debugging process. Therefore, a user interface, named Quick Simulation Modeling System, was specially designed for modeling all type of assembly systems. It includes all type of assembly line's operation logic. QSMS's users, even have no knowledge about simulation, can model various assembly lines very quickly simply by selecting one of the model logic and preparing the input data only. Time and effort required for simulation modeling will become less. High quality of the model operation logic of various type of assembly system can be ensured and statistical results can be obtained rapidly for making management decisions.
2. Objectives

As you are not a beginner in simulation, the objective of carrying out this simulation exercise is:

*To see how many hours should be taken for completing this exercise with your most familiar Software Package.*
3. Background information for the case

3.1 You are a Manufacturing Engineer......

Supposing you are a manufacturing engineer in a automobile manufacturing company. Your company has decided to open an assembly line in Mainland China and you are responsible for the design of this new assembly line. So, at the very beginning, you collected all the information concerned and start designing the assembly line accordingly. One month later, 3 possible alternatives designs come out and you and your boss thought that further analysis have to be done in order to decide which should be employed. After having considered several common analytical techniques, you decided to use Simulation.

The primary motivation for the use of simulation models rather than analytical techniques was related to the level of detail required to describe the operations adequately. An analytical approach was not well suited to the solution of this problem because of the stochastic nature of the duration and yields for the some operations inside the assembly system. In addition, you are interested in studying the dynamic behaviour of the new assembly line, which precluded the use of the analytical techniques that yield steady-state results.
3.2 Product Information

The products to be assembled are the housing of the automobile which is a new design and will be appear in the market in the early 1997. Since the product variety of the design of the automobile is great, their manufacturing procedures and the use of material of the housing are not necessary all the same. In order to maximise the production efficiency, all the subassemblies of the automobile have been classified into different families with respect to the use of material and the manufacturing procedures. So, the housing going to be assembled in the new assembly line are belong a family which contain 2 different types of housing with similar assembling procedures.

Since this automobile type is the new design in your company, the production forecast for the next 2 years are not so certain. It is possible that the production volume will be increase very fast once the first batch appear in the market, so you have to ensure that the new line should be capable of expanding the production capacity in a very short period once it is happened. But at this moments no one can say for certain this situation will definitely occur.
3.3 The 3 Proposed Designs

Your 3 possible designs of the new assembly line will be explained one by one. Since the main difference of the 3 designs are the use of material handling equipment for transferring the parts between stations, let first consider the flow of the workplaces, i.e. the housing named Part1 and Part2 and it is depicted in Figure 1.

![Diagram of assembly line flow](image)

**Figure 1. The Flow of Part1 and Part2**
Part 1 entering the system are placed at a staging area for transfer to Assembling Station. After the parts have completed processing at the first station, they are transferred to a paint station manned by a second worker, then to a packaging station where they are packed by a third worker, and then to a second staging area where they exit the system.

The time between the arrivals of Part 1 at the system is exponentially distributed with a mean of 28 minutes. The processing time at the Assembling is uniformly distributed between 21 and 25 minutes. The paint time is normally distributed with a mean of 22 minutes and a standard deviation of 4. The packing time follows a triangular distribution with a minimum of 20, mode of 22, and maximum of 26.

The second type of housings is Part 2 and they will be painted in different colour. 30 percent of the arriving housings are randomly designated as type of Part 2. The operation for both parts at the first Assembling station requires the same amount of time. The Paint and Pack time remain the same for Part 1; however, Part 2 required the addition of a different station NewPaint with a painting time that is normally distributed with a mean of 49 minutes and a standard deviation of 7 minutes. After the painting operation, Part 2 is transferred to the packing station and requires a packing time that follows a triangular distribution with a minimum value of 21, a mode of 23, and a maximum value of 26.
3.3.1 Free Path Transporter - Fork Truck

The first alternative for the assembly line is the use of Free Path Transporter, Fork Truck, as a material handling equipment in the assembly line. Part1 and Part2 are transferred by using two fork trucks that travel at an average speed of 150 feet per minute. The distances between the stations are provided in Figure 2. Both the drop-off and pickup points at a station are at the same physical location. Once the truck reached the pickup/drop-off station, it requires a contestant load/unload time of two minutes.

Figure 2. Assembly line with Free Path Transporters
Let assume that all requests for transfer are handled on a first-come, first-served basis. This assumption seems realistic for the system being considered because all materials move in a forward direction with no looping or backtracking.

The performance measures of interest are the utilisation and work-in-process (WIP) at the Assembling, Paint, Newpaint, Packaging operations and the 2 Fork Trucks. We also want to collect statistics on the part flowtime, i.e., the total time a part spends in the system.
3.3.2 Guided Vehicles - AGVs

Consider the other alternative in Figure 3. The part flow and production times remain the same as the previous case, the only difference is that in this design AGVs transport the housings within the system. Assume that three AGVs are used, each having a travel velocity of 100 feet per minute.

Figure 3. Assembly line with Guided Vehicles
The numbers in the diamonds on the network identify intersections, and the connections between these intersections are labeled as links to be included in the system map. Intersection 12 lies on an off-loop, which is used as a vehicles staging area. When a vehicle has completed its task and there are no other requests for transport, the vehicle is sent to the staging area at Intersection 12 to await the next request. If more than one vehicle arrives at the staging area, they automatically accumulate along link 11, behind the vehicles already there. This simple method of control prevents an idle vehicle from blocking another vehicle that is attempting to carry out a transport. Also note that the system map only allows one-way travel, thus preventing vehicle deadlock. The one exception to one-way travel is the spur link for reaching the Enter area at Intersection 6. Temporary blocking can occur if a vehicle is loading or unloading a part at an operation and another vehicle needs to pass that operation. In this case, the second vehicle waits until the first vehicle has completed its task and has moved out of the way.

The performance measures of interest are the utilisation and work-in-process (WIP) at the Assembling, Paint, NewPaint and Packaging operations and the 3 AGVs. We also want to collect statistics on the part flowtime, i.e., the total time a part spends in the system.
3.3.3 Accumulating Conveyor - Power Roller Conveyor

In the last alternative, we replace the fork trucks and AGVs with power roller conveyor, which allows accumulation, to accomplish all part transfers as shown in Figure 4. One power roller conveyor is now used to transfer Part1 through the entire system (Enter-Assembling-Paint-Pack-Exitsystem). A second power roller conveyor will be used to transfer Part2 from the Assembling to the NewPaint operation and then to Pack.

![Diagram of Accumulating Conveyor](image)

**Figure 4. Assembling Line with Accumulating Conveyor**
Because this conveyor provides a buffer, we can eliminate the current buffer in front of each operation and use the conveyor until the operation become available. When a part stays on the conveyor waiting for the operation, it creates a local blockage. Both conveyors operate at the same speed, i.e., 50 feet per minute. The amount of space required to place each Part1 on the conveyor is 2 feet of conveyor and 3 feet for each Part2. When the parts accumulate, however, they require 1 foot less space: 1 foot for a Part1 and 2 feet for a Part2. The placement of Parts on the conveyor requires the additional space for clearance.

This time we want to examine more closely what would be happen when the time the conveyors fail, with an exponentially distributed time between failures with a mean of 55 minutes. The time to repair is represented by a normal distribution with a mean of 5 minutes and standard deviation of 1 minutes.

The performance measures of interest this time are changed. The previous statistics on number in queue prior to each operation have been deleted. These statistics represent the WIP at each station, however, in this case the WIP is maintained on the accumulating conveyors. So statistics wanted to be collected are the utilisation at the Assembling, Paint, NewPaint, Packaging operations, both conveyors and the total time a part spends in the system. In addition, the conveyor status, number of accumulating housing and the length of the accumulating housing on both conveyors are also of our interested.
4. Details Data for the models

Regarding to the 3 designs of the new assembly line, the values of all the system parameters have to be known in order to make any simulation model. Following are the systems parameters for the 3 design alternatives, which includes 3 types of information. They are workpiece related data, resources related data and simulation related data.

4.1 Workpiece Related Data

4.1.1. Arrival Rate:

Part1 and Part2 : EXPO (28,1)

4.1.2. Arriving Probability:

Part1 : 0.7
Part2 : 0.3

4.13 Workstations:

Enter
Assembling
Paint
NewPaint
Pack
ExitS
Staging (only for System with AGV)
4.1.4. Routing Sequence:

Part 1: Enter - Assembling - Paint - Pack - ExitS

Part 2: Enter - Assembling - NewPaint - Pack - ExitS

4.1.5. Processing time:

            Paint     : NORM (22,4,2),
            Pack      : TRIA (20,22,26,2)

            NewPaint  : NORM (49,7,2),
            Pack      : TRIA (21,23,26,2)

4.2 Resources Related Data

4.2.1. Name of Secondary Resources in:

Assembling : Worker,
Paint        : Painter,
NewPaint     : NewPainter
Pack         : Packer
4.2.2. Name of the queue in each station:

Assembling: AssemblingQ,
Paint: PaintQ,
NewPaint: NewPaintQ,
Pack: PackQ

4.2.3. Distance between stations in the first alternative (Free transporter):

<table>
<thead>
<tr>
<th>(in feet)</th>
<th>Enter</th>
<th>Assembling</th>
<th>Paint</th>
<th>NewPaint</th>
<th>Pack</th>
<th>ExitS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter</td>
<td>------</td>
<td>325</td>
<td>445</td>
<td>455</td>
<td>565</td>
<td>815</td>
</tr>
<tr>
<td>Assembling</td>
<td>------</td>
<td>120</td>
<td>130</td>
<td>240</td>
<td>490</td>
<td></td>
</tr>
<tr>
<td>Paint</td>
<td>------</td>
<td>250</td>
<td>120</td>
<td>370</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NewPaint</td>
<td>------</td>
<td>130</td>
<td>380</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pack</td>
<td>------</td>
<td>250</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ExitS</td>
<td>------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.2.4. Free Path Transporter:

Name of transporter: Truck
No. of transporter: 2
Average speed of the transporter: 150 feet / minute
Load/unload time: 2 minutes
Rules used for request for transfer: first-come first-served
Name of transporter queue: TruckQ
4.2.5. Guided Transporter:

Name of transporter : AGV

No. of transporter : 3

Average speed of the transporter : 50 feet / minute

Load/unload time : 2 minutes

Rules used for request for transfer : first-come first-served

Name of transporter queue : AGVQue

4.2.6. Intersections

<table>
<thead>
<tr>
<th>Intersection No.</th>
<th>Intersection I.D.</th>
<th>Travel length through intersection</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IntAssembling</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>IntPaint</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>IntPack</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>IntNewPaint</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>IntExit</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>IntEnter</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>Int7</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>Int8</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>Int9</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>Int10</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>Int11</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>IntStaging</td>
<td>1</td>
</tr>
</tbody>
</table>
### 4.2.7. Links

<table>
<thead>
<tr>
<th>No.</th>
<th>Link I.D.</th>
<th>Beginning Int.</th>
<th>Ending Int.</th>
<th>No. of zone</th>
<th>length of zone/feet</th>
<th>Link Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Link1</td>
<td>IntAssembling</td>
<td>Int7</td>
<td>8</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>2</td>
<td>Link2</td>
<td>Int7</td>
<td>IntPaint</td>
<td>4</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>3</td>
<td>Link3</td>
<td>Int7</td>
<td>IntNewPaint</td>
<td>4</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>4</td>
<td>Link4</td>
<td>IntNewPaint</td>
<td>Int8</td>
<td>8</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>5</td>
<td>Link5</td>
<td>IntPaint</td>
<td>Int8</td>
<td>6</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>6</td>
<td>Link6</td>
<td>Int8</td>
<td>IntPack</td>
<td>7</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>7</td>
<td>Link7</td>
<td>IntNewPaint</td>
<td>IntExit</td>
<td>25</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>8</td>
<td>Link8</td>
<td>IntExit</td>
<td>Int9</td>
<td>22</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>9</td>
<td>Link9</td>
<td>Int8</td>
<td>Int9</td>
<td>10</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>10</td>
<td>Link10</td>
<td>Int9</td>
<td>Int10</td>
<td>11</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>11</td>
<td>Link11</td>
<td>Int9</td>
<td>IntStaging</td>
<td>10</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>12</td>
<td>Link12</td>
<td>IntStaging</td>
<td>Int10</td>
<td>3</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>13</td>
<td>Link13</td>
<td>Int10</td>
<td>Int7</td>
<td>8</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>14</td>
<td>Link14</td>
<td>Int10</td>
<td>Int11</td>
<td>4</td>
<td>10</td>
<td>unidirectional</td>
</tr>
<tr>
<td>15</td>
<td>Link15</td>
<td>Int11</td>
<td>IntEnter</td>
<td>23</td>
<td>10</td>
<td>Spur</td>
</tr>
<tr>
<td>16</td>
<td>Link16</td>
<td>Int11</td>
<td>IntAssembling</td>
<td>9</td>
<td>10</td>
<td>unidirectional</td>
</tr>
</tbody>
</table>
### 4.2.8. Conveyor

<table>
<thead>
<tr>
<th></th>
<th>Conveyor 1</th>
<th>Conveyor 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Name</strong></td>
<td>Main</td>
<td>Loop</td>
</tr>
<tr>
<td><strong>2. Segments</strong></td>
<td>Enter-Assembling- 325</td>
<td>Assembling-NewPaint-130</td>
</tr>
<tr>
<td></td>
<td>Assembling-Paint-120</td>
<td>NewPaint-Exit-130</td>
</tr>
<tr>
<td></td>
<td>Paint-Pack-120</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pack-Exit-250</td>
<td></td>
</tr>
<tr>
<td><strong>3. Velocity</strong></td>
<td>50 feet / min</td>
<td>50 feet / min</td>
</tr>
<tr>
<td><strong>4. Length of each conveyor cell</strong></td>
<td>1 foot</td>
<td>1 foot</td>
</tr>
<tr>
<td><strong>5. Initial Status</strong></td>
<td>Active</td>
<td>Active</td>
</tr>
<tr>
<td><strong>6. Max no.of cells occupied by any Part</strong></td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td><strong>7. Mean Time between Failure</strong></td>
<td>EXPO(55,3)</td>
<td>EXPO(55,3)</td>
</tr>
<tr>
<td><strong>8. Repair time</strong></td>
<td>NORM(5,1,4)</td>
<td>NORM(5,1,4)</td>
</tr>
<tr>
<td><strong>9. Type of conveyor</strong></td>
<td>Accumulating</td>
<td>Accumulating</td>
</tr>
</tbody>
</table>
4.3 Simulation Related Data

4.3.1 Performance Measures of 1st Design

1. Utilisation of Worker, Painter, NewPainter, Packer.

2. Utilisation of 2 Fork Trucks.

3. Work-in-process (WIP) at the queue for Assembling, Paint, Newpaint, Packaging station.

4. Work-in-process (WIP) at the queue for 2 Fork Trucks.

5. The total time a part spends in the system.

4.3.2. Performance Measures of 2nd Design

1. Utilisation of Worker, Painter, NewPainter, Packer.

2. Utilisation of 3 AGVs.

3. Work-in-process (WIP) at the queue for Assembling, Paint, Newpaint, Packaging station.

4. Work-in-process (WIP) at the queue for the 3 AGVs.

5. The total time a part spends in the system.
4.3.3. Performance Measures of 3rd Design

1. Utilisation of Worker, Painter, NewPainter, Packer.
2. Utilisation of the 2 accumulating conveyors.
3. Number of parts on both conveyors.
4. Length of parts on both conveyors.
5. Both conveyor status. (active / inactive / idle / busy)
6. The total time a part spends in the system.

4.3.4. Experimentation

1. Number of replication : 1
2. Start Time : 0
3. End Time : 700000
4. Warm-up Period : 20000
5. Your Tasks

5.1 Study this Handout

Please spend about 30 minutes to study this handout carefully. Make sure you understand what you are going to do.

5.2 Build 3 simulation models

Based on the background information in Ch.3 and details data for the designs in Ch.4, carry out three simulation studies by the provided software, the 3 studies are:

Design 1: Assembly line with Free Path Transporter - Fork Truck,
Design 2: Assembly line with Guided Vehicles - AGVs,
Design 3: Assembly line with Accumulating Conveyor - Power Roller Conveyor

5.3 Questionnaires

Do the Questionnaires in CH. 6 as a evaluation of this exercise.
6. Questionnaires

Section I : Personal Information

Name : ______________
Student ID : ______________
Dept. / Year : ______________

Section II : Background about your skills in Simulation

1. Have you learnt simulation before ? (Please “✓”)

   Yes ( )       No( )

If yes,

1.1 Which simulation software you have learnt ? Please fill in one or two simulation software you are the most familiar.

1. _______________________
2. _______________________

1.2 How many hours you have taken to learn this software ?

1. _______________________
2. _______________________
1.3 Do you know these software well? (Please "✓")

For 1, excellent () good () fair () very little ()
For 2, excellent () good () fair () very little ()

Section III: Comments about using your software.

1. The total time it takes to finish this exercise. (Please "✓")

() Less than 1 hour
() 1-2 hours
() 2-3 hours
() 3-4 hours
() 4-6 hours
() more than 6 hours
2. Please comment the difficulties you have encountered in doing this exercise.

Comment: ________________________________________________________________

____________________________________

____________________________________

____________________________________

____________________________________

____________________________________

____________________________________
8. Questionnaires

Section I: Personal Information

Name: LEE NGAR YEE RITA
Student ID: 94131500
Dept. / Year: IEEM Yr. 3

Section II: Background about your skills in Simulation

1. Have you learnt simulation before? (Please "✓")
   Yes ✓ No □

If yes,

1.1 Which simulation software you have learnt? Please fill in one or two simulation software you are the most familiar.

1. Promodel
2. __________________________

1.2 How many hours you have taken to learn this software?

1. 150 hours
2. __________________________
1.3 Do you know these software well? (Please "✔")

For 1, excellent ☐ good ☐ fair✔ very little ☐
For 2, excellent ☐ good ☐ fair ☐ very little ☐

Section III: Comments about using QSMS.

1. The total time it takes to finish this exercise. (Please "✔")

☐ Less than 1 hour
✔ 1-2 hours
☐ 2-3 hours
☐ 3-4 hours
☐ 4-6 hours
☐ more than 6 hours
2. Do you think it is easy to you QSMS? (Please “✓“)

☐ Very Easy
☑ Easy
☐ Fair
☐ Quite Difficult
☐ Difficult

3. Please advise what else can be improved about QSMS?

Comment: It would be better for the user to correct any data input in the program about the interface, let the user to confirm his/her input by clicking <ENTER>.
8. Questionnaires

Section I : Personal Information

Name : [Handwritten Name]
Student ID : [Handwritten ID]
Dept. / Year : [Handwritten Dept. / Year]

Section II : Background about your skills in Simulation

1. Have you learnt simulation before? (Please "✓")
   
   Yes ✓ No □

   If yes,
   
   1.1 Which simulation software you have learnt? Please fill in one or two simulation software you are the most familiar.
   
   1. [Handwritten Software 1]
   2. [Handwritten Software 2]

   1.2 How many hours you have taken to learn this software?
   
   1. [Handwritten Hours 1]
   2. [Handwritten Hours 2]
1.3 Do you know these software well? (Please "✓")

For 1, excellent [ ] good [ ] fair [ ] very little [ ]
For 2, excellent [ ] good [ ] fair [ ] very little [ ]

Section III: Comments about using QSMS.

1. The total time it takes to finish this exercise. (Please "✓")

[ ] Less than 1 hour
[ ] 1-2 hours
[ ] 2-3 hours
[ ] 3-4 hours
[ ] 4-6 hours
[ ] more than 6 hours
2. Do you think it is easy to you QSMS? (Please "✓")

Very Easy
Easy
Fair ✓
Quite Difficult
Difficult

3. Please advise what else can be improved about QSMS?

Comment: For user with no simulation or manufacturer process experience, more explanation/guideline needed in join to the terms in QSURS.

Include some features that can allow users to avoid incorrect entries.
8. Questionnaires

Section I: Personal Information

Name: HO CHAU TONG

Student ID: 94192528

Dept. / Year: IEMS YR. 3

Section II: Background about your skills in Simulation

1. Have you learnt simulation before? (Please “✓”)
   
   Yes ✓
   No □

If yes,

1.1 Which simulation software you have learnt? Please fill in one or two simulation software you are the most familiar.

1. PRONODEL

2. 

1.2 How many hours you have taken to learn this software?

1. ≥ 10 hrs.

2. 

154
1.3 Do you know these software well? (Please "✓")

For 1, excellent □  good ✓  fair □  very little □
For 2, excellent □  good □  fair □  very little □

Section III: Comments about using QSMS.

1. The total time it takes to finish this exercise. (Please "✓")

☐ Less than 1 hour
✓ 1-2 hours
☐ 2-3 hours
☐ 3-4 hours
☐ 4-6 hours
☐ more than 6 hours
2. Do you think it is easy to you QSMS? (Please “✓”)

☐ Very Easy
☐ Easy
☒ Fair
☐ Quite Difficult
☐ Difficult

3. Please advise what else can be improved about QSMS?

Comment:
6. Questionnaires

Section I : Personal Information

Name : WONG KA YIU RUDOLF
Student ID : 94093461
Dept. / Year : IEEI - Yr 3

Section II : Background about your skills in Simulation

1. Have you learnt simulation before ? (Please “✓”)
   Yes ✓ No

If yes,

1.1 Which simulation software you have learnt ? Please fill in one or two simulation
   software you are the most familiar.

   1. Pro-Model
   2. ______________________

1.2 How many hours you have taken to learn this software ?

   1. Learn 10 hrs practice 20 hrs
   2. ______________________
1.3 Do you know these software well? (Please “✓”)

For 1,  excellent  good  fair  ✓  very little
For 2,  excellent  good  fair  very little

Section III: Comments about using QSMS.

1. The total time it takes to finish this exercise. (Please “✓”)

Less than 1 hour
1-2 hours
2-3 hours
3-4 hours  ✓
4-6 hours
more than 6 hours
2. Please comment the difficulties you have encountered in doing this exercise.

**Comment:**

Case 1: Don't know how to set two parts arrive at the same location at different t.

Case 2: Don't know how to deal with different processing time of different part at 'pack' station.

1. Solved by 0.3 output of station 'enter' as part 2, 0.7 output of station 'enter' as part 1.

2. Solved by setting two packing stations namely 'PACK' and 'NewPack' which both share 1 user and the distance between two stations is zero.

Case 3: Don't know how to build up the main conveyor and loop conveyor.

2. Don't know how to set space between each entities.

In running the experiment, the end time is too large.
6. Questionnaires

Section I : Personal Information

Name : Ho Yiu Wai
Student ID : 94152205
Dept. / Year : IEEM Yr. 3

Section II : Background about your skills in Simulation

1. Have you learnt simulation before ? (Please “ ✓”)
   Yes ✓ No ☒

If yes,

1.1 Which simulation software you have learnt ? Please fill in one or two simulation software you are the most familiar.

1. Prolanad

2. _______________________

1.2 How many hours you have taken to learn this software ?

1. 3 hours

2. _______________________
1.3 Do you know these software well? (Please "✓")

For 1, excellent □ good □ fair □ very little □
For 2, excellent □ good □ fair □ very little □

Section III: Comments about using CIMS.

1. The total time it takes to finish this exercise. (Please "✓")

Less than 1 hour
1-2 hours
2-3 hours
3-4 hours
4-6 hours
more than 6 hours
2. Please comment the difficulties you have encountered in doing this exercise.

Comment: 1) I have read the tutorial, but the examples are not enough. If there is more detailed information and explanation, I think I would know it better.

2) During the running process, I clearly observed there were some mistakes. But I just did not know how I can fix it.
6. Questionnaires

Section I : Personal Information

Name : LAM YW. TYNQ

Student ID : 9484474

Dept. / Year : E2M Y: 3

Section II : Background about your skills in Simulation

1. Have you learnt simulation before ? (Please “✓”)

Yes ✓ No ✗

If yes,

1.1 Which simulation software you have learnt ? Please fill in one or two simulation software you are the most familiar.

1. ___________

2. ___________

1.2 How many hours you have taken to learn this software ?

1. ___________

2. ___________
1.3 Do you know these software well? (Please "✔")

For 1. excellent ✔ good ✓ fair ✗ very little ✗
For 2. excellent ✗ good ☑ fair ✗ very little ✗

Section III: Comments about using QSMS.

I. The total time it takes to finish this exercise. (Please "✔")

Less than 1 hour
1-2 hours
2-3 hours
3-4 hours
4-6 hours
✔ more than 6 hours
2. Please comment the difficulties you have encountered in doing this exercise.

Comment: 1. The phases of activities have to be clearly understood prior to building the model.  
   2. No vey appropriate items available for building model.
APPENDIX

1. SIMAN's MODEL FILE : Free Path Transporter

BEGIN.
CREATE.
ASSIGN:
EXPO(28.1):
MARK(TimeIn):
M=Enter:
NS=DISC(0.7, Part1, 1, Part2, 10):
GetTruck QUEUE.
REQUEST:
DELAY:
TRANSPORT:
truckQ:
2.0:
truck, SEQ:
STATION,
DELAY:
FREE:
QUEUE,
SEIZE:
DELAY:
RELEASE:
Assembling:
2.0:
Truck:
AssemblingQ:
Worker:
OpTime:
Worker: NEXT(GetTruck):
STATION,
DELAY:
FREE:
QUEUE,
SEIZE:
DELAY:
RELEASE:
Paint:
2.0:
Truck:
PaintQ:
Painter:
OpTime:
Painter: NEXT(GetTruck):
STATION,
DELAY:
FREE:
QUEUE,
SEIZE:
DELAY:
RELEASE:
Pack:
2.0:
Truck:
PackQ:
Packer:
OpTime:
Packer: NEXT(GetTruck):
STATION,
DELAY:
FREE:
QUEUE,
SEIZE:
DELAY:
RELEASE:
NewPaint:
2.0:
Truck:
NewPaintQ:
NewPainter:
OpTime:
NewPainter: NEXT(GetTruck):
STATION,
DELAY:
FREE:
TALLY:
DISPOSE:
ExitS:
2.0:
Truck:
Flowtime, INT(TimeIn):
END.
2. SIMAN's EXPERIMENT FILE: Free Path Transporter

BEGIN;

PROJECT:

ATTRIBUTES: Timeln: OpTime;


RESOURCES: Worker: Painter: NewPainter: Packer;


TRANSPORTERS: Truck, 2. DISTANCE. 150;

Assembling OpTime=UNIF(21.25,2) & Paint OpTime=NORM(22.4,2) & Pack OpTime=TRIA(20,22.26,2) & ExitS.
2. Part2.

TALLIES: FlowTime;

DSTATS: NQ (Assembling). Assembling WIP;
NR (Worker)*100. Worker Utilization;
NQ (Paint). Paint WIP;
NR (Paint)*100. Painter Utilization;
NQ (Pack). Pack WIP;
NR (Packer)*100. Packer Utilization;
NQ (NewPaint). NewPaint WIP;
NR (NewPainter)*100. NewPainter Utilization;
NQ(TruckQ). Waiting For Trucks;
NT (Truck). Busy Trucks;

REPLICATE.1,0,700000...20000;

END.
3. SIMAN's MODEL FILE : Guided Vehicle

BEGIN:
CREATE: EXPO(28.1):
MARK(TimeIn):
ASSIGN: M=Enter:
NS=DISC(0.7, Part1, 1, Part2, 10):
ToStart QUEUE:
REQUEST: AGVQue:
DELAY: AGV(SDS):
TRANSPORT: AGV, SEQ:
STATION: StationSet:
ASSIGN: SetIndex=MemIds(StationSet,M):
DELAY: 2:
BRANCH. 1:
IF.NQ(AGVQue)>0,FreeAGV.ELSE.SendStage:
FreeAGV FREE: AGV:
WorkQue QUEUE:
SEIZE: QueueSet(SetIndex):
DELAY: WorkerSet(SetIndex):
RELEASE: OpTime:
SendStage BRANCH. 2:
ALWAYS.SendBack: ALWAYS.WorkQue:
SendBack MOVE:
STATION: AGV.Staging:NEXT(FreeAGV2):
DELAY: ExitS:
TALLY: FlowTime, INT(TimeIn):
BRANCH. 1:
IF.NQ(agsQue)>0,FreeAGV2:ELSE.SendBack:
FreeAGV2 FREE: AGV:
DISPOSE:
END.
4. SIMAN's EXPERIMENT FILE : Guided Vehicle

BEGIN:

PROJECT:

ATTRIBUTES: Timeln: OpTime: SetIndex:


QUEUES: AssemblingQ: PaintQ: PackQ: NewPaintQ: AGVQue:

RESOURCES: Worker.1: Painter.1: Packer.1: NewPainter.1:


INTERSECTIONS:
1. IntAssembling.1:
2. IntPaint.1:
3. IntPack.1:
4. IntNewPaint.1:
5. IntExitS.1:
6. IntEnter.1:
7. Int7.1:
8. Int8.1:
9. Int9.1:
10. Int10.1:
11. Int11.1:
12. IntStaging.1:

LINKS:
1. Link1. IntAssembling.Int7.8.10:
2. Link2. Int7. IntPaint. 4.10:
3. Link3. Int7. IntNewPaint. 4.10:
5. Link5. IntPaint. Int8. 6.10:
7. Link7. IntPack. IntExitS. 25. 10:
8. Link8. IntExitS. Int9. 22. 10:
10. Link10. Int9. Int10. 11. 10:
11. Link11. Int9. IntStaging. 10. 10:
12. Link12. IntStaging. Int10. 3. 10:
13. Link13. Int10. Int7. 8. 10:
15. Link15. Int11. IntEnter. 23. 10. Spur:
16. Link16. Int11. IntAssembling. 9. 10:

NETWORKS:
1. AgvPath. 1-16:

TRANSPORTERS: 1. AGV. 3. NETWORK(AgvPath) -s. 100. LINK (11)-A-ZONE(1):
   Assembling, OpTime=UNIF(21,25.2) & Paint, OpTime=NORM(22,4.2) &
   Pack, OpTime=TRIA(20,22,26.2) & ExitS.
   2. Part2.
   Assembling, OpTime=UNIF(21,25.2) & NewPaint, OpTime=NORM(49,7.2) &
   Pack, OpTime=TRIA(21,23,26.2) & ExitS.

TALLIES: FlowTime:

DSTATS: NQ (Assembling), Assembling WIP.
NR (Worker)*100, Worker Utilization:
NQ (Paint), Paint WIP:
NR (Painter)*100, Painter Utilization:
NQ (Pack), Pack WIP:
NR (Packer)*100, Packer Utilization:
NQ (NewPaint), NewPaint WIP:
NR (NewPainter)*100, NewPainter Utilization:
NQ(AgvQue), Agv Queue:
NT (AGV)*100/3, Busy Agvs/(no. of busy units)

REPLICATE.1,0,7000000...200000;

END:
5. SIMAN's MODEL FILE: Accumulating Conveyor

BEGIN:

CREATE: EXPQ(2, 2):
MARK(Timeln):
ASSIGN:
M=ENTER:
NS=DISC(0.7, Part1, 1.0, Part2, 10):
Size=NS+1:
ASize=NS:

Conv QUEUE. MainQ:
ACCESS. Main.Size:
CONVEY. Main.SEQ:
STATION. Assembling:
QUEUE. AssemblingQ:
SEIZE. Worker:
DELAY. OpTime:
RELEASE. Worker:
BRANCH. 1:IF NS==1, Main2 ELSE OLOOPQ:
Main2 CONVEY. Main.SEQ:
OLOOPQ EXIT. Main.Size:
QUEUE. LoopQ:
ACCESS. Loop.Size:
CONVEY. Loop.SEQ:
STATION. Paint:
QUEUE. PaintQ:
SEIZE. Painter:
DELAY. OpTime:
RELEASE. Painter:
CONVEY. Main.SEQ:
STATION. Newpaint:
QUEUE. NewpaintQ:
SEIZE. NewPainter:
DELAY. OpTime:
RELEASE. NewPainter:
CONVEY. Loop.SEQ:
STATION. Pack:
BRANCH. 1:IF NS==1, Main1 ELSE OLOOP2:
OLOOP2 QUEUE. MainQ2:
ACCESS. Main.Size:
EXIT. Loop.Size:
Main1 QUEUE. Packer:
SEIZE. Packer:
DELAY. OpTime:
RELEASE. Packer:
CONVEY. Main.SEQ:
STATION. ExitS:
EXIT. Main.Size:
TALLY. Flowtime. INT(Timeln):
DISPOSE:
CREATE:
FailMain DELAY: EXPO(55.3);
STOP:
DELAY: NORM(5.14);
START: Main: NEXT(FailMain);
CREATE:
FailLoop DELAY: EXPO(55.3);
STOP:
DELAY: NORM(5.14);
START: Loop: NEXT(FailLoop);
END:
6. SIMAN's EXPERIMENT FILE: Accumulating Conveyor

BEGIN:

PROJECT:


QUEUES: AssemblingQ. PaintQ. NewpaintQ. PackQ. MainQ2. MainQ. LoopQ;


CONVEYORS: 1. Main. MainSet. 100. 1. a. 3. A. ASize;
2. Loop. LoopSet. 100. 1. a. 3. A. ASize;


OpTime=TRIA(20.22.26.2) &ExitS;

TALLIES: FlowTime;

DSTATS: NQ(AssemblingQ). Assembling WIP:
NR(Worker)*100. Worker Utilization:
NQ(Paint). Paint WIP:
NR(Painter)*100. Painter Utilization:
NQ(NewPaint). Newpaint WIP:
NR(NewPainter)*100. NewPainter Utilization:
NQ(Pack). Pack WIP:
NR(Packer)*100. Packer Utilization:
NEC(Main). Main Conveyor:
NEC(Loop). Loop Conveyor:

REPLICATE .1.0.700000...20000;

END:
**Exercise A: Results for Free Path Transporter model**

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Average</th>
<th>Variation</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Observation</th>
</tr>
</thead>
<tbody>
<tr>
<td>FlowTime</td>
<td>51351.</td>
<td>.51011</td>
<td>2617.9</td>
<td>1.0053E+05</td>
<td>37844</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Average</th>
<th>Variation</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Final Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assembling WIP</td>
<td>.71018</td>
<td>1.4829</td>
<td>.00000</td>
<td>7.0000</td>
<td>2.0000</td>
</tr>
<tr>
<td>Worker Utilization</td>
<td>81.040</td>
<td>49.660</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Paint WIP</td>
<td>.04475</td>
<td>3.4989</td>
<td>.00000</td>
<td>3.0000</td>
<td>.00000</td>
</tr>
<tr>
<td>Painter Utilization</td>
<td>53.272</td>
<td>.94565</td>
<td>.00000</td>
<td>100.00</td>
<td>.00000</td>
</tr>
<tr>
<td>NewPaint WIP</td>
<td>.15453</td>
<td>2.3242</td>
<td>.00000</td>
<td>5.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>NewPainter Utilization</td>
<td>41.468</td>
<td>1.0101</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Pack WIP</td>
<td>.30380</td>
<td>1.9762</td>
<td>.00000</td>
<td>7.0000</td>
<td>.00000</td>
</tr>
<tr>
<td>Packer Utilization</td>
<td>71.061</td>
<td>.47640</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Waiting for Truck</td>
<td>1233.1</td>
<td>.54053</td>
<td>76.00</td>
<td>3455.0</td>
<td>3121.0</td>
</tr>
<tr>
<td>Busy Truck</td>
<td>100.00</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

Execution time: 1.45 minutes

Simulation run complete.
**Exercise A: Results for AGV model**

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Average</th>
<th>Variation</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Observation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow Time</td>
<td>55371.</td>
<td>.54072</td>
<td>2812.8</td>
<td>1.0553E+05</td>
<td>30277</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Average</th>
<th>Variation</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Final Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assembling WIP</td>
<td>.78018</td>
<td>1.4729</td>
<td>.00000</td>
<td>10.000</td>
<td>2.0000</td>
</tr>
<tr>
<td>Worker Utilization</td>
<td>80.040</td>
<td>.49936</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Paint WIP</td>
<td>.08475</td>
<td>3.4989</td>
<td>.00000</td>
<td>3.0000</td>
<td>.00000</td>
</tr>
<tr>
<td>Painter Utilization</td>
<td>52.162</td>
<td>.95765</td>
<td>.00000</td>
<td>100.00</td>
<td>.00000</td>
</tr>
<tr>
<td>New Paint WIP</td>
<td>.15853</td>
<td>2.8842</td>
<td>.00000</td>
<td>5.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>New Painter Utilization</td>
<td>.49.468</td>
<td>1.0106</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Pack WIP</td>
<td>.30380</td>
<td>1.9762</td>
<td>.00000</td>
<td>7.0000</td>
<td>.00000</td>
</tr>
<tr>
<td>Packer Utilization</td>
<td>75.061</td>
<td>.57640</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>AGV Queue</td>
<td>1973.1</td>
<td>.54093</td>
<td>80.000</td>
<td>3805.0</td>
<td>3801.0</td>
</tr>
<tr>
<td>Busy AGV</td>
<td>100.00</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

Execution time: 15.02 minutes

Simulation run complete.
### Exercise A: Results for Conveyor model

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Average</th>
<th>Variation</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Observation</th>
</tr>
</thead>
<tbody>
<tr>
<td>FlowTime</td>
<td>252.24</td>
<td>52668</td>
<td>95.910</td>
<td>1536.3</td>
<td>33988</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Average</th>
<th>Variation</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Final Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assembling WIP</td>
<td>.00000</td>
<td>---------</td>
<td>.00000</td>
<td>.00000</td>
<td>.00000</td>
</tr>
<tr>
<td>Worker Utilization</td>
<td>82.277</td>
<td>.46410</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Paint WIP</td>
<td>.00000</td>
<td>---------</td>
<td>.00000</td>
<td>.00000</td>
<td>.00000</td>
</tr>
<tr>
<td>Painter Utilization</td>
<td>55.158</td>
<td>.90164</td>
<td>.00000</td>
<td>100.00</td>
<td>.00000</td>
</tr>
<tr>
<td>NewPaint WIP</td>
<td>.00000</td>
<td>---------</td>
<td>.00000</td>
<td>.00000</td>
<td>.00000</td>
</tr>
<tr>
<td>NewPainter Utilization</td>
<td>52.667</td>
<td>.94801</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Pack WIP</td>
<td>.40975</td>
<td>1.2002</td>
<td>.00000</td>
<td>1.0000</td>
<td>.00000</td>
</tr>
<tr>
<td>Packer Utilization</td>
<td>81.844</td>
<td>.47099</td>
<td>.00000</td>
<td>100.00</td>
<td>100.00</td>
</tr>
<tr>
<td>Main Conveyor</td>
<td>1.5365</td>
<td>.80372</td>
<td>.00000</td>
<td>14.000</td>
<td>2.0000</td>
</tr>
<tr>
<td>Loop Conveyor</td>
<td>.16184</td>
<td>2.4421</td>
<td>.00000</td>
<td>5.0000</td>
<td>.00000</td>
</tr>
</tbody>
</table>

Execution time: 9.47 minutes

Simulation run complete.
APPENDIX D. FLOW CHART FOR QUICK SIMULATION MODELING SYSTEM

give a file name to SIMAN model frame, ext. must be mod. 1

give a file name to SIMAN Experiment frame, ext. must be exp. 2

First, let QSMS select the most appropriate Assembly system for you 3

Choose Assembly system 4

Input details about workstations 5

Input details about Material Handling system 6

Input details about flow of workpiece 7

Input details about simulation data 8

Do you want to make another SIMAN model? 9

Yes

No

Exit
Is there any variation among the products which are going to be assembled by your assembly system?

Y → the products have been grouped into family?
  → all are mechanical operation?
    → MULTI-STAGE CELL IS SUGGESTED
    → Throughput will be adjusted by parallel servers?
      → FLEXIBLE FLOW LINE IS SUGGESTED
      → MULTI-STATION CELL IS SUGGESTED
        → SINGLE-STATION CELL IS SUGGESTED
          → you have to select the assembly system again
  → MANUAL ASSEMBLY SYSTEM IS SUGGESTED
  → Human are used?
    → Single robot is used?
      → N
      → N

N → A

5
All are mechanical operations?

Y
Operations are performed on both sides of products?

Y
LINEAR INTERMITTENT LINE IS SUGGESTED

N
MANUAL ASSEMBLY LINE IS SUGGESTED

5
Extremely high throughput is required?

Y
CONTINUOUS MOTION ASSEMBLY LINE IS SUGGESTED

N
ROTARY INTERMITTENT LINE IS SUGGESTED

5

5
APPENDIX E: 1. ACD of workpieces in Continuous Motion Assembly Line

- **Arrive**
  - *wait 1*
  - *move and processing*
    - **wait 2**
    - *leave*

  *processing while workpieces are moving*
2. ACD of workpiece in Intermittent Assembly Line

- **Arrive**
- **wait 1**
- **move**
- **wait 2**
- **processing**
- **wait 3**
- **leave**

Indexing in conveyor
3. ACD of workpiece in Manual Assembly Line

Arrive

wait 1

move

wait 2

processing

wait 3

leave

outside world

idle

conveyor or free path transporter
4. ACD of workpiece in Flexible Assembly Flow Line

Arrive

wait 1

loading

wait 2

move

wait 3

unloading

wait 4

processing server >1

wait 5

leave

outside world

idle

conveyor or free path transporter or guided vehicles
5. ACD of workpiece in Flexible Assembly Cell

outside world

Arrive

wait 1

move

idle

wait 2

processing server = 1

wait 3

leave

county or free path transporter or guided vehicle
REFERENCE


