"COMPUTER MODELLING OF A MANUFACTURING PROCESS"

BY: JOHN DAVID BEVAN

A MASTER'S THESIS

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE AWARD OF MASTER OF PHILOSOPHY

OF THE LOUGHBOROUGH UNIVERSITY OF TECHNOLOGY

SEPTEMBER 1990

DIRECTOR OF RESEARCH: PROFESSOR J.R. HEWITT
DEPTARTMENT OF MECHANICAL ENGINEERING

SUPERVISORS : MR. I.C. WRIGHT
ENGINEERING DESIGN INSTITUTE

MR. R. HARRISON
DEPARTMENT OF MANUFACTURING

ABSTRACT for "Computer Modelling Of A Manufacturing Process."

Teaching Company Scheme: DCE Limited, Leicester / Loughborough University Of Technology.

DCE is based on the north eastern outskirts of Leicester, is part of the BTR group and manufactures Dust Control Equipment (similar in concept to large vacuum cleaners) for industry. The company employs approx. 500 people at Leicester, having subsidiaries worldwide.

The objective of the project, was to construct a component Data Base for the sheet metal work processed by the factory. Leading onto a Computer Simulation Model of the Cut/Punch/Fold operations carried out on this work. The work was to be processed in a "Family" order, similar gauge and shaped work being processed in a set order, to minimise the setting time between jobs. This was achieved, using LOTUS 123 for the Data Base and PCMODEL software by Simcon Ltd. for the computer simulation.

With a change in Production Control management came a complete change in Philosophy, towards that of MRP II. This change was mirrored by the project enabling actual weeks production loadings to be modelled. The comparison of the Family Part Concept/ work processed by gauge(scheduled)/ Kanban processing logic/ actual factory production was undertaken.

Work processed by gauge was found to be the most effective method of production. When the comparison with the methodology used by the factory(6 wks) was made; the simulation(2.5 wks) was found to be 59% quicker to completion of a weeks' work. Upon investigation the following points were found to contribute to the discrepancy in times: a lack of rigidity in processing the work, method of employee payment, factory layout, working practice and work station ergonomics.

KEY WORDS: LOTUS 123, PC MODEL, COMPUTER MODEL, COMPUTER SIMULATION, WORK STUDY, METHOD STUDY, FLOW PROCESS CHART, SHEET METAL.

John Bevan.

20/10/90.

ACKNOWLEDGEMENTS

The author wishes to acknowledge the assistance given by the following individuals:-

- Mr. I. Wright - Dept. of Mechanical Engineering, Loughborough University of Technology.
- Mr. R. Harrison - Dept. of Manufacturing Engineering, Loughborough University of Technology.
- Mr. G. Singh - Dept. of Mechanical Engineering, and Manufacturing Systems Coventry Polytechnic.
- Dr. C. Smith - Technical Director, DCE.
- Mr. W.G. Musson - Work Study Manager, DCE.

and others who, knowingly or unknowingly, have contributed to the completion of this project.

PROJECT REPORT - CONTENTS

LIST OF PLATES

LIST OF TABLES, DIAGRAMS AND FLOW CHARTS

CHAPTER 1

INTRODUCTION

1.0 INTRODUCTION

The following project was undertaken at Dust Control Equipment Limited (DCE) based at Thurmaston, Leicester, (see plate 1 and Appendix 1, Layout drawing of factory floor) liaising with the Manufacturing and Mechanical Engineering Departments of Loughborough University of Technology, and the Mechanical Engineering and Manufacturing Systems Department of Coventry Polytechnic, through the Teaching Company Associate Scheme (T.C.A.).

DCE use large amounts of sheet steel in the construction of Dust Collectors (see plate 2\2.0). The Work in Progress (WIP) and machinery used to process this steel take up a large part of the factory floor space and is a financial burden.
The projects aim was to use computer modelling to review the sheet steel process to show improvements by:

- Decreasing the production process time.

- Decreasing the amount of WIP.

PLATE 1 - AERIAL VIEW OF DCE'S SITE, LEICESTER

1.1 THE TEACHING COMPANY ASSOCIATE SCHEME

The T.C.A. scheme is funded by the Department of Trade and Industry (D.T.I.) and the Science Engineering Research Council (S.E.R.C.). A T.C.A. is employed on a two year contract, and is paid jointly by ~~the~~:

- The Academic Institute (Loughborough University of Technology)
- The Company (DCE Ltd.)
- The T.C.A. Scheme (The S.E.R.C. and D.T.I.)

The T.C.A. scheme being effectively used as a training scheme for graduate level people, enabling enrolment on courses, seminars and such, so that they can move easily into Management at the end of the scheme. At the end of the scheme the Associate is normally taken on by the Company she\he was working for whilst on the T.C.A. Scheme. The aims of the Scheme are to:

- Enable new technology to be incorporated successfully into industry and financially justified.

- Bring an industrial awareness to the Academic Institution and vice versa.

- Develop close links between Academic Institute and local industry.

- Complete a specified project using the T.C.A. to coordinate between a company and an Academic Institute.

- Utilise the Academic Institute research resources to aid the company.

- To provide a useful cornerstone for the T.C.A.'s future career.

- To enrich both academic and industries knowledge and provide useful gains for both.

1.2 DCE LIMITED (DUST CONTROL EQUIPMENT)

The following section is a precis of DCEs' Company history to date, highlighting changes in Company name, product lines and factories. After which is given a Company overview

- On the 22nd February 1919 the Company started life as Kelly Manufacturing Company Limited - specialising in razors and razor blades, in Leicester.

- In 1920 the Company name changed to Dallow Lambert and Company Limited, and changed direction to become fan engineers and sheet metal workers.

- By January 1922 the Company was advertising dust collection plant.

- In 1924 the Company moved to Spalding Street, Leicester.

- By 1938 the company was making the first 'DL' unit dust collectors, following the 1937 Factories Act.

- In 1940 the buildings at Spalding Street were extended.

- By 1944 the T21 Unit, later named Drytex, had been introduced.

- By 1946 the Company was concentrating its activities exclusively on dust control, and introduced the Drymat unit.

- By 1950 a larger Drytex machine was added to the range, the T25.

- In 1951 the Company opened the No.2 Works at Barkby Road, Leicester at the same time introducing the Dustmaster.

- In 1952 Dustmaster units was formed as a subsidiary.

- In 1955 the Head Office and No. 1 Works moved to Thurmaston, Leicester (25,000 sq. ft. factory area).

- In 1959 the name Dustmaster was superceeded by Unimaster, UM series.

- In 1960 the Company's name was changed to Dallow Lambert Limited.

- In 1962 the Dalamatic filter was introduced and the Unimaster UM replaced by the Unimaster UMA series.

- In 1963 the Company was sold to the Thomas Tilling Group, and the Company's name changed to Dust Control Equipment Limited.

- The Dalamatic DLM 60 series was introduced.

- In 1965 a Research and Development block was opened at the Thurmaston site.

- A German subsidiary was incorporated, DCE Entstaubuhgsanlagen GmbH, now called DCE Deutschland GmbH.

- In 1968 the No. 1 Works factory area was extended from 25,000 sq. ft. to 50,000 sq. ft.

- In 1970 the Head Office block was extended. The Company bought into a South African Company, now the wholly Group owned DCE Vokes Pty Ltd.

- In 1971 the No. 3 Works were opened at Thetford.

- In 1972 an Australian subsidiary was opened in Melbourne, now integrated into DCE Vokes Pty Limited.

- The Vokes Group was acquired by Thomas Tilling Group.

- In 1973 the DCE Vokes Group was reorganised with DCE joining the Group within the Thomas Tilling Group, and the operation being reorganised as DCE Vokes Group.

- The No. 1 Works factory area was extended from 50,000 sq. ft. to 100,000 sq. ft.

- In 1974 the No. 4 Works were opened at Cotes Park, Alfreton, Derbyshire.

- A French Sales Company commenced trading as DCE Vokes S.A., now called DCE S.A.

- In 1976 and American Sales Company was incorporated as DCE Vokes Inc., now called DCE Inc.

- In 1977 the Dutch sales operation was integrated into DCE Vokes B.V., now called DCE Benelux B.V.

- In 1978 the Scandinavian Sales Company was incorporated as DCE Vokes A\S, now called DCE Scandinavia A\S.

- In 1979, the Japanese Sales Company was incorporated as DCE Vokes KK, now called Huyck DCE KK.

- In 1982 the No. 4 Works at Cotes Park, Alfreton, Derbyshire, were closed due to the transfer of work to Vokes Limited.

- In 1983 the Thomas Tilling Group were acquired by BTR, British Tyre and Rubber.

- In 1985, the No. 3 Works at Thetford, Norfolk, was closed as it was no longer cost effective.

- In 1986 and Indian Licensee, ACCO, Calcutta, was appointed.

- In 1987 the Spanish Sales Company was incorporated as DCE Iberica S.A.

- The No. 2 Works at Barkby Road, Leicester was closed as it was no longer cost effective.

- DCE Filters Limited were established in New Zealand.

- In 1988 an new range of filters, the Sintamatic, was launched.

- The Company was re-named DCE Group Limited. The United Kingdom operations being conducted as DCE Limited.

- In 1989 the Company was awarded the Prix de Promotion et de Prestige for on going diligence in the defence of the environment.

- A 9000 sq. ft. warehouse and 2000 sq. ft. extension were added to the factory area at Thurmaston.

DCE COMPANY OVERVIEW

DCE manufacture dust extraction/filtration equipment (see plate 2/2.0) used in solving industrial dust problems worldwide. To this end DCE maintains a unique dust library of over 10,000 different samples, which have been catalogued according to their physical and chemical properties.

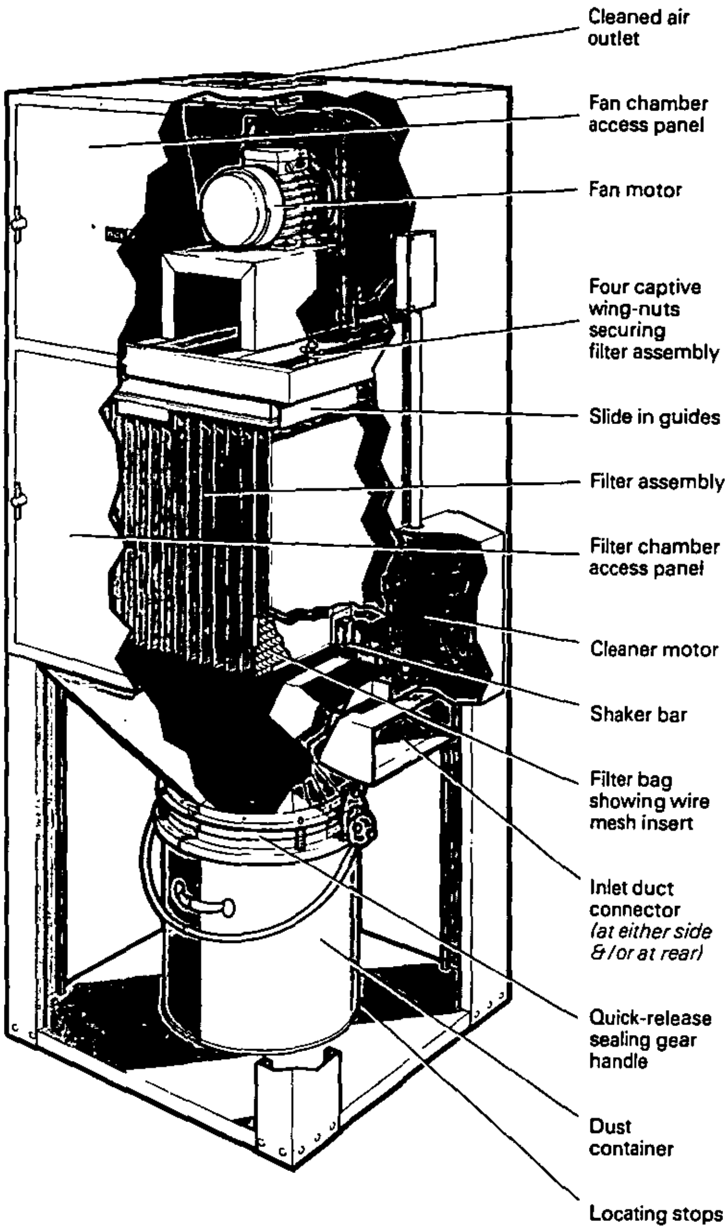


PLATE 2 - A CUTAWAY VIEW OF A DCE UNIMASTER

DCE is part of the BTR (British Tyre and Rubber) Group of Companies, which include the likes of Dunlop, Pretty Polly, and Beaufort. DCE Currently employs between 600\650 people worldwide and 500 people in the United Kingdom. Some of DCE's competitors in the U.K. are:

- Midac
- Dustraction
- Venduct   to name but three.

They are all smaller in size than DCE, and as such do not represent a serious threat currently in such a large expanding market place. Abroad there is more competition especially from France, Germany and the USA.

The filters produced fall into 4 different ranges. With an ever expanding range of options of filter media case material etc. being available. The main breakdown being:

Unimaster - 40% of production
Dalamatic - 40% of production
Sintamatic - 10% of production
Small Units - 10% of production

PLATE - 2.0 DCE UNIMASTER IN ACTION

1.3 DCE'S CURRENT MANUFACTURING METHODS

Currently, DCE's manufacturing procedure is to produce primarily on a four weekly basis, using batch production techniques. The size of a batch of components to be produced is based on past usage calculations. The components being used to replenish a stock holding prior to final assembly operations. The route through the factory for the vast majority of sheet steel is:

Bought in Sheet Steel

Shear (Salvagnini)

Punch (Behrens)

Fold (Hammerle Brake Press)

Weld (Sub-Assembly)

Paint

Store

Assembly

Test

Pack

Despatch

(For technical details on the relevant machines, see section 1.7, Current Operations). Expanding on the operation stages:

- The Shear with integral steel store is programmed to optimise its cutting, to produce a minimum of scrap. In doing this the quantities of steel become large, together with the time taken, up to several days. The parts are produced randomly so implying a long wait for a complete batch of components.

- The components are sorted, put onto pallets with documentation, and put into a store awaiting punch operations.

- The components are then taken from the store, have holes punched and lasered into them, and placed into another store awaiting folding.

- The components are then taken from the store, folded, and placed into a store awaiting welding.

- The components are then taken from the store, welded into sub-assemblies, and placed into another store awaiting painting.

- The components are then taken from the store, painted and placed into a store awaiting welding.

- The components are then taken from the store, assembled into units, some packing placed around them and moved to testing, tested and then moved onto the Despatch area, where they are crated, the necessary paperwork done and the unit despatched.

This situation is a parallel to that described in "Simulation - Taking the risk out of investment". (1)

With the work-in-progress being stored randomly in so many different places with no ridgidity, (see plate 2.1) the section leaders spend most of their time sorting out the work to be done next in their section, not leading their section. Multiplication of jobs due to "loosing" them on the shop floor is common as there are so many places that they can be left, there being no single storage point.

PLATE 2.1 - VIEW ON FOLDING SECTION SHOWING THE AMOUNT AND RANDOM STORAGE OF WIP

As the system is loaded from component requirements further down the shop, without any relationship to actual component requirement, inventory quickly builds up in an unorganised manner, strangling the floor space and capacity, implying delays in starting jobs and any problems with programming or loading of the machines further exasperating the problem. As the manufacturing process is moved through, there is a knock-on effect of delays which become more significant the further down the manufacturing system the component moves, the times being cumulative, leading to jobs being performed randomly in an effort to keep the machines busy. In some of the worst cases setting up machines for the same jobs twice in the same day.

As quoted in "Design of Just-in-Time Manufacturing System". (2) The reason for manufacturing lead times being so long, is that the Company is not supplying to Customer demand, but to manufacturing requirements. Such that if 20 components are required by a Customer, but manufacturing states that 25 are to be made, 25 are made. These figures are the average for DCE. So that looking at 2,500 different components manufactured when 20 off of each are required (50,000), 62,500 items are manufactured, i.e., as 25% surplus to requirement. When this reasoning is put to the whole production range (high and low usage items), over-production is the result constrained by finite resources. This gives a slow moving production line, through a long and complex route on the shop floor, moving from one crisis to the next. To summarise DCE's current production problems, there are:-

- Large amounts of work in progress (WIP) preceding assembly.
- Complex logistics.
- Difficulties in maintaining accurate component control on the shop floor.
- Components being produced in a random order.
- Excessive setting times being required.
- Production of components not actually required.

- Slow production of components urgently required.
- Slow implementation of design changes, due to stock first having to be used or written off.

1.4 DCE'S COMPANY PROFILE

To consider the current condition of the factory it is necessary to review both technical and financial aspects of the Company:

1.4.1 TECHNICAL

DCE is the market leader in its section of the market. It has achieved this by two quantum leaps in its technology. The first was in the layout of dust extraction equipment. Originally when a factory was laid out, and the ducting and extraction equipment was fitted throughout a factory to one collector. DCE's products changed this to individual self contained units attached to a set machine or process. Enabling a factories equipment to be added to or moved around with less expense, less down time, and more efficient dust control.

The second, much more recently, has been the development of the Sintamatic Element. All traditional filter bags have been made of woven material, typically cotton or cotton\polyester. The new element, called the Sintamatic, is a sintered plastic element with a covering of P.T.F.E. (P.T.F.E., or Telfon is the acronym for Polyetrafluoroethylene). This cost effective filter improves performance, and has an infinitively longer life than the traditional filters.

As more and more legislation comes into force with regards to 'clean air' the market becomes larger as more companies are forced\volunteered to invest in dust control equipment. As the market leader in this area, DCEs' sales are ever increasing. With no forseeable down turn in the market trend, DCE is in an enviable position.

1.4.2 FINANCIAL

The following section investigates more closely the financial aspects of DCE. Firstly giving a brief financial overview of the Company and Group, then using a typical unit as an example, breaking it down to show the time involved to product it. Demonstrating what savings in time and money can be achieved, by adopting the "typical week" concept and by investing in new machinery that minimises setting times. Finally, reviewing the ordering\manufacturing\delivery times to demonstrate the financial risk. In the 1988 financial year:

- DCE Group turnover was £33 million. (Consolidated)

- DCE Ltd. Turnover

	- Home Sales	£ 7.594 Million
	- Direct Export Sales	£ 0.524 Million
	- Inter DCE Group Sales	£ 8.347 Million
	Total	£16.465 Million

With 650 people employed worldwide, each person is responsible for £0.05 Million turnover.

With 500 people employed in the UK, each person is responsible for £0.03 Million turnover.

Examining DCE Ltd (UK) performance more closely:

- Work in Progress (W.I.P) - Financial year to April 1989 was £1.72 million (average per month)

- With an inventory to cost of sales of 119 days, for WIP. Using a 360 day year this gives a 4 monthly turnover of stock, i.e. stock changes 3 times per year.

- The inventory as of July 1989 was £800,000 with an inventory to cost of sales of 34 days, 10 times per year.

- Implying an average total stock holding of £2.521 Million (15% of DCE's annual turnover).

To clarify these costs, a typical unit, a UNIMASTER UMA 102 G3, similar to that shown in plate 2 was examined.

Using a batch size of 10 and current manufacturing practices gives 19% of time spent on setting, 81% of time spent on manufacturing.

For a batch of 1 this changes to : 70% of time spent on setting, and 30% of time spent manufacturing.

By using the relationships of Production to delivery times the amount of time available for planning of production can be seen.

1.5 DCE'S PAST OPERATIONS

By reviewing past operations and products, it is hoped to show that DCE having moved into dust filtration equipment, invested in the best technology available, whilst showing that their moves through the years continued this philosophy. Demonstrating where the Company has come from , where it is (section 1.6), and where it intends to go (section 1.7), using the manufacturing equipment available.

Before the Salvagnini Cl Shearing Centre was commissioned, 50% of sheet steel that was delivered was cut to the required blank size, and 50% of the sheet steel was cut by DCE on three guillotines.

Before the Behrens Punch machines and the Hammerle Brake Presses, Redman Tools with "Blow Plates" were used, or Brake Presses. DCE owned eleven Brake Presses and started batch production using them in the late 1950's. The Redman Tools at that time were very expensive, costing several thousand pounds each.

There was no paint line as such, only three spray booths. DCE originally went to a company at Thetford to look at the paint line with the aim of purchasing one. They bought the Company! DCE then proceeded to use Thetford to manufacture the Dalamatic range of units, and use its paint line for the complete range of units with painted work. Each day one lorry would travel to DCE from Thetford, and one would travel to Thetford from DCE.

The Unimaster (UMA) range, approx. 75/week production range, and the ADT range, approx. 50/week production rate, have always been manufactured at Thurmaston. Excepting the UMA meshes (the inserts that maintain the shape of the filter bags) that were made at Thetford. At this time DCE Thurmaston batch produced parts for VOKES AIR, namely a disposable filter for British Nuclear Fuels. Cotes Park manufactured most of the metal work for "Roll filters" and frames for air conditioning filters for VOKES AIR (who only manufactured air cleaners). The ranges of filters no longer manufactured are:

- Dry Tube Filters, these consisted of 50 filter bags placed in tubes.

- Wet Collectors, manufactured at Barkby Road. Mainly used in foundries. They could be very large in size, and used water as the filter media and deposited a sludge of water and dust.

1.6 DCE'S CURRENT OPERATIONS

The factory total shop floor area, including 3 mezzanine floors, basement, warehouse and paintline extension is:

147,600 sq. ft.

Examining the method of working\specification of each of the processing (machines) and material to be investigated:

- Sheet Steel
- Tivox Steel Store
- Salvagnini Cl Shear
- Behrens Punches
- Hammerle Brake Presses

Sheet Steel

The sheet steels used by DCE are:

- Mild Steel (CR4) in 1250 x 2500 mm sheets. In sizes 3.0\2.5\2.0\1.5\1.0\0.9 mm

- Mild Steel (CR4) in 1500 x 3000 mm sheets. In sizes 2.0 mm

- Mild Steel (CR1) in 1250 x 2500 mm sheets. In sizes 3.0\1.5\1.2 mm

- Stainless Steel (SS304 and SS316) 1219 x 2438 mm sheets. In sizes 2.5\2.0\1.5\1.0\0.9mm

Also any 1.5 mm sheet may be ordered as 1.6 mm (16 swg) as this is the industries standard thickness but with 1.5 mm sheets for every 16 sheets purchased one is free!

The steel specification is closely monitored on arrival to ensure that no problems are encountered when loading the Tivox Steel Store or in using the Salvagnini Shear. These are shown below in list 1.7.

LIST 1.7 - STANDARD SHEET STEEL SPECIFICATIONS

1) MATERIAL

Prime CR4\CR1 general purpose oiled material conforming to BS 1449. Surface to be free from scratches, blemishes, rust, inclusions scale or roll impressions.

2) DIMENSIONS

Width tolerance to BS 1449 i.e. - 0 + 5mm.
Decoiled length tolerance - 0 + 4mm.

OUT OF SQUARE SHEET IS NOT ACCEPTABLE.

3) FLATNESS

Flatness is critical. Maximum bow is 3mm over a linear metre i.e. over a 2500 length, maximum bow 7.5mm.

4) PACKING

Packs to be labelled with our reference numbers, quantity, quality, size and gauge.

Bundles should be edge protected with topsheet protected from weather. Only one grade\gauge\size per pallet. Bundle not to exceed 2 tonnes.

5) PALLETS

Bundles for forklift off-load.

Pallets should be substantial enough to allow safe unloading and storage, and allow flat presentation of steel. Pallets should have 4 long members and 4 cross members (of equal dimensions), equidistant to prevent movement and sagging. It is stressed, supporting members must be substantial. Steel on un-safe pallets will NOT be accepted into these works.

As of 21st February 1989

It should be noted that stainless steel (SS304 and SS316) are bought in to a similar specification although in a highly polished un-oiled state.

TIVOX STEEL AUTO-STORE

The sheet steel, still on its pallet is loaded by fork-lift truck into the Tivox Auto-Store. The Tivox unloads and deposits the sheet steel, sheet by sheet on designated shelves. The empty pallet is then replaced with a new pallet with steel to be loaded. This continues until the Tivox Auto-Store is full. (The Tivox is shown on the right in the background of plate 3)

Currently 0.5 weeks requirement of steel is kept in the Tivox with an additional 2.4 weeks worth of additional steel being kept in hand, next to the Tivox (October 1989 figures). The specification of the magazine unit (2 off) is as follows:-

No. of different packs which can be stored	7*
Max. sheet size	3000 x 1500 mm
Min. sheet size	2000 x 1000 mm
Max. sheet thickness	4mm
Max. sheet width	140kg
Max. weight\stocking position	3000kg
Max. pack length	125mm
No. of vacuum circuits	3
Installed power	3.3kw
Length x width x height	4000 x 3900 x 4490 mm

*As there are two units, the number of stocking positions is 14 in total (42 tons of steel).

Added to the units are:

- Roller table with centering station: this adjoins the roller table, which forms an integral part of the magazine unit.

- Wire mesh fencing of the magazine units.

- Double sheet thickness check (weighing system).

- Stock check unit OP393 (one per store).

- Interface with Salvagnini Cl Shear.

- Control by Siemens P.L.C. type S5 - 115U.

The double Tivox HL91-4 automatic magazine unit with automatic feeder and centering station is used to feed shearing and punching machines. In this case a Salvagnini Cl Shear.

Tivox is a Swedish Company, with a U.K. distributor, Lomir of Cinderford, Gloucestershire.

The Tivox machine will supply a sheet from any one of its storage positions to the Cl Shear when required. This means that a sheet is waiting to be fed to the Cl Shear as soon as the preceding sheet has been finished. Production can be interupted for emergency shearing at any time, and any material in the store will be ready to be cut within a few seconds. The machine feeds itself, using the vertical and horizontal motion of the arm as well as feeding sheets out to the Shear.

Installation and commission was completed at the same time as the Salvagnini Shear, in November and December 1988. DCE took possession on 2nd January 1989. The total capital expenditure was in the region of £500,000 for the Tivox store, Salvagnini Shear, and changes to the factory floor and contingencies.

SALVAGNINI C1 SHEAR

The sheet steel having been conveyed by roller from the Tivox auto-store, has the edge facing the Shear raised and deflected by a set of vertically moving suckers. A roller the width of the conveyor is then activated beneath the sheet to give some support. Each of the vertical suckers is independent of the others. Those suckers which would touch the edge or miss the sheet of steel entirely are automatically excluded when the sheet is narrower than 1500 mm. (Plates 3 and 4 show the Salvagnini Shear and sheet steel being manoeuvred on its table).

The edge of the sheet of steel which has been lifted by the suckers is gripped by a pincer, fixed to the front of the Shear, and the pulled towards the shearing table after the suckers have released it. The pincer of the transfer feed has two movements in addition to closing and opening in order to pick up and release the sheets of steel. These are a horizontal movement for pulling the sheet and for returning to the pick-up position, and a vertical one for leapfrogging over the sheet which has just been fed and remains on the table.

The movement of the transfer feeder is numerically controlled. The pincer is not rigidly connected to the carriage which carries it and can be moved with reference to it, along two axies orthogonal against elastic restraints. It pulls the sheet against a reference stop. Just before it reaches the stop, the transfer rate slows, to avoid damage to the edge of the sheet steel.

With the sheet of steel on the machine table, the machine starts cutting, it is of conventional construction with hydraulic drives. The handling of the sheet of steel incorporates a computer controlled manipulator, a rotator, and a return to sender system which allows several pieces of steel which may be of different widths to be stored at the back of the shear before being returned to the front where they are related and sheared again.

Two forms of stacking have been used:-

- Manual, used for small pieces and scraps.

There are three different places where this steel is deposited, at the rear of the machine, on a table, under the machine in a trolley (mainly off cuts), and under the machine in a trolley divided into two for small and then longer pieces.
This is also required for non-magnetic material.

- Automatic, used for longer\heavier pieces.
  There are three stations, with a total of six stacking positions.

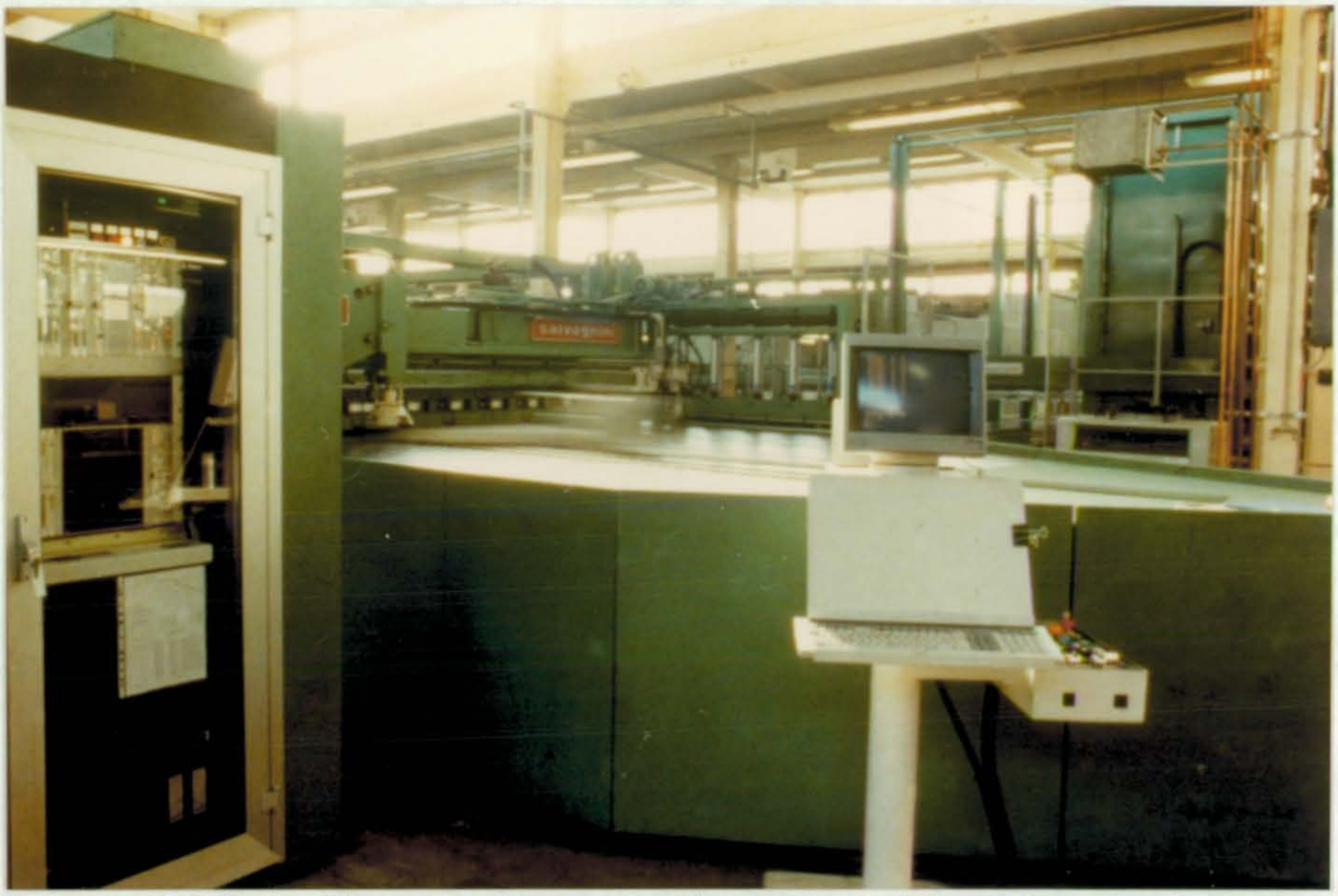


PLATE 3 View showing control panel, table, Tivox, and transfer mechanism of Salvagnini Shear.

PLATE 4 Showing sheet steel being manipulated by C1 Salvagnini Shear.

The Salvagnini Shear was introduced in 1988, to fulfill the need for an automatic shear capable of cutting both large and small pieces in a random pattern from a single sheet or from many sheets. The software was developed to optimise the use of material in the most effective manner. Performance details are as follows:-

Material: All types of sheet metal. That with a delicate surface is protected with an adhesive polythene film.

Thickness: 0.5 to 2.5mm for AISI 304 and AISI 430.
0.5 to 3.0mm for all other types of sheet metal.

Maximum Dimensions of incoming sheet : 3010 mm x 1510 mm

Trimming capacity : 3000 mm x 1500 mm

Maximum weight of pack : 5000 kg

Minimum size for stacking : 200 mm x 100 mm

Maximum size which can be cut and dropped in the base behind the shear blade : 116 mm x 4 times material thickness

The cutting programme aimed for by DCE is:

- 8 hour duration +50% allowance for unloading pallet changes.

- Production of 1550 components.

- Use of 14 tons of steel. Equivalent to 200 sheets.

The Salvagnini Shear is an Italian made machine. It was installed and commissioned at the same time as the Tivox Auto Steel Store, in November and December 1988. DCE took possession on 2nd January 1989. The total capital expenditure was in the region of £500,000 for the Tivox Store, Salvagnini Shear, and all changes to the factory floor and contingencies.

The U.K. distribution company for Salvagnini is Lomir International of Cinderford, Gloucestershire.

PLATE 5 Behrens Punch Laser Machine

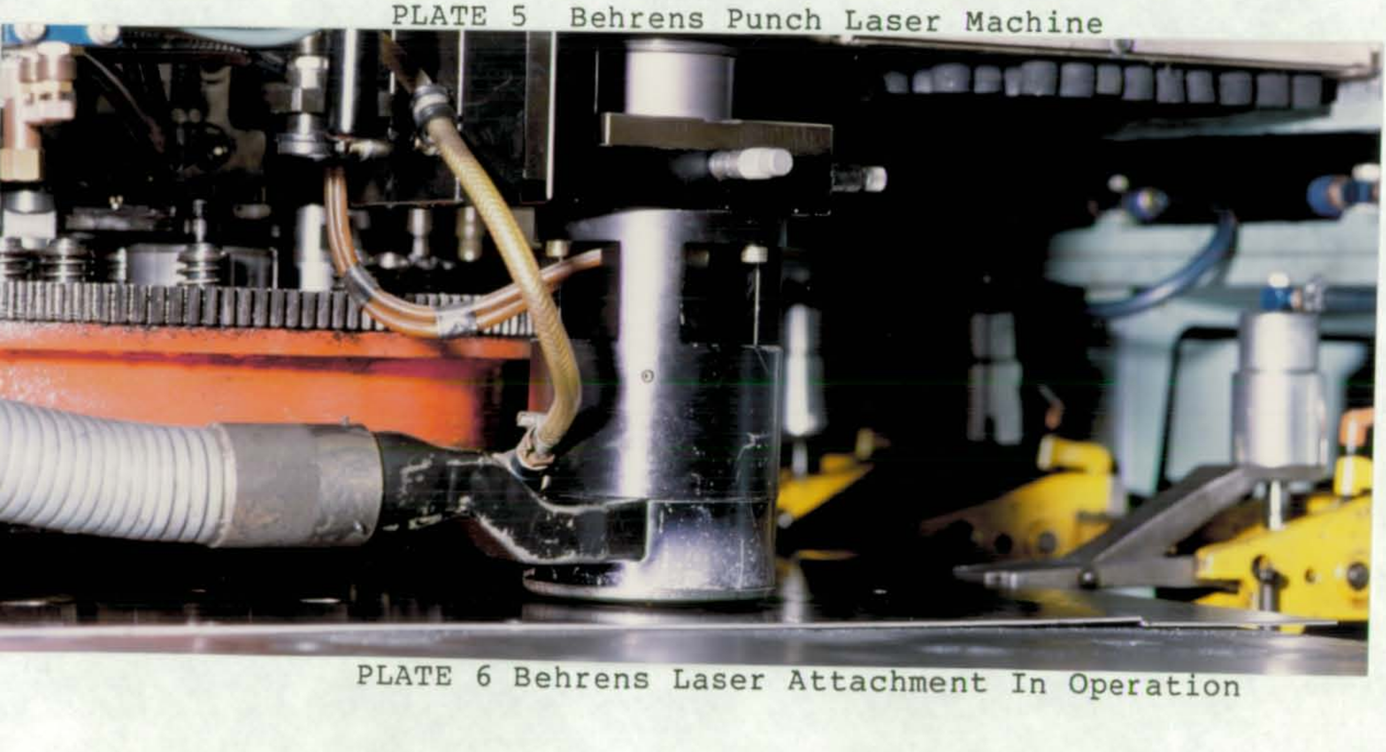


PLATE 6 Behrens Laser Attachment In Operation

BEHRENS PUNCH\LASERS

The sheet steel is conveyed by pallet from the Salvagnini shear to the Behrens Punch\Laser machines, (see plates 5 and 6) where it is manually loaded, punched, and if necessary lasered and conveyed to a number of collection points, where it is manually off loaded.

Punching is a form of shearing. Deformation to shear, failure of the steel, enabling shapes to be punched out. The laser is a highly collimated, monochromatic coherent light source. Used in this case for cutting shapes out of steel. It has a low overall efficiency but produces narrow kerfs, and small heat-effected zones with minimal effects on the rest of the work piece (4). The laser enables complex shapes to be cut out of the sheet steel easily and efficiently when compared to punching out the shapes. Laser is the acronym for Light Amplification by Stimulated Emission of Radiation. The DCE laser gas mix is:

82% Helium
13.5% Nitrogen
4.5% Carbon Dioxide

An oxygen gas cloud shielding the actual laser on the work piece.

For the DCE Behrens Punch\Laser technical data see Appendix. 6.3.4.

The Behrens are West German made machines with a Behrens U.K. subsidiary.

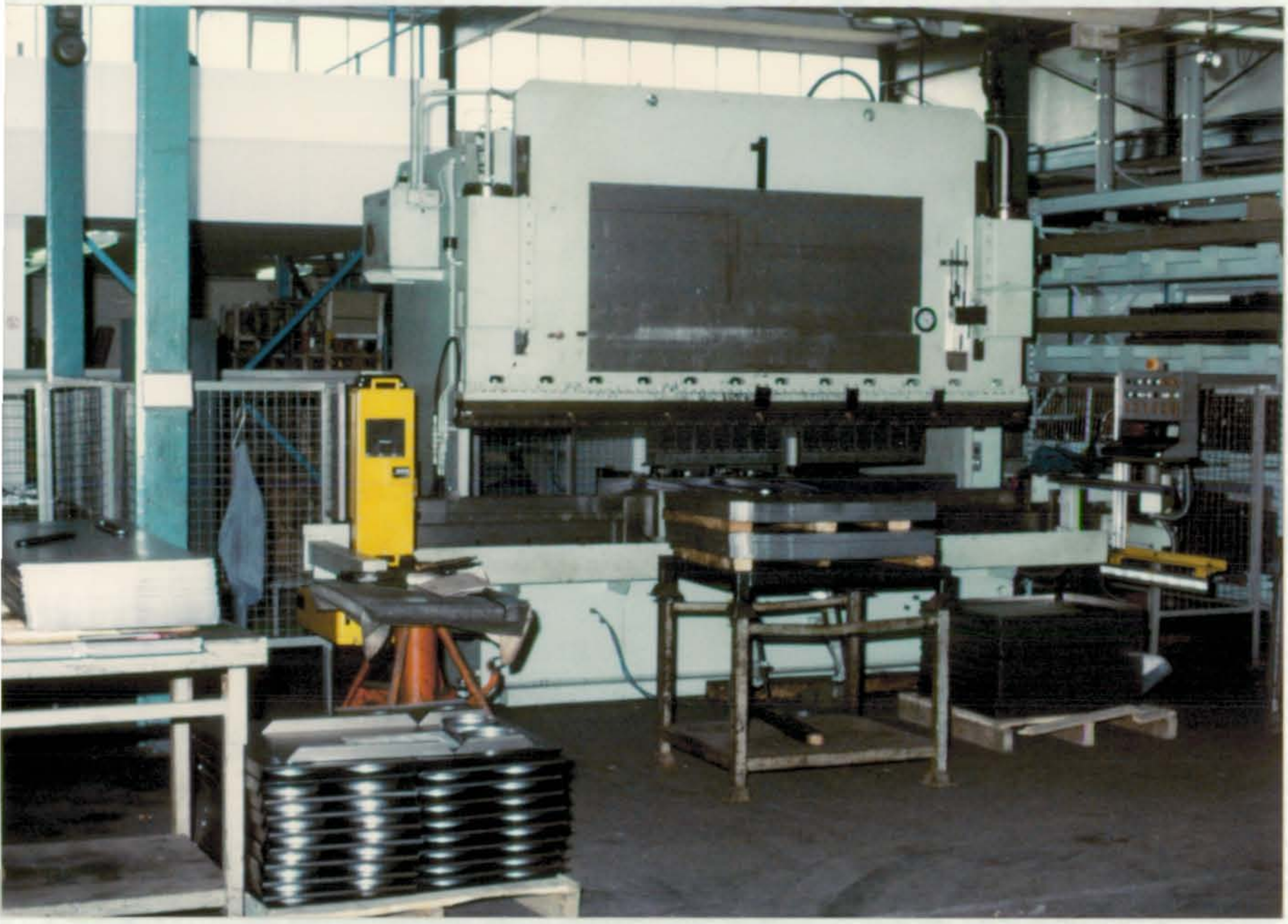


PLATE 7 HAMMERLE N.C. BRAKE PRESS

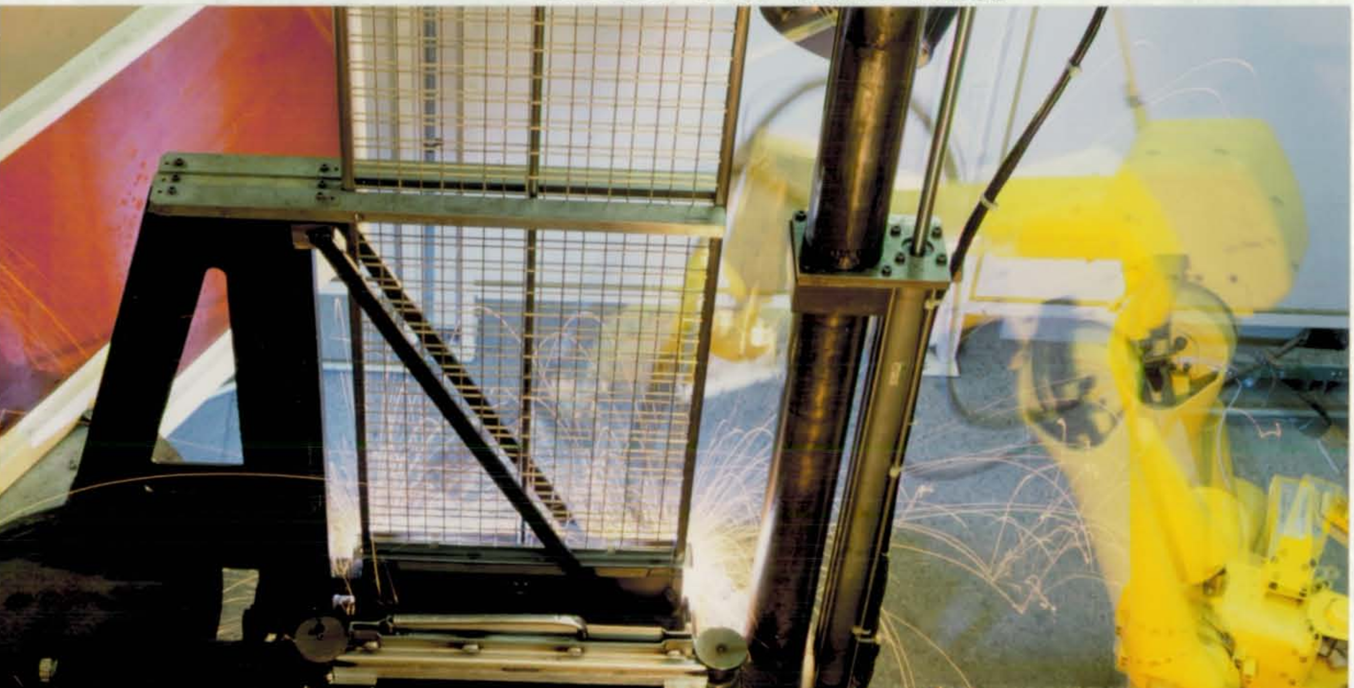


PLATE 8 ROBOTIC WELDING CELL

HAMMERLE PRESS BRAKES

The sheet steel is conveyed by pallet from the Behrens to the Hammerle, where it is manually loaded, folded, and off loaded. (See plate 2.1 and 7) The press brake is a form of press, hydraulically driven, in the case of the Hammerle, to different given capacities. They have long, narrow beds with short, slow adjustable strokes. The work piece is placed between interchangeable dies which are bolted to the bed and the ram. Upon pressing the operating pedal the ram moves down to the work piece. Allowing for a final adjustment of the metal to be made. When more pressure is applied to the pedal the ram comes into contact with the work piece and forces it into the die. The ram then returns to its starting position. Allowing for a new work piece to be inserted or for the work piece to be fed inwards to produce various types of repeated bends, such as corrugations (4). Traditional presses have to change the die to change the fold, but the Hammerle Press (3 point bend) overcomes this by allowing the centre of the die to be adjusted.

The Hammerle are Swiss made machines using Marti Hydromach as their U.K. distributors. For further details see Appendix 6.3.4.

The sheet steel conveyed by pallet from the Hammerle to the welding\fettling sub-assembly area.

1.7 DCE'S FUTURE OPERATIONS

DCE has a planned investment programme instigated by the Manufacturing Operations Review (M.O.R.) Committee. This Committee is chaired by DCE's Technical Director. The Committee's mandate is to review current operations and practices and, where possible, improve on the current situation. If and as necessary investing new capital equipment. The stages followed in an investigation are normally:

Investigation	- Obtaining and analysing data - Reviewing alternative machinery - Technical background - Selecting method of improvement
Costing	- Costing current methods - Obtaining costs for equipment
Justification	- Feasibility of project (Financial and Practical) - Obtaining BTR's consent - Obtaining the monies required for the project.
Placement of orders	
Implementation	- Receipt of equipment - Installation - Training of Operatives - Maintenance - Support - Motivation

DCE is in the fortunate position that as long as a project pays back (PAY BACK) its investment within 4 years, and as such can be justified, the Capital Expenditure (Capex) will be signed by BTR, the Parent Company. This has enabled the M.O.R. Committee to approach its task with confidence and review the operations and practices within the whole Company. Leading to other subsidiaries. e.g. DCE Inc. (USA) setting up their own M.O.R. Committees. Recent and near future investment following the above route includes.

- 2 off Tivox HL91-4 Automatic steel stores
  (Dec. 1988 - £500,000 total)

- 1 off Salvagnini C1 Shearing Centre
  (Dec. 1988)

- 1 off Fanuc Robotic Welding Cell.(Plate 8)
  (Feb. 1989 - £70,000)

- 1 off Rhodes OP Press
  (Jan. 1990 - £120,000)

Other investments made by the Company and involving the M.O.R. Committee have been:

- Paint Line
- Bag making machine
- Relocation of the sheet steel store and handling.
- Sintamatic element manufacture
- BS 5750 approval
- New computer and associated MRPII software for controlling all ordering and stock control.
- Improved method of manufacture of the Unimaster range of units.
- Computer simulation of the factory floor to enable Management to see what is actually happening to parts being made. Hence initiate improvements, via the model to manufacturing time and lower work in progress (WIP), starting with the front end of the shop floor, with the cut\punch\fold operations.
- Investment in new punch\fold equipment to minimize handling and setting times.

The projects\capital expenditure itemized above have been planned and executed during the time scale of this project and will continue after the end of the project with future investment. By continual investment in its manufacturing processes, DCE maintains a cost efficient means of manufacturing its products. By ensuring that the product cost is less and delivery is quicker, thus giving the marketing edge to DCE, together with the technical lead in Dust Control Equipment. As competitors are constantly improving so must DCE. This results in a constantly on-going investment process.

1.8 PROJECT OBJECTIVES

- Construct a component data base using the LOTUS 123 spread sheet computer software package, for the sheet metal work processed by the factory.
- Construct a computer model of the CUT\PUNCH\FOLD process carried out on the sheet metal work, using PC MODEL by Simcon Ltd.
- Create a link between the LOTUS 123 and PC MODEL software packages, so that the data base can be used to provide the data required by the computer model.
- Verify the computer model by comparing predicted results with results obtained from the factory.
- Demonstrate what and where improvements could be made to the sheet metal manufacturing process, using method study techniques.

CHAPTER 2

ACTION TAKEN

2.0 INTRODUCTION

As stated in the Project Objectives (section 1.9) the work within this project was initially to utilise a data base to enable computer modelling of a factory process to take place.

This chapter looks in more detail at why the specific spread sheet (LOTUS 123) and computer modelling (PC MODEL) packages were chosen. Then giving descriptions of the equipment used within the course of the project.

2.1 REQUIREMENT FOR DATA BASE

Due to the Company rapidly expanding its product range, no complete record has ever been kept of the data required to manufacture the sheet metal parts of its filter units. As part of the Manufacturing Operations Review (M.O.R.) Committee tasks, the assimilation of data and ordering of that data was deemed necessary for the following reasons:

- The capacity of the factory was unknown enabling no long term planning to take place. Double shifting was ordered as and when Management thought it required, with output ever increasing it was not known at what stage the factory would not meet requirements.

- There was no linking of data. It was possible to have 3 numbers for a part.
  The Brisch Number - A component code by shape, size etc.
  The Part Number
  The Drawing Number
  There was also a tape number and colour for manufacturing the part on the Behrens presses, together with a second tape number if the part was manufactured on the numerically controlled Hammerle Brake Presses. All the numbers for a component were held in different offices around the Company, i.e.

Brisch Number) Drawing Number)	Drawing Office
Part Number) Drawing Number) Descriptions)	Production Control (Burroughs Computer)
Tape Number) Drawing Number)	Tape Preparation Production Engineering

- It was laborious work finding the data, as each drawing\tape etc. had to be looked up manually. Going through approximately 2,500 components' paperwork. The time scale rapidly becomes excessive. One Manager, when requested to reveal information was heard to reply "how long is a piece of string?" As he had no idea of information being asked of him, and did not have the time to find out!

- It was not known what quantities were required to be made each week to fulfill DCE's production requirement. Answering the classic, how\when\where, for the production of each component.

- DCE stated the requirement to obtain BS 5750 release. Thus all items must be traceable and set manufacturing procedures adopted. Requiring a standardisation of its number system and knowing where things are going\coming from.

Thus the requirement of a file to bring together all these aspects of production data was very clear. The data file contained the following information about a component:

- Part number
- Description
- Drawing number
- Tape number (colour\N.C.\laser)
- Tools used in Behrens punch machines
- Gauge of material (blank size)
- Total number of tools used on the Behrens
- High\low gauge material
- Quantity of the component required in a typical week
- Time per component (allowed) Behrens
- Time total (allowed) Behrens for that component
- Setting Behrens (actual)
- Family Number
- Time per component (allowed) Hammerle
- Time total (allowed) Hammerle for that component
- Setting Hammerle (actual)
- Route through machines
- Range of units that component belongs to

This was achieved on a Lotus 123 spreadsheet of 2,700 lines long, and has been minuted to become a Company Document in August 1989 Minutes of the M.O.R. Committee Meeting. (See Appendix 3.1.4.)

2.2 WHY CHOOSE LOTUS 123 FOR THE DATA BASE

With the inception of the Manufacturing Operations Review (M.O.R.) Committee a number of projects were started to discover where the Company was in terms of its production facility. Two examples being:

- How many different parts were there on the O.P. Presses

- How sheet metal work could be catagorized. Both on the brake presses and the punch\laser machines (Behrens)

This data had to be displayed on some form of spread sheet so that it could easily be updated, make hard copies obtainable, and numerical calculations carried out.

The main frame stock control computer (the Burroughs) did not have all these facilities. So with the large number of personal computers in the Company, it was decided to use a specific piece of spread sheet software. The most widely used spread sheet software in the Company being Lotus 123.

It should be noted that other more efficient data bases are available similar to LOTUS 123 such as dBase III or IV, or alternatively a specific program could have been written to sort the data supplied into the specific order required, and even modify it to the shape necessary for the transfer into the computer modelling package. DCE Management did not wish to follow this latter route.

Initially Lotus 123 release 2.01 was used on the Victor 285 P.C., but when the Compaq 386S was purchased, Lotus 123 release 3.0 was also purchased. The reason for this purchase rather than release 2.01 was that leaves could be added to the spread sheet. Thus making it into a form of 3-D spread sheet similar to a book. Both releases were capable of linking with PC model the computer simulation software.

For technical summaries see

- Lotus 123 - Release 2.01 - Section 2.2.1.
- Lotus 123 - Release 3.0 - Section 2.2.2.

2.2.1 LOTUS 123 RELEASE 2.01

This is a computer based software package produced by:-

Lotus Development Corporation,
55, Cambridge Parkway,
Cambridge. MAO 2142

The package is a powerful analytical software programme combining the three most useful business analysis functions - spreadsheet, graphics and database - into one fully integrated package.

The 1-2-3 spreadsheet provides an electronic environment for working with numbers. The spreadsheet is expansive - 256 columns x 8192 rows, so that large, complex jobs can be handled with ease, with no worry about running out of work space. The spreadsheet can then be manipulated and sorted as required. As 1-2-3 only recalculates those cells that have changed since the last recalculations, a fast response time is achieved.

1-2-3 allows a graph to be created from information contained on the spreadsheet and shows the graph on screen in seconds. The choices are from a line graph, x-y graph, scatter graph, bar chart, and stacked bar chart, enabling a graphic illustration of date, trends of patterns that may not be apparent in row numbers to emerge boldly and clearly.

1-2-3 allows a database of 8191 records with up to 256 fields to be developed. The database can be used to store and update information, sort data by primary and secondary categories in ascending or descending order, select and edit those records needed for more detailed analysis, or perform statistical calculations such as mean, count, standard division and variance. (7)

2.2.2. LOTUS 123 RELEASE 3.0

This is a computer based software package produced by:-

Lotus Development Corporation,
55, Cambridge Parkway,
Cambridge. MAO 2142

The package is an update of Lotus 1-2-3 Release 2.01. Release 3.0 retains the same format as previous releases, but also offers as stated in the Lotus 1-2-3 Release 3.0 reference manual (8):

- New worksheet features, including multiple worksheets in the same file, the ability to see 3 worksheets at the same time, and cell mapping.

- New file features, including multiple files in memory, file protection, and file compatibility with previous releases of 1-2-3.

- New graph features, including additional graph types, a hot-view graph window, and greater flexibility in graph customising.

- New data features, including the ability to read data from external database tables into 1-2-3, searching for data in multiple input ranges, and enhanced sorting.

- New print features, including background printing, graph printing form written 1-2-3, and merging text and graphs for reports.

- New range features, including search and replace, new formatting options, and other ease-of-use and editing enhancements.

- New Macro features, including additional advanced macro commands and macro key names, an unlimited number of macro range names and keystroke re-ordering to simplify building macros.

- New @ functions, additional function keys and pointer - movement keys, new file types and extensions, more flexible use of memory and network support.

2.3 WHY USE COMPUTER SIMULATION?

With a large data base rapidly coming together, the requirement to put it to use came to the fore. With output requirement ever increasing and a large work in progress level, a method of judging the efficiency of the factory was required and then testing 'what - if' experiments, without the constant upheaval of moving the factory floor around with the associated costs and chaos entailed. There were four basic methods of investigation available to achieve the changes required on the factory floor:

- Initiative - the author and colleagues 'gut-feel' for what should be done and how the production process could be improved, reaching a decision on what changes were to be made by discussion and with reference to similar projects. With the large amounts of work and money at stake, this option was not viable.

- Analytical - using logrithms, accurate data collected from the system, using simultaneous equations to predict what will happen to the system. Using system data. This method relies on the data extracted from the system to predict the future. At DCE the system was not stable and so could not accurately predict the future. The data base that was constructed only had the machining and setting times in it. The travelling times for the work varied greatly. The end results are complex in that there are just lists of figures and equations. Nothing graphical that can instantly grip managements attention at a committee meeting.

- Numerical - using data collected from the system again with an inherently protracted time scale, using matracies to predict the outcome of events. Again the production process at DCE had too many variables and processed in a random\haphazard way so that nothing predicted would be meaningful. The Lotus 123 file is a numerical array and as such its results can be shown on graphs. But it does not show the inter-relationships of the machines, the queueing times and the transport times incurred in the real process.

- Modelling - this compliments the numerical investigation by graphically displaying the process to be investigated, demonstrating the inter-relationships between the machines the effects of queueing, and transport times. But the results, as with any other method of investigation, are only as good as the model. Time spent accurately describing detail is time well spent.

DCE have effectively backed themselves into numerical analysis required something more dramatic and true to life if anything was to be gained. As numerically using the machine times everything on the factory floor should have been rosy! It was not. Once the model was installed and the process modelled and results obtained, changes would be made to the model and more results obtained and compared to the first. The model thus develops. The model will never be able to tell what to do as it does not process intelligence. But by carrying out a number of structural experiments an optimum result can be achieved.

This evaluation process (5) is one of the most difficult in re-arranging\planning a production facility. Of the evaluation techniques available a proportion rely on subjective criteria and minimum of quantitative data. The following methods are listed by Angel R. Almodouor (5).

- Listing Advantages and Disadvantages.
  Basic but fast method but there is no useful information to be gleaned from this method, just for or against.

- Ranking
  Options are compared against common criteria. The problem is that not all the criteria may be listed and with too much information available the relative importance of each may be lost.

- Weighted Factory Comparison
  This method is similar to Ranking, but using this method each criteria is given a weighting depending on its desirability. Each alternative is then marked against these criteria, which is tabulated in the form of a matrix, the alternative with the highest score is selected.

The problem with the ranking method applies to this, together with the weighting of each criteria.

- Economic Comparison
  This method compares cost justification\breakdown, and is used when the total cost is the only factor to be considered.

- Material Flow
  This method uses computer algorithms operating on block diagrams describing material flow or relationships. These are useful initially for layouts, but require manual adjustments, but have with operational and iteration problems.

- Simulation

  This method shows what happens graphically on a screen, and allows modifications, but is only as good as the information supplied. Although some simulation methods are purely statistically based and only give numerical results.

DCE decided to use simulation, due to the complex nature of the problem, also allowed the Management Board to view what was proposed before any action was taken. As A.A.B. Pritsker (11) states, there are 4 levels of simulation possible:

- As explanatory devices to define a system or problem.

- As analysis vehicles to determine critical elements, components and issues.

- As design assessors to synthesize and evaluate proposed solutions.

- As predictors to forecast and aid in planning future developments.

The objective of the simulation model was to take the guesswork out of decision making process, and was chosen to achieve this. (10).

To quote Partha Protini Bose - "It (simulation) reduces risks associated with process and plant start-ups; speeds the implementation of new manufacturing systems; helps optimize the efficiency, productivity, and utilisation of current or proposed manufacturing systems; and through the use of models, helps in the evaluation and understanding of manufacturing strategies".

It was the optimisation\efficiency\productivity of DCE's current shop floor layout, and methods of production, compared with their proposed shop floor layout and methods of production that DCE were primarily concerned with, but also hoped to obtain the other advantages mentioned by Partha Protini Bose.

Simulation reduces the problems when changing or implementing new plant etc. It is a form of "risk-management", by using the simulation beforehand options can be investigated and using iteration and optimum loading\plan can be achieved, saving money, time and nervous energy! As demonstrated in two very similar problems to DCE which have already been addressed by (1), (2), Mr. D. Heron and Mr. A. Garside.

The first, (1), was a Company that was a major subcontractor to the aerospace industry. With some two thousand part numbers active on the shop floor, the Company having effectively lost control of its manufacturing process. To remidy this an MRPII package was to be installed. The process involved raw material being input into a machine shop, through the use of simulation up to 75% saving in leadtimes, and work in progress were shown to be possible. When relating this to DCE (at 75%), the W.I.P. would drop from £1.72 million to £0.43 million, and the lead time would drop from the historical 4 weeks to 1 week.

The second, (2), used the PC Model software to achieve the results. The Company concerned was an electrical components manufacturer who wanted to improve performance by reducing excessive lead time. Again a problem that DCE has, although not in the same market. The results of the simulation showed that reductions in W.I.P. and throughput times of between 85 and 94 % could be obtained.

Both examples show that large savings and increases in productivity can be made with the use of simulation. Without the need for major expenditure, just optimising the facilities that the Company already possesses and scheduling the work through the manufacturing process.

2.4 COMPUTER SIMULATION BACKGROUND

With every increasing complexity in life (11), and expense, it is necessary to find a way of proving ideas and solving problems without exposing the process to changes. A form of simulation\modelling of the process is required. Alternative methods of evaluation of a process are listed in section 2.3 (5). Upon deciding on simulation, the necessary desirable features have to be examined for the relevant simulation package to be purchased. There are 12 sections to be considered (12):

- Input Flexibility
  The software design should be able to develop models in a batch mode or interactively - using graphics.

- System
  This must be user friendly. When it is good, rapid progress of the model is achieved with few mistakes.

- Structural Modularity
  It should be possible to develop the model in modules. Some useful modules are initial conditions, requirements, equipment characteristics. This enables one to be changed without the others.

- Modelling Flexibility
  Software using event scheduling orientation possesses the greatest power and flexibility, but the development is a tiresome process.

  Network based systems make this simpler by describing the process by the available modes\blanks. But they represent only certain logic, and cannot represent complex decisions.

  For more complex modelling a lower level capability is required using event scheduling, user written process interaction, orientations or interfacing with Fortran, pascal programming languages or similar.

- Modelling Conciseness
  Requirements of a blank\mode being able to select a path downstream, together with user-written process interaction orientations.

- Macro Capability and Hierarchical Modelling
  Development macros, eg. a machining centre to expedite de-bugging time and the development of hierarchy in the model.

- Material Handling Modules
  Modules that represent AGV's robots, conveyors, enable faster de-bugging and model build times. As a major part of manufacturing is taken up in the transport of materials.

- Standard Statistical Generation
  The requirement of the buffers utilisation of machinery etc. should be automatic or simple to specify.

- Data Analysis
  To be able to analyse the data going into and coming out of a model is required. Together with more specific specialised data analysis displayed on charts, tables etc.

- Animation
  Playback or simulation time animation can be used; both have their advantages and disadvantages, but animation is required to show the movement of work through the model.

- Interactive Model De-bugging
  This allows complete control over the models execution and access to data being collected.

- Micro\Main Frame Compatibility (e.g. MRPII)
  The ability to start on a micro computer and to change onto a main frame is an advantage as micro's are more controllable. Most control systems are run from main frames so that real time modelling can be effected.

It should also be noted that good support from the supplier and costs have also to be considered. The PC Model package by Simcon Ltd., chosen for this project has:

- Discrete programming
- General purpose application
- User written process orientation
- Inter-active De-bugging

Once the simulation package has been obtained; simulation can start in earnest, although if started earlier, ie. brought to solve a specific problem, the problem formulation (11) will have a bearing on what is required, and should exert a healthy influence on the decision.

The development of a model\simulation process should follow the following course:

- Problem Formulation
  What is the problem? A statement of the objective is required.

- Model building
  Simulating the process by mathematical relationships with regard to the problem formulation.

- Data Acquisition
  Identification, specification and collection of data.

- Model Translation
  Preparation of the model for computer processing.

- Verification
  Ensuring that the model runs as it should.

- Validation
  Has the desired accuracy or correspondence between the simulation model and the real system been achieved.

- Strategic and Tactical Planning
  Establishing the experimental conditions for using the model.

- Experimentation
  during the simulation model to obtain experimental results.

- Analysis of Results
  Reviewing the simulation models results, in tables, graphically etc., drawing conclusions from them and making recommendations.

- Implementation and Documentation
  Putting the results of the simulation model into practice and writing up the project.

Whilst moving through the process of computer simulation\modelling, there are a number of problems to be avoided, (13) as listed:

- Pitfall Number 1

  Failure to have a well-defined set of objectives.

  A simulation model can only answer a certain number of questions. These questions need to be set before the modelling process begins, eg. the simulation may show the throughput of a factory, but not be able to show the size of the buffers required.

- Pitfall Number 2:

  Treating a simulation study as a programming exercise.

  A number of companies have decided on a programming language for their simulation model. Appoint a team, and trained them, but have not taken into account the actual manufacturing system to be modelled, the statistics and probability, mathematics and operations research techniques, particularly the behaviour of queueing systems.

- Pitfall Number 3:

  Failure to communicate with Management\Decision makers on a regular basis.

It is most import that Management\Decision makers are involved with the simulation project, from its inception. As their knowledge and approval of the model through its development will raise less queries at its conclusion. If in fact the problem that the Managers set has been addressed.

- Pitfall Number 4:

  Software which makes simulation accessible by "anyone".

  With simulation software that is graphics or menu run, there is no requirement for programming and as such little or no validation, input modelling and output data analysis is done. The results can be misleading, or at worst extremely dangerous, as the model may not reflect the shop floor activity.

- Pitfall Number 5:

  Misuse of animation.

  It is important that decisions are made not on the animation alone, especially over short elapsed time. There is an added problem in that some simulation packages allow changing of conditions in mid-run, hence giving an unreal result.

- Pitfall Number 6:

  Replacing distribution by their means. With jobs having random values, taking the mean (average value) does not take into account all the long jobs arriving together and the associated queueing. Also it does not take into account all the small jobs arriving together. So that by taking the mean the "extremes" of the simulation are erased so that a completely misleading situation can be shown.

- Pitfall Number 7:

  Incorrect choice of input probability distributions.

When the supply of jobs to a machine is not known for the future, it is necessary to base the supply on historical data using a distribution to extrapolate the results, eg. exponential, gamma, Weibull, lognormal, and normal distributions. Ensuring that the best fit distribution is used and that it is within the expected limits of those that know the system.

- Pitfall Number 8:

  Incorrect modelling of machine breakdowns.

  Machines will breakdown and as such have to be modelled. Re-cycling the processing time by the percentage of time that the machine is broken down can give quite erroneous results. The rate of machine breakdown and length of time has to be based on the time that the machine is actually broken down.

- Pitfall Number 9:

  Making only one simulation run for a particular system.

  To base any results on one simulation run are extremely hazardous. As with randomly generated numbers in the system. No two runs are going to be the same. There will be a distribution of results and trends will have to be looked for. Storage areas will have to allow for the maximum numbers of components arriving at one time.

- Pitfall Number 10:

  Failure to warm-up a simulation

  When looking at a simulation model of a manufacturing system it is important to allow the simulation to reach a "steady-state". As no normal manufacturing system starts off with no work in progress. Not to do this will give misleading results.

Using all the above pointers\methods it is to be hoped that the simulation model constructed is a true representation of the system being modelled, and that the results can be used in the most appropriate manner, enabling further iteration on the system and hence greater efficiencies as and when the new system is instigated.

2.5 SIMULATION PACKAGES

This is not a listing of simulation packages that are the best, or most expensive. They are listed not in any particular order. Listed below are the names, addresses and telephone numbers of those companies come across manufacturing simulation packages during the course of this project:

Simcon Technology Ltd - 'PC Model' Version 8.40 XP - 01/01/88
'PC Model/XP+'
'CAD motion'

Simcon Technology Ltd.
Anchorage House,
17, Earnley Road,
Hayling,
Hants.
United Kingdom Tel: 0705 468908

PC Model is a graphical modelling system for production processes. It interfaces with spreadsheets and allows the user to interrupt and modify results at any time. The simulation is animated with times for movement\operations specified to add realism.

Istel Ltd - 'Witness'

Istel Visual Interactive Systems Ltd.,
Highfield House,
Headless Cross Drive,
Redditch,
Worcs. B97 5EQ
United Kingdom Tel: 0527 550 330

Witness is based on menu formated screens to define the model. The model can be made as simple or complex as required, including tool stores etc.

CACI Products Company - 'Symfactory with Animation'

CACI Products Company,
Regent House,
89, Kingsway,
London. WC2B 6RH
United Kingdom Tel: 01 528 7980

Symfactory is based on menu formated screens to define the model. The software checks the programming and highlights errors. Analysis is carried out within the software.

Citroen Industrie U.K. - 'Mast'

Citroen Industrie U.K.,
Automation Division,
Bedford Court,
Bedford Street,
Leamington Spa,
Warwickshire. CV32 5DY Tel: 0926 88201

MAST is designed specifically for FMS. Where loading is programmed the layout of the factory. The results of the simulation give the output performance. Work on MAST within an expert shell is now being undertaken.

2.6 THE SELECTION OF P.C. MODEL

This section falls into 2 parts:

- The selection of PC Model.

- PC Model : model principles.

2.6.1 THE SELECTION OF PC MODEL.

Three computer simulation packages were reviewed and the problem discussed with the 2 Higher Education establishments offering those packages named.

Loughborough University of Technology - VS6
Manufacturing Department - Mast

Coventry Polytechnic
Mechanical Engineering Department - P.C. Model

Coventry Polytechnic also had the following computer simulation packages at their disposal:

- Hocus
- Witness

The Mast computer simulation package by Citroen Industrie U.K. was unable to be used due to software bugs and lack of confidence in the package at that time.

With Coventry Polytechnic putting forward one computer package as the ideal solution to the problem, Loughborough University of Technology with one computer package on offer, the two were compared, although at different times and places. This comparison was documented in the form of a report included as Appendix 2.
On the basis of this report PC Model was purchased. The package being able to interface helped with Lotus 123 (the database) and was inexpensive. Both packages being obtainable through Teaching (educational) concessions, proving to be a lot less expensive than if purchased as a commercial tool by a Company in business. Another consideration was that Coventry Polytechnic had considerable experience with PC Model having used it for other consultancies and as a teaching aid, and were actively involved with the technical support for the product.

2.6.2 PC MODEL: MODEL PRINCIPLES

The PC Model software package allows the viewing of the incremental movement of jobs through the process being modelled. The Logic file controls the jobs movement and the process times at specific positions. Thus representing the actual movement of a job through a process. An example of this logic programming is shown in plate 9. The overall construction of the Logic used in the programs is shown in flow charts 3.2, 4.2, 5.2, and 5.3. The Logic controls how the jobs flow through the picture of the process, the overlay. This is constructed in another part of the software, and is very similar in concept to a computer aided design drawing package.(See Appendix 3.2.1, plate 15)

When large quantities of data are to be input into the model, an array is used. This holds all the data for that model, and again is constructed in a different part of the software. Alternatively as used in this project, data can be imported to the array screen from another software package, in this case LOTUS 123. The data in LOTUS 123 has to be arranged in such way that the PC Model array will accept it. There are two differently types of array screen, one for quantities\numbers and a second for times, see plates 11 and 12.

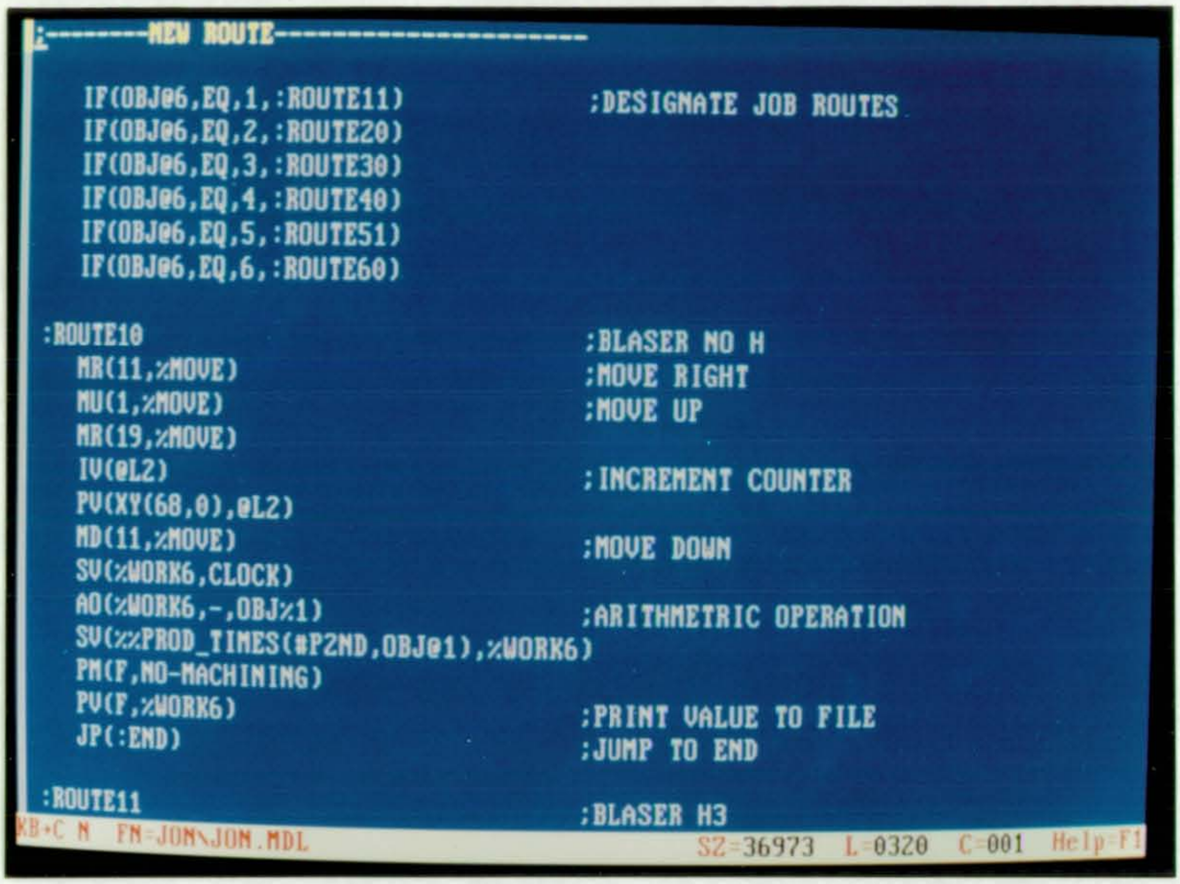
Once the simulation model has been initiated, the utilisation statistics screen can be viewed. This shows the percentage time that a designated position has been occupied by jobs in a specific hour. The number of jobs output by the model in the specified hour are also displayed, see plate 13. The utilisation file can be transfered to another spreadsheet package for further analysis. This was carried out within the project, the file being transferred to LOTUS 123, which enabled graphical representation of the results.

A report file can also be generated showing the order in which the machines are used and at what time, see plate 14. This again can be transferred to an external spreadsheet software package.

In brief:

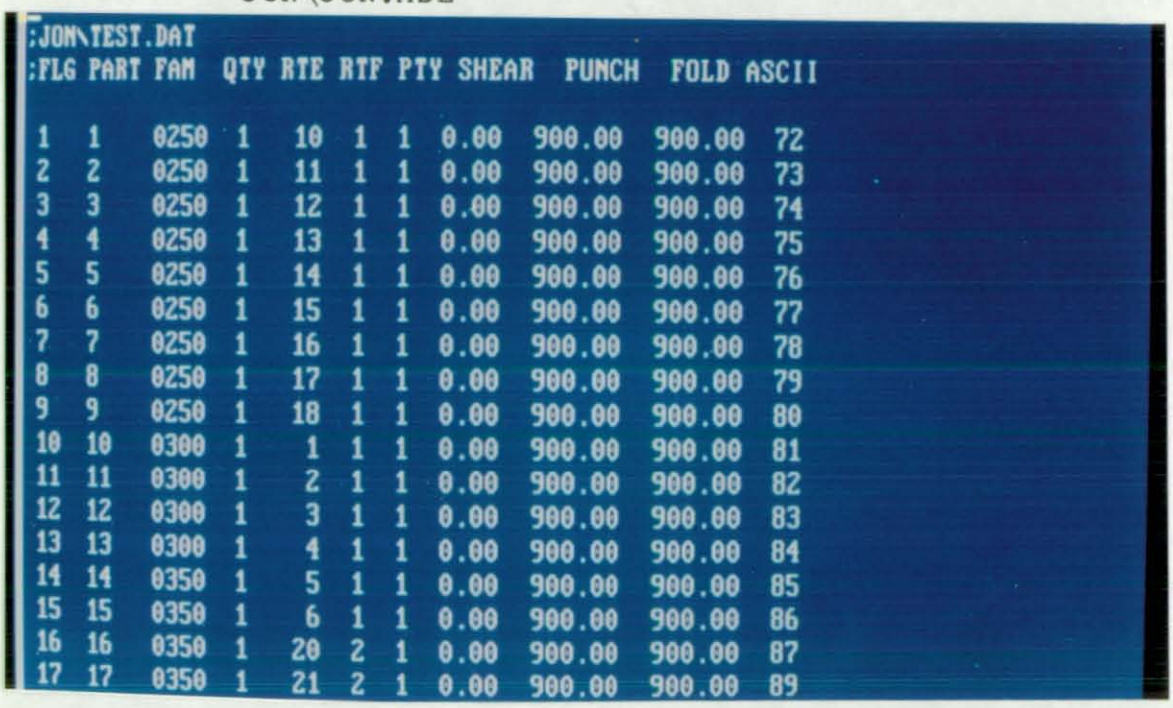
The Logic file name is suffixed .MDL
The Overlay file name is suffixed .OLY
The Array file name is suffixed .DAT
The Report file name is suffixed .RPT
The Utilisation file name is suffixed .STS

More detailed print outs of programs used in the computer modelling are given in Chapters 3, 4, and 5. Showing the development from initial through to the final models.



```
;--------NEW ROUTE---------------------

   IF(OBJ@6,EQ,1,:ROUTE11)            ;DESIGNATE JOB ROUTES
   IF(OBJ@6,EQ,2,:ROUTE20)
   IF(OBJ@6,EQ,3,:ROUTE30)
   IF(OBJ@6,EQ,4,:ROUTE40)
   IF(OBJ@6,EQ,5,:ROUTE51)
   IF(OBJ@6,EQ,6,:ROUTE60)

:ROUTE10                              ;BLASER NO H
   MR(11,%MOVE)                       ;MOVE RIGHT
   MU(1,%MOVE)                        ;MOVE UP
   MR(19,%MOVE)
   IV(@L2)                            ;INCREMENT COUNTER
   PV(XY(68,0),@L2)
   MD(11,%MOVE)                       ;MOVE DOWN
   SV(%WORK6,CLOCK)
   AO(%WORK6,-,OBJ%1)                 ;ARITHMETIC OPERATION
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK6)
   PM(F,NO-MACHINING)
   PV(F,%WORK6)                       ;PRINT VALUE TO FILE
   JP(:END)                           ;JUMP TO END

:ROUTE11                              ;BLASER H3
KB+C N  FN=JON\JON.MDL                SZ=36973  L=0320  C=001  Help=F1
```

PLATE 9 - COMPUTER SCREEN SHOWING PART OF THE PROGRAM JON\JON.MDL



;JON\TEST.DAT

;FLG	PART	FAM	QTY	RTE	RTF	PTY	SHEAR	PUNCH	FOLD	ASCII
1	1	0250	1	10	1	1	0.00	900.00	900.00	72
2	2	0250	1	11	1	1	0.00	900.00	900.00	73
3	3	0250	1	12	1	1	0.00	900.00	900.00	74
4	4	0250	1	13	1	1	0.00	900.00	900.00	75
5	5	0250	1	14	1	1	0.00	900.00	900.00	76
6	6	0250	1	15	1	1	0.00	900.00	900.00	77
7	7	0250	1	16	1	1	0.00	900.00	900.00	78
8	8	0250	1	17	1	1	0.00	900.00	900.00	79
9	9	0250	1	18	1	1	0.00	900.00	900.00	80
10	10	0300	1	1	1	1	0.00	900.00	900.00	81
11	11	0300	1	2	1	1	0.00	900.00	900.00	82
12	12	0300	1	3	1	1	0.00	900.00	900.00	83
13	13	0300	1	4	1	1	0.00	900.00	900.00	84
14	14	0350	1	5	1	1	0.00	900.00	900.00	85
15	15	0350	1	6	1	1	0.00	900.00	900.00	86
16	16	0350	1	20	2	1	0.00	900.00	900.00	87
17	17	0350	1	21	2	1	0.00	900.00	900.00	89

PLATE 10 - COMPUTER SCREEN SHOWING A SECTION OF JON\TEST.DAT ARRAY

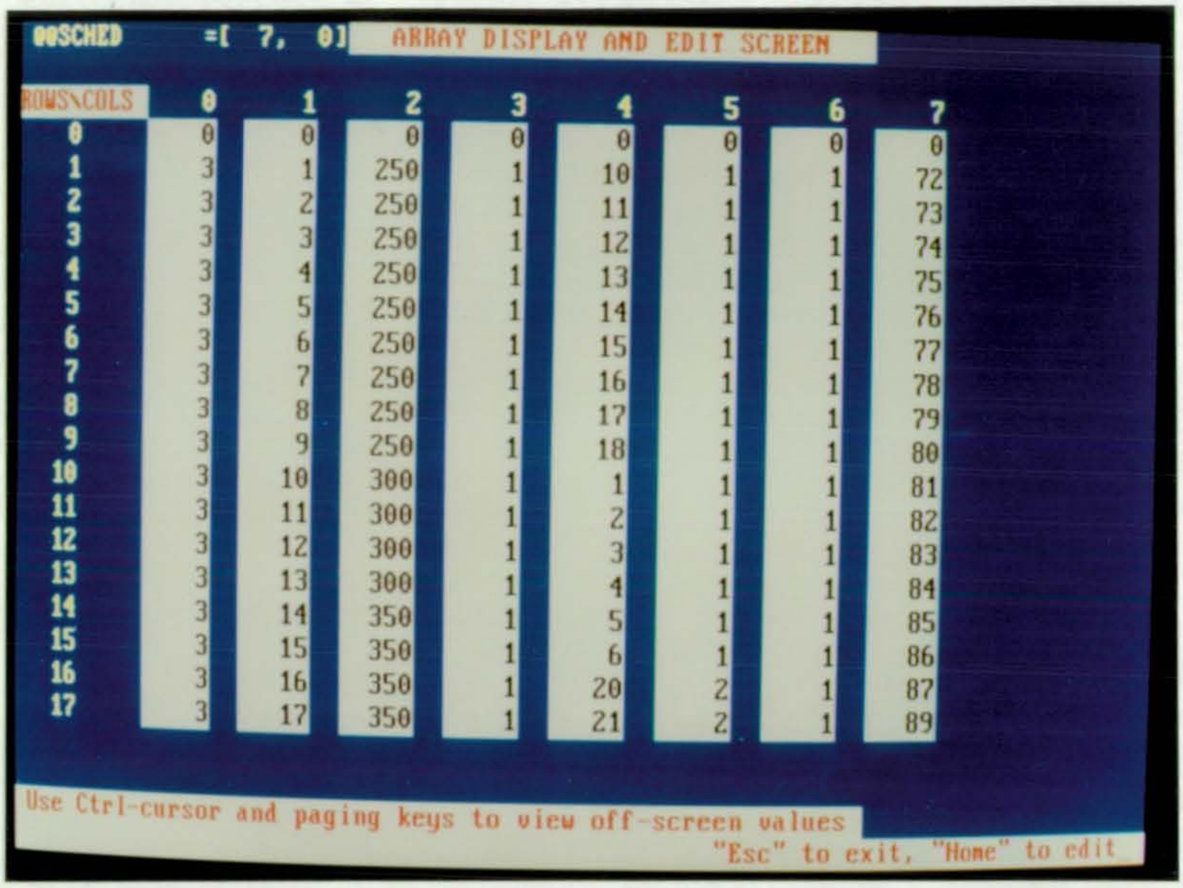


PLATE 11 - COMPUTER SCREEN SHOWING THE DISPLAY AND EDIT SCREEN (ARRAY) FOR QUANTITIES.

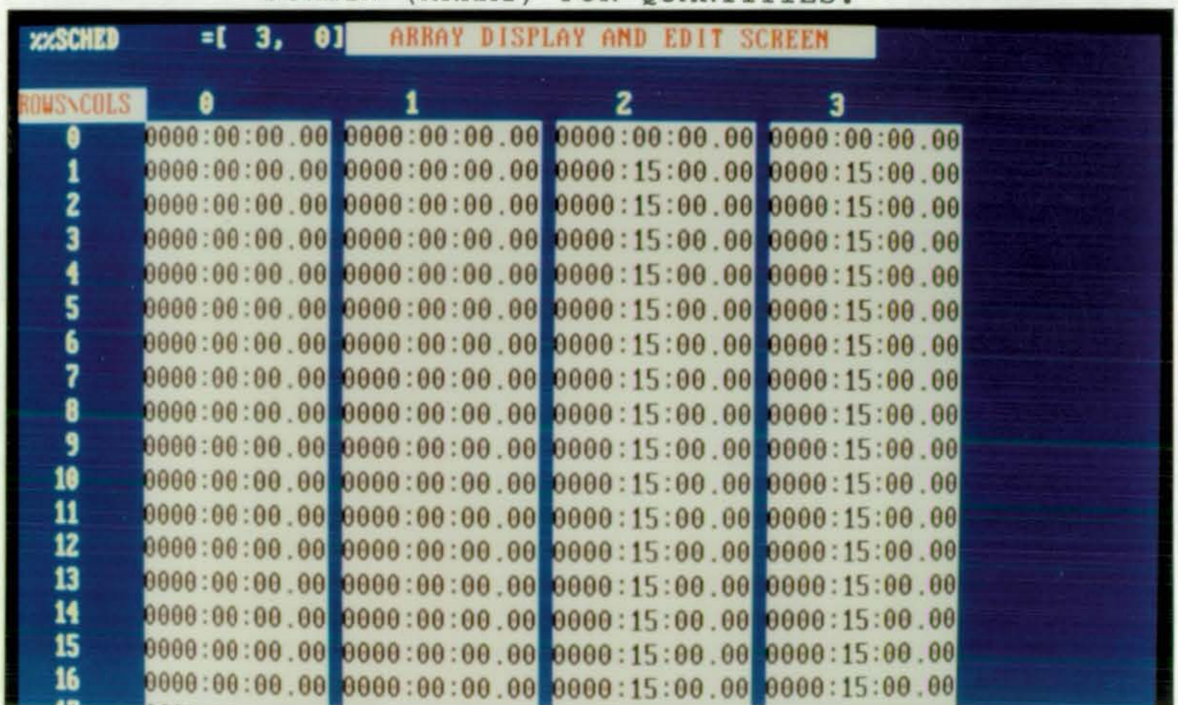


PLATE 12 - COMPUTER SCREEN SHOWING THE DISPLAY AND EDIT SCREEN (ARRAY) FOR TIMES .

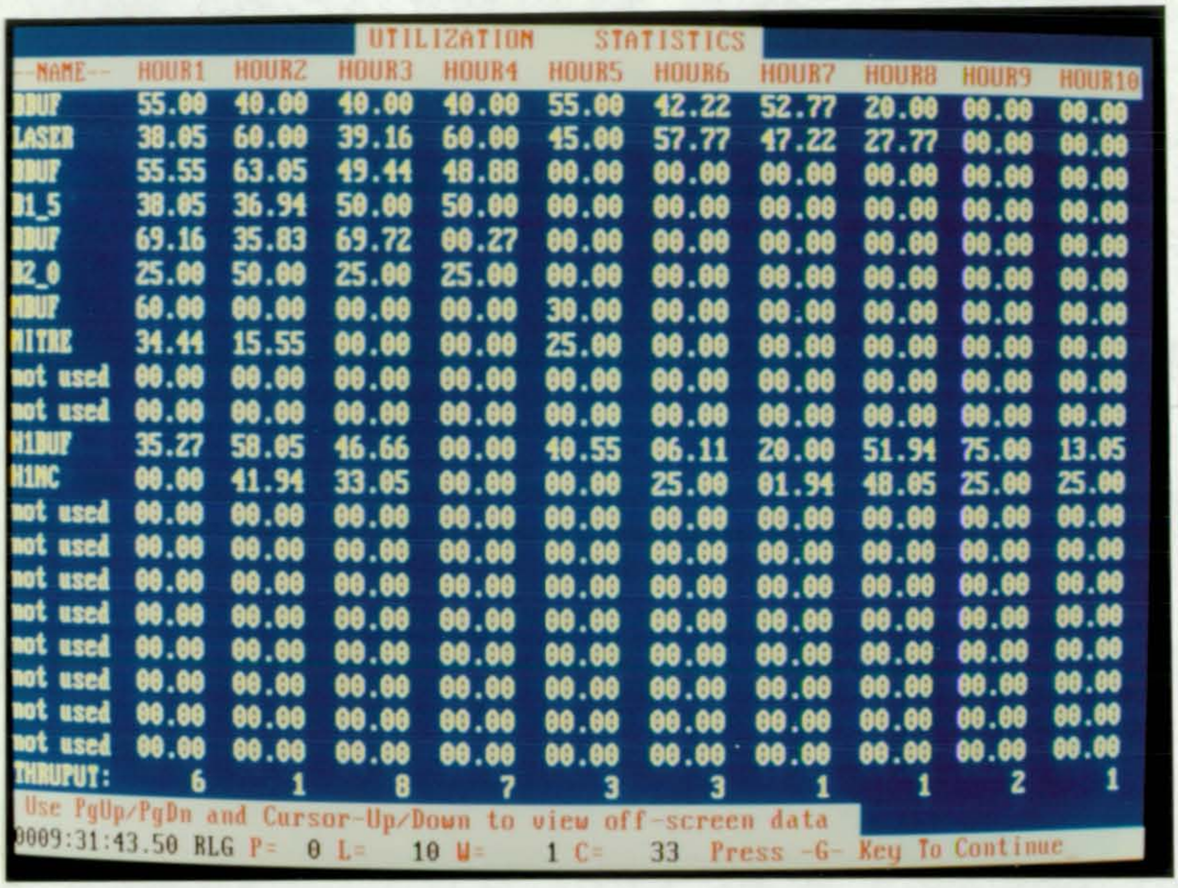


PLATE 13 - COMPUTER SCREEN SHOWING THE UTILISATION STATISTICS

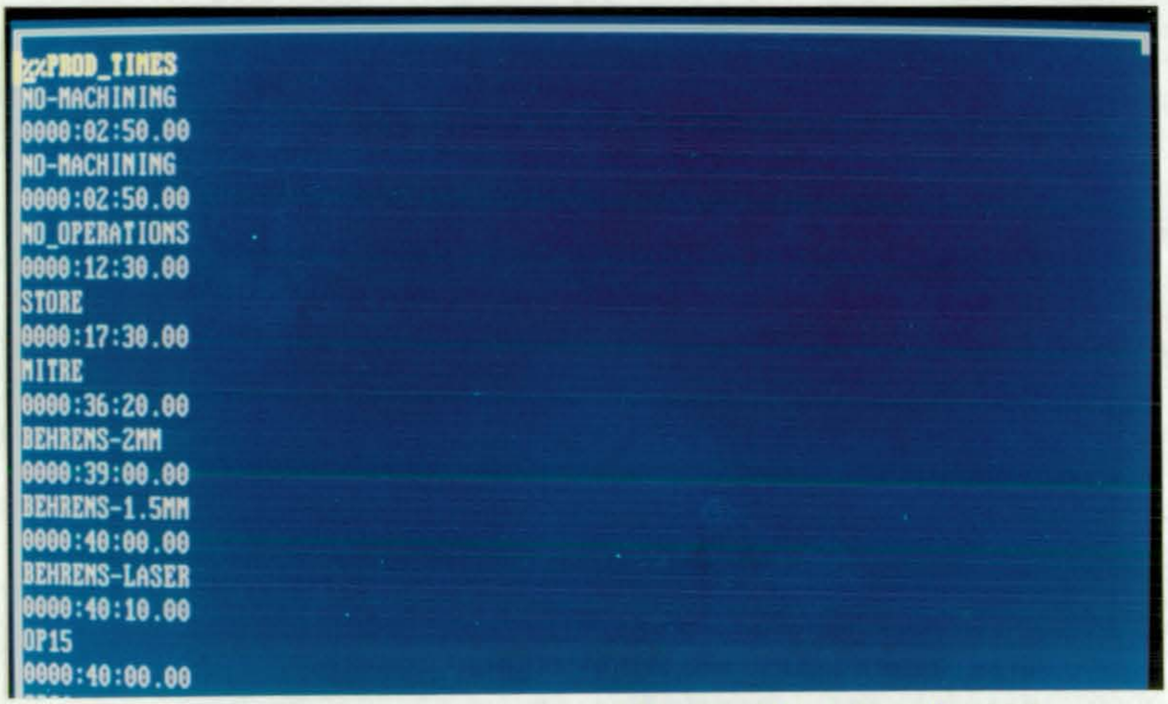


PLATE 14 - COMPUTER SCREEN SHOWING THE REPORT FILE

2.7 COMPUTER - HARDWARE

The equipment used was initially a Victor VPC II 286 Personal Computer and then a Compaq 386S Personal Computer.

2.7.1 VICTOR VPC III 286 PERSONAL COMPUTER TECHNICAL DETAILS

This was a personal computer produced by:

Victor Technologies, Inc.,
380 EL PUEBLO Road,
Scotts Valley,
California.
95066.

and was the property of Loughborough University, although it was permanently based at DCE Ltd., Leicester. The Computer had both hard disc and 5.25" floppy disc drive, with EGA graphics. The random access memory of 640K bytes, was extended in February 1989 to enable the complete Lotus data base to be loaded. The hard disc memory was 18916 K bytes. Even with a maths co-processor on board the calculation time became excessive.

When the usage of the computer was taken into account, it became necessary to justify a new computer. This was done on a comparison basis and the report is included in Appendix 2.

The programs (software) was used on the Victor P.C. were:

- Lotus Release 2.01
- P.C. Model version 8.40 XP - 01\01\88

2.7.2 COMPAQ 386S PERSONAL COMPUTER

Technical Details

This was a personal computer produced by:

Compaq Computer Limited,
Ambassador House,
Paradise Road,
Richmond,
Surrey. TW9 1SQ

The specification of the computer was as follows:

- 1 company 386S Model 40
  One Mb 16 bit memory
  One 5.25" 1.2 Mb disk drive
  One 40 Mb fixed disk drive

- 1 3.5" 1.44 Mb disk drive
  1 4 mb RAM module
  1 VGA colour monitor
  1 MS DOS

The Compaq 386 utilises the Intel 80386SX, 16 MHZ chip, which gives a 60% increase in speed over the Victor 286.

The compaq P.C. was purchased in September 1989, purchased by DCE, together with an IBM Proprinter III x L Printer, so that hard copies of data could be obtained.

The cost of the whole package was: £4,613.25.

The programme Software used on the Compaq P.C. were:

- Lotus Release 3.0
- P.C. Model version 8.40 XP - 01\01\88

CHAPTER 3

INITIAL MODELLING

3.0 INTRODUCTION

With the process to be examined decided on, Cut\Punch\Fold for sheet steel. The promise of benefits to be gained has been shown in similar projects, and the software to be used chosen, LOTUS 123 and PC MODEL. The actual data base construction and computer modelling has now to be started. This chapter shows the progress through the initial stages of this process, up to the review of the data base and the change in method of reading the data into the model from sequential to simultaneous, and the results obtained.

3.1 DEVELOPMENT OF THE LOTUS 123 DATA BASE

The Lotus 123 Data Base was originally conceived as an all embracing data base for the Cut\Punch\Fold components manufactured by the Salvagnini\Behrens\Hammerle machines, so that any information required for a part would be readily available. Five different Departments in DCE colaborated in its construction:

The Design Drawing Office
The Works Supervision
The Process Planning Department
The Work Study Department
The Production Planning Department

Appendix 3.1.1 - diagram showing the route of data collection and the people involved. The data base was started in mid 1988 and completed in mid 1989. As with all data collection of this size there were inaccuracies and for the last 6 months only 2 people were working on the file so that as the product range changed the data base did not.

Initially Mr. S. Richards of the Design Drawing Office, grouped like shaped components together (see appendix 3.1.2) although the gauges might have been different this was done to minimize the setting time between jobs on the Hammerle (Brake Presses), putting through a family at a time.

Then the author placed the 'typical week' quantities onto a copy of this A4 lever arch file 'Families Part Folder'. Then placing all the part numbers onto the main frame computer and obtained the machining times for the components (see Appendix 3.1.3). These lists were then handed to Mr. T. Wells who placed them onto the Lotus 123 spread sheet with Mrs. A. Greaves help. He also added the gauge, punches used, tape numbers, total tool quantity, and a description of the component, (appendix 3.1.4)

Then with the help of Work Study Department any machinery times that were missed were added and with the help of the Works Supervision using their 'Product Knowledge' to put the families in the correct order. At a later date the drawing number of each component was added as requested by the Design Drawing Office.

A family name, a unique number in the correct order of families to be processed was added. During the compiling of the data base it was found that the families grouped themselves very much by gauge, falling into 2 categories above 1.6mm or 1.6mm and less. This helped the manufacturing process greatly as this was the gauge break of the machines, ie. different punch dies and folding tools were used, above and below 1.6mm.

HOW THE DATA LINK FROM LOTUS 123 TO THE PC MODEL WAS ACHIEVED

With the Lotus being able to link with PC Model, it was intended to use the data base as the data for the simulation model. To this end quantities which come from a typical week were included. A typical week was obtained by taking the sales figures of units for the last 3 years, averaging them. Finding a week when this production requirement was met, then using the Burroughs main frame computer the unit was broken down into constituent parts with the quantity required given. When comparing these 'average' figures with actual production figures they had an acceptable error.

To transfer the Lotus data file from Lotus to PC Model, it's format had to be changed. To do this the following actions were taken, to obtain the form found in Appendix 3.2.3.

- Insert a line number on left of the file (Numbers 1 - 999 consecutively (PROD).

- Delete the first and last part of the family name (FAMILY)., i.e. F03.0210.003, became 0210, and was inserted to the right of PROD.

- Move the quantity per week (QTY) so that it was to the right of FAMILY.

- Examine the gauges and routes required by the job and insert the relevant number to the right of QTY.

- The priority (PRTY) was then inserted to the right of this. The priority number starting at zero and ending at 255.

- The allowed hours were changed by dividing them all by 1.67 to give actual hours. The setting hours also being actual hours required to do the job.

- The hours being arranged so that from the left they read CUT\PUNCH\BEND.

- An ASCII code value was placed to the right of this.

- All other data from the file was then deleted.

The file was then transferred out of Lotus as a print file. Being changed into a data file when it was input to PC Model. As can be seen from referring to Appendix 3.1.4, this was a laborious task, with any errors causing the simulation to stop. (Appendix 3.3.3).

HOW THE DATA LINK FROM PC MODEL TO LOTUS 123 WAS ACHIEVED

The link from PC Model to Lotus 123 was achieve using software supplied with PC Model. Before running the model a file was opened to store the utilisation data (results). At the end of the model run this file was converted using the software supplied into a form that Lotus 123 would accept, and then transferred from PC Model to Lotus 123 for further graphics and graphical representations of the result.

The Lotus data base, whilst being continually modified\added to\revised during the project, the next major change occurred with the review of the whole data base as outlined in Chapter 4.

3.2 DEVELOPMENT OF THE SIMPLE FACTORY MODEL

The development of the simple factory model started in February 1989 through a liaison with Coventry Polytechnics' Mechanical Engineering and Manufacturing Systems Department. The model constructed was that that DCE Management envisaged for the new routing of work through the factory. The initial overlay (Drawing DCE15.OLY) is shown in Appendix 3.2.1. The routing to be followed is shown below in Diagram 3.2.

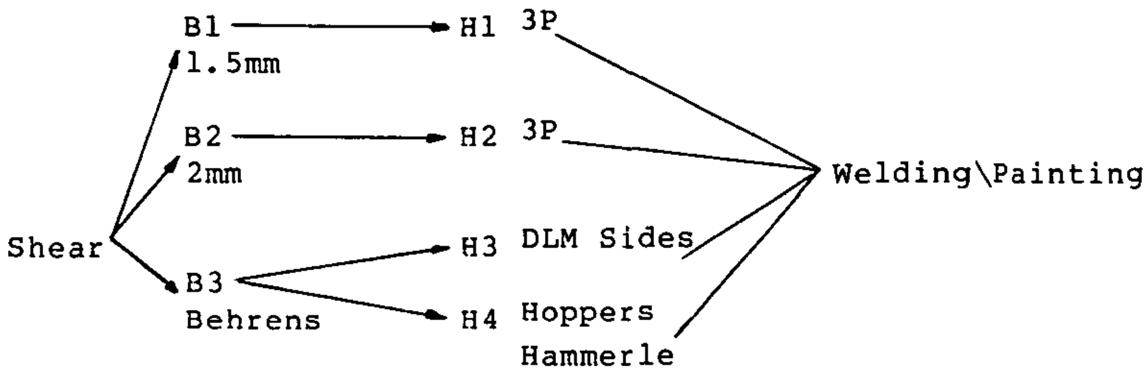


DIAGRAM 3.2 SHOWING PROPOSED ROUTING OF JOBS

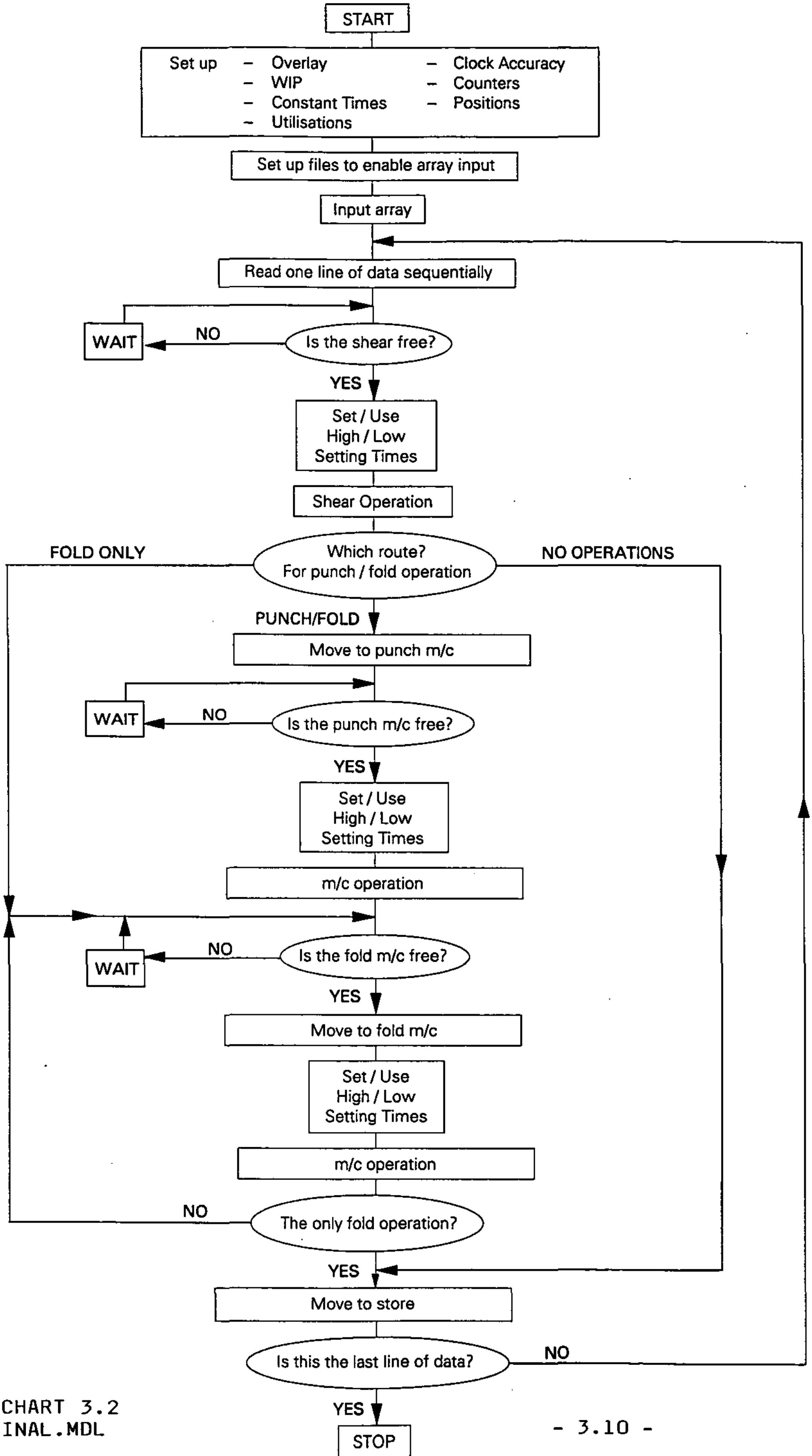
The flow chart for DCE\FINAL.MDL, the final program in this section can be found overleaf.(Flow Chart 3.2) This shows the reasoning and order of the programming. The initial program controlling the routing is to be found in Appendix 3.2.2. (DCE\DCE16.MDL). In this programme the initial link from Lotus to PC Model to Lotus has been achieved. There are only 4 routings included in this model. With only one of those punching and folding. The other just punching the material and sending it on to the end of the simulation.

The data input from Lotus 123 to PC Model was via an ARRAY (listing), a test Array for the DCE\DCE16.MDL is included (Appendix 3.2.3.). Examining the ARRAY (DCE\DCE16.DAT).

- The first column (PROD) is a sequential number. Ending in 999 to tell the simulation model the file's end has been reached.

- The second column (FAMILY) is the family identification number. The simulation model records this figure and changes the setting time from % SET CUT - SAME VALUE to % SET CUT - DIFF VALUE when this number changes.

- The third column (QTY) is the quantity of the job. This is multiplied by the time to give the correct delay time of the relevant machine.



FLOW CHART 3.2
DCE\FINAL.MDL

```
SV(%PUNCH,%%SCHED(#SPUN,OBJ@1))       ;Set value
AO(%PUNCH,*,%%SCHED(#SQTY,OBJ@1))     ;Arithm. operation
WT(%PUNCH)                            ;Wait
```

- The fourth column (ROUTE) is the route that the job will take through the simulation. In this case 1, 2, 3,5,6. The routes taken by jobs in the DCE and JEF directories are shown in "DCE\JEF DIRECTORY - ROUTINGS". (Appendix 3.2) This appendix shows an overlay with the machines labeled with the routes showing to which machine the job will travel. At the left hand end of the overlay "Pallets" are indicated, this is where the jobs first appear on screen representing the back of the Salvagnini Shear. The jobs leave the screen on the right hand side in "welding, etc"

- The fifth column (PRTY) is the priority of the jobs varying from 0, the highest, to 255, the lowest.

- The sixth column (CUT) is the time for the job to take in the cutting process per part, in Hours : Minutes : Seconds.

- The seventh column (PUNCH) is the time for the job to take in the punching process per part, in Hours : Minutes : Seconds.

- The eighth column (BEND) is the time for the job to take in the bending process per part, in Hours : Minutes : Seconds.

- The ninth column (ASCII) is a number which is interpreted by the simulation model and shown on the screen a block identifying that job during the simulation.

The results of this Array being run through the simulation are shown in Appendix 3.2.4. This shows the throughput of jobs in hours so that in the first hour 6 jobs were completed. No utilisation percentages have been shown.

The first model was developed through the iterations:

DCE\DCE17.MDL
DCE\DCE18.MDL
DCE\DCE19.MDL
DCE\DCE21.MDL
DCE\DCE22.MDL
DCE\DCE23.MDL
DCE\DCE26.MDL
to DCE\FINAL.MDL

Where DCE	\ FINAL.MDL
Is the Directory that the program is stored in on the computer.	Is the name of the individual program with Arrays\Overlays\ etc. having the same name as the program.

The main changes to the programs will be highlighted, with appendices where necessary. A full print-out of DCE\FINAL.MDL is given in Appendix 3.10.1.

Iterating on DCE\DCE16.MDL the overlay has been slightly changed (DCE\DCE17.OLY). The second route no longer being present from the cutting process, but coming from the main branch before the Behrens. The program has been greatly added to, effectively doubling its length. New programming includes:

- A set work in progress limit (L=(10))
- A set simulation clock accuracy (C=(0))
- Machine locations (*B32=(XY(26,6))
- Machine utilisation (U=(1,B_BUF1,*B12))
- The option of setting the times immediately at the Shear (SV(%%Prod_Times(#PFLG,OBJ@1),OBJ%1)
- Jobs only moving from one machine to the next if its buffer has no jobs in it, ie. no queueing at the second machine.
- 8 routes for jobs are completed with punching and folding operations.

The changes that Array has undergone (DCE\DCE.DAT) are:

- The family numbers have been changed, so that both setting times can be checked.
- The Route numbers now read from 1 to 8 to check all 8 possible routes.

The results of this Array being run through the simulation are shown in Appendix 3.3.4. The utilisation of the various machines and buffers are now given as they have been programmed into the simulation. Explaining these:

Total Hours = 10 (The total simulation time)
Total Tools = 16 (The number of utilisations shown)
For B BUF 1.
MEAN = 31.15 (The mean of the distribution)
STD-D = 38.09 (The standard deviation of the distribution)
MAX = 100.00 (The maximum value)
MIN = 0.00 (The minimum value)

Hour B_BUF1
1 42.63 (The square B_BUF1 was occupied 42.63 % of hour 1)

With the results now shown in Lotus 123, the Lotus graphics package can now be used.(Appendix 3.3.5. for example of B_BUF1) In developing DCE\DCE17.MDL to DCE\DCE18.MDL (Appendix 3.4.1), the following revisions were made:

- The single loop (link statement) added at the end of each routing was changed to the move statements at the end of each routing.

- Both set value statements for %% PROD TIMES have been omitted.

- The time counter for the job completion times has been omitted throughout the programme.

The first change makes no difference to the logic of the program, it is just a more laborious task to type out. The second and third changes account for differing utilisation statistics. The overlay and data files are the same as those used in DCE\DCE17. MDL.

In developing DCE\DCE18.MDL to DCE\DCE19.MDL (see also Appendix 3.5.1) the following revisions were made:

- An extract column was added to the array.
- The logic controlling the setting time for the machines was changed.
- The set value statement deleted in DCE\DCE18.MDL are now included.
- The time counter for the job completion times are now included.

The first change aimed to give an identification column in the array. This failed as the column was transferred as zeroes. The second change was to control the setting time. Previously a new batch of works' setting time started as soon as the last batch of work began to be processed by the machine. With the new modification the set up time only starts when there is no work being operated on.

By comparison with DCE\DCE18 the utilisation statistics it can be seen that the change in setting time increases the machinery cycle time. The overlay and data files are the same as those used in DCE\DCE17.MDL.

In developing DCE\DCE19.MDL to DCE\DCE21.MDL (see also Appendix 3.6.1) the following revisions were made:

- The end of the file has changed from 999 to 9999. To allow for larger data arrays.
- The setting times are more defined, having a different setting for the Behrens and the Hammerle.

- The identification column has been deleted in the array and program.

- The (link) loop for putting working into stores has been included throughout the program.

- The control of setting has reverted to that previously used.

The first change, increasing the size of the files will enable the entire data file to be read into the model. The last four changes are program changes to obtain the most efficient running of the program. By comparison to DCE\DCE19 utilisation statistics, there are changes that originate from changing the setting times. Although DCE\DCE18 and DCE\DCE19 approach each other. The overlay and data file is the same as used in DCE\DCE17.MDL.

In developing DCE\DCE21.MDL to DCE\DCE22.MDL (see also Appendix 3.7.1.) the following revisions were made:

- A new overlay, DCE\DCE22.OLY was used. With loops in the routes to allow more jobs onto the simulation.

- A new data file, DCE\DCE22.DAT was used, to allow a new Route (Route 9) to be included.

- The number of different setting times were increased to give greater sensitivity.

- The name labels in the utilisation statistics were changed to coincide with the name changes on the overlay.

- Various distance moves have changed values to maintain parity with the overlay.

- TP(*Store) - Test position of the store has been deleted as no work waits here.

By comparison to DCE\DCE21 the utilisation statistics there are changes to the results, which come from adding of Route 9, changing the data file together with the different distances moved.

In developing DCE\DCE22.MDL to DCE\DCE23.MDL (see also Appendix 3.8.1.) the following revisions were made:

- A counter inserted so that each job has a different colour.

- The end logic changed so that as soon as the last job was completed the timer (clock) was to jump to the next 10 hour period and stop.

- A move of 7 places to the right (MR(7,%move)) was inserted between the Behrens and the Hammerle Machines.

Neither of the first two points worked satisfactorily. The counter changing to object did not aid the identification of symbols to any great extent. The end logic proved problematic. None of the jobs would exit from the screen and so with a large data file the model 'locked up' and gave meaningless results. By comparison to DCE\DCE22.MDL the utilisation statistics, the changes are notable in that only one item passes through DCE\DCE23.MDL the other columns of the statistics coming close. The overlay and data files used are the same as those used in DCE\DCE22.MDL.

In developing DCE\DCE23.MDL to DCE\DCE26.MDL (Appendix 3.9.1) the following revisions were made:

- No Route 9

- There are only 4 setting times

- The programme logic to obtain the correct setting times

The first two points are reverting back to previous programs. The third point is ensuring that as a new job is processed on a machine no other jobs can be moved into the buffer station and the setting time started.

By comparison to DCE\DCE23.MDL the utilisation statistics the results were found to be somewhat different due to the lack of Route 9. Due to the setting times the run time for DCE\DCE26.MDL was longer. The overlay used was DCE\DCE26.OLY the same as

DCE\DCE17.OLY with only machine names changed. The data used was DCE\DCE26.DAT.

The cutting process always taking 0.5 hr (listed in seconds)
The punch process always taking 1 hr (listed in seconds)
The bend process always taking 1 hr (listed in seconds)

In the development of DCE\FINAL.MDL (Appendix 3.10.1) all the advantageous modifications of the previous computer simulation programs (DCE\DCE16 - 26) have been utilised. In this program the report array has been investigated and corrected. The report array records the time of a job:

- Through cutting (# PFLG=(0))
- Through Cutting\Punching (# PBEH=(1))
- Through Cutting\Punching\Folding (# PHAM=(2))
- Through Cutting\Punching\Folding\Welding (# PPROD=(3))

See diagram 3.10.4 below

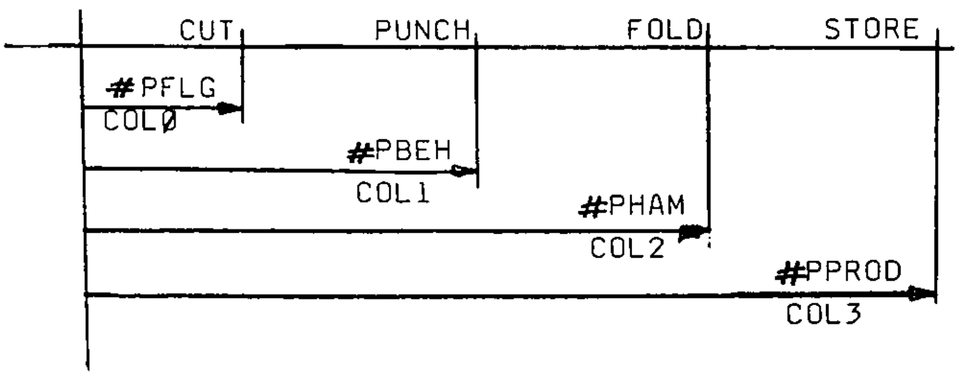


DIAGRAM 3.10.4 SHOWING WHAT LINES THE REPORT ARRAY SHOWS

There are extra lines added to the program as itemised in the remarks columns of Appendix 3.10.1:

- New % work counters have been added.

- SV(OBJ%1,CLOCK) has been moved to set the time before the job is started by the Salvagnini.

- The reporting statements values have been changed to correspond with the new counters.

- Some of the logic to obtain the correct process times for the Report Array have been changed.

After the Test Data Array had been run through the program, the Report Array was viewed on screen. These results are not the same as the Data Array, as there is an added time factor for queueing, waiting in stores, and movement of the job from one machine to the next.

By comparing the utilisation statistics with those of DCE\DCE22.MDL the results for the job completion are similar (throughput), although the utilisation for job completion are different. This was due to the difference in the setting logic. The overlay used was the same as in DCE\DCE22.MDL, DCE\DCE27.OLY (Appendix 3.7.1.) The data used was the same as in DCE\DCE22.MDL, DCE\DCE27.DAT

In summary of section 3.2, Development of the Simple Factory Model:

- The section reviews the changes made to the simulation model during its development. Results have been given for the same or similar data throughout.

- The Final Model (DCE\FINAL.MDL) is a compilation of all the advantageous points of the previous models and utilises the Report Array in its correct format.

- The section ends as the next development involved a major change in the models logic in reading data. This took place after the model DCE\FINAL.MDL.

3.3 RESULTS OF INITIAL MODELLING

The following section gives the results obtained from the Lotus file during the time the DCE\ file PC Model computer simulation models were being developed and used. The results have been presented separately for clarity.

3.3.1 RESULTS FOR 123 LOTUS DATA BASE

With the Lotus file nearing completion it was necessary to total the Lotus machining and setting times. The first problem encountered was that the times for machining were "Allowed" hours from the computer. These had to be changed to "Actual" hours, the length of time that it would actually take to machine a component. How this was achieved is shown below:

1.00 Hours timed - the job takes this time
+12% Rest allowances and contingencies

1.12 Hours	JOB TIME
x 1.67	Wages (Historic Addition)
1.87 Hours	Allowed Time

So to obtain the Actual Time the Allowed Time is divided by 1.67.

The Setting\Change over times allowed, were estimated :

0.1 hrs for parts within a family.
0.3 hrs for a change in family.

and called Judgement Times (J). These represent change over times for similar components and change over time for different gauges and dis-similar components. The table of times given in table 3.3.1. giving the Behrens Punches, and then the Hammerles Brake Press folding times from these machines, obtained from the Lotus file at this time.

MACHINE	MACHINING ACTUAL HRS	JUDGEMENT ACTUAL HRS	TOTAL PER MACHINE
Behrens - 5 Non-Laser Complex Parts	2.5	4.5	7.0
Hammerle - 7	3.2	3.9	7.1
Hammerle - 9	2.7	1.4	4.1
Behrens - 6 Laser with gauge	78.	77.1	155.1
Hammerle - 4	10.4	11.1	21.5
Hammerle - 5 (N.C.)	36.6	40.6	77.2
Hammerle - 6	4.2	6.6	10.8
Behrens - 7 Laser low gauge	69.3	50.4	119.7
Hammerle - 1 (N.C.)	49.6	41.1	90.7
Hammerle - 2 inc. (H2\H1)	13.5	6.5	20.0
Hammerle 3\1\3	34.7	0.7	35.4
Hammerle - 6	0.1	0.1	0.2
Hammerle - 1\Nibbler	1.0	1.7	2.7

TABLE 3.3.1 - ACTUAL TIMES FROM LOTUS 123

DATE 25/05/89

As can be seen from Table 3.3.1. the non-laser Behrens is very highly loaded. By receiving the components flowing through the various machines, the following balancing of times was achieved in Lotus 123.

MACHINE	MACHINING ACTUAL HRS	JUDGEMENT ACTUAL HRS	TOTAL PER MACHINE
Behrens - 5 Non-laser	37.1	31.0	68.1
All Routes other than N.C. 53.9	28.0	81.9	
Behrens - 6 Laser	50.3	57.0	107.3
Hammerle - 5 (N.C.)	43.1	44.0	87.1
Behrens - 7 Laser	63.5	46.0	109.5
Hammerle - 1 (N.C.)	88.6	46.0	134.6

With the Behrens working double shifts, these figures imply that the largest operating machine is the Hammerle 1, at 134.6 hrs. At 40 hrs per week this implies that the work will take nearly 3.5 weeks.

3.3.2 RESULTS FOR PC MODEL

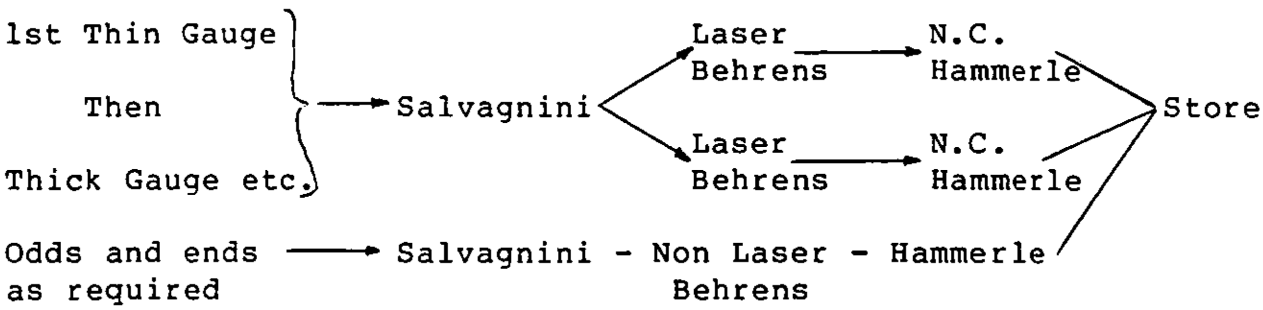
The first results from the computer simulation were obtained on 27th June 1989.

The original Lotus 123 data file being WKLYLDNG.WK1. This was changed onto an acceptable form for PC Model and called DCE.DAT. The results file from this data file was DCE.WK1. The simulation run time was 153 hrs. On the same program with the same constant values (movement time\WIP\clock accuracy) the following results were obtained:

- The data file JDCE.WK1, Actual times for machining gave the results file JDCE1.WK1. The simulation run time was 113 hrs.
- Changing all the priorities of jobs to the same value gave the same simulation run time at 113 hrs.

The data file was then arranged so that a minimum run time was achieved. The following reasoning was used:

Work



The simulation run time dropped to 68 hrs. Then re-arranging the data taking into account the laser parts gave a further reduction in simulation run time to 63 hrs. These results were questionable as the simulation program at the time was not totally correct.

Assuming that the results are constant with errors. The method of scheduling work through the simulation is very important. Giving a 26% reduction in time from the allowed hours to actual hours, and a further 44% reduction in completion time from the first Actual hours simulation run to the last.

With further changes to the program file the following data files were run through the DCE\DCE21.MDL to obtain the minimum run time possible on this will ultimately dictate how many shifts on what machines will be required. Again there were errors in the data files as itemised. The result being that the program ends at the first error.

The settings throughout the following simulation runs were:

```
L=(10)                         ;Work in Progress Limit = 10
C=(0)                          ;Clock Accuracy = Seconds
%Move=(0:00:10.00)          ;Movement = 10 Seconds per move.
%Set_Same=(0:06:00:00)      ;Setting times
%Cut_Same=(0:00:05.00)
%Set_DiffB=(0:18:00:00)
%Set_DiffH=(0:18:00.00)
%Cut_Diff=(0:00:05.00)
```

The results were as follows:

<u>JDCE.DAT</u>

This file contains the original routes and actual times for the typical weeks jobs, being processed as in the Lotus file. The original Run time was 102 hrs. This changed to 247 hrs after the following data file errors had been found and rectified.

Line 719, 1221, 1525 - wrong line numbers.

<u>J2DCE.DAT</u>

This file contains the same data. The routes have been attempted to be balanced using families. The original run time was 77 hours. This changed to 205 hours after the following data file errors had been found and rectified.

Lines 719, 1221, 1525 - wrong line numbers.

J3DCE.DAT

This file is shorter than JDCE DAT and J2DCE DAT, as all the zero quantities and multiple lines have been deleted. Running as the family part file had been constructed. The original run time was 155 hours. This changed to 260 hours after the following data file errors had been found and rectified.

Lines 744, 811 - line missing.

J4DCE.DAT

This file is J3DCE DAT with an attempt to balance the routes by putting families onto different machines. The original run time was 155 hours. This changed to 216 hours after the following data file errors had been found and rectified.

Lines 744, 811 - line missing

J5DCE.DAT

This file was edited to optimise the first 100 lines and then the next etc. The original run time was 139 hours. This changed to 236 hours after the following data file errors had been found and rectified.

Lines 744, 811 - line missing.

These results show that if the data file is not correct the modifications to it can in fact make the processing times longer (J4DCE.DAT to J5DCE.DAT).

What has not yet come through is running the gauges high \low \high etc. along the specified routes. Although the original concept of working through the families as they were listed was very quickly found to be inefficient.

Thus for the data has been run using a work in progress limit of 10 leaving all the other settings in DCE\DCE21.MDL the same and only using J5DCE.DAT the effect of changing the work in progress limit was then investigated.

W.I.P.	Run Time (to the next whole hour)	
1	1324 hrs	
5	347 hrs	
10	236 hrs	
50	202 hrs	constant queues forming
100	201 hrs	
250	201 hrs	
500	201 hrs	
750	201 hrs	
1000	201 hrs	
1500	201 hrs	

(More than total number of jobs in data file (1081) lines).

As can be seen from the above figures and graph 3.3.1. the improvements in total run time are dramatic when initially raising the W.I.P. limit, but as soon as constant queues start forming (W.I.P.=50) there is no improvement in the run time. When considering what is happening, the model only has a set amount of storage space and the machines could only work so fast, so that the model became saturated and the minimum run time was achieved. When comparing this to reality, the factory could become congested with excessive W.I.P and the times for job completion would be expected to increase not stay constant in the model due to confusion and time taken to move jobs to get at the job required, the trouble with controlling the W.I.P. becomes horrendous.

The next part of the program to be investigated was the clock accuracy. All other settings remain constant as before:

C=(0) One second resolution of clock
Run time = 236 hrs.

C=(1) Tenth second resolution of clock
Run time = 237 hrs.

C=(2) Hundredth second resolution of clock
Run time = 237 hrs.

If the clock is set at C=(0) the time delay parameters will be treated as precision units, eg.

```
WT(113) - will delay 113 seconds when C=(0)
        - will delay 11.3 seconds when C=(1)
        - will delay 1.13 seconds when C=(2)
```

The waiting took a shorter time using a higher precision, this must be due to the queueing factor building up in the simulation. Simcon claim that "using higher precision will degrade performance especially in clock increment mode". This, together with the number of hours run forced the approach of using C = (0) one second resolution of clock as it was more than accurate enough for the trials being undertaken.

The effects of changing the job movement time were investigated:

%Move=(0:00:00.00)	Run Time = 225 hrs (to next whole hour)
(0:00:00:01)	225 hrs
(0:00:00:10)	225 hrs
(0:00:01:00)	228 hrs
(0:00:10:00)	236 hrs
(0:01:00:00)	296 hrs
(0:10:00:00)	1184 hrs
(1:00:00:00)	6343 hrs

As can be seen from the above figures and graph 3.3.2., the time taken to move a job between machines has a dramatic effect on the time taken to complete the jobs for that week.

Upon reviewing the result it was decided to lengthen the lines between the Salvagnini and the Behrens punch machine. Thus allowing for more work in progress to build up and hopefully reduce the run time. This was run on DCE\FINAL.MDL. The data file used was again DCE\J5DCE.DAT. The settings throughout the following simulation were:

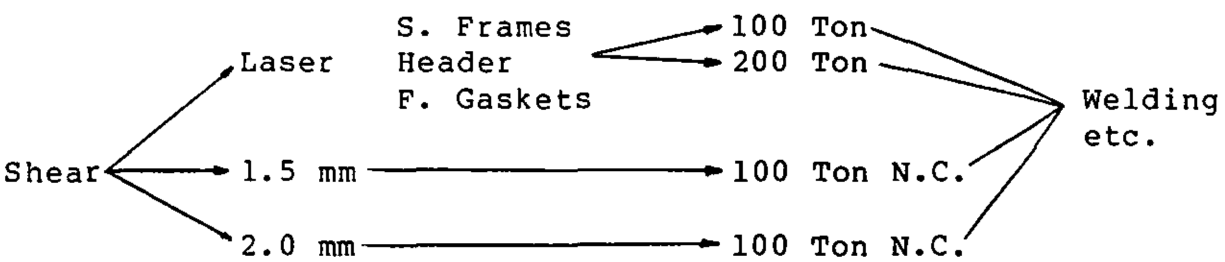
```
L=(10)                    ;Work in Progress limit changing.
C=(0)                     ;Clock accuracy - seconds.
%Move=(0:00:10.00)        ;Movement = 10 seconds per move.
%Set_SameB=(0:06:00.00);Setting times
%Set_SameH=(0:06:00.00)
%Cut_Same=(0:00:05.00)
%Set_DiffB=(0:18:00.00)
%Set_DiffH=(0:18:00.00)
%Cut_Diff=(0:00:05.00)
```

W.I.P.	Run Time (to the next whole hour)	
1	1384 hrs	
5	347 hrs	
10	265 hrs	
50	199 hrs	constant queues forming
100	199 hrs	
250	198 hrs	
500	198 hrs	
750	198 hrs	
1000	198 hrs	
1500	198 hrs	

(More than total number of jobs in data file (1081) lines).

As can be seen from he above figures and Graph 3.3.3., there is a very close similarity between the modified FINAL.MDL programme and the original DCE21.MDL program, ultimately only 3 hours are saved by being able to put considerably more work in progress into the simulation.

The following results were obtained after a request from the Manufacturing Operations Review Committee on 8th August, 1989. As follows:



The only jobs to be considered laser job were seal frames, headers, and fan cases. All other jobs were to be processed on the 1.5mm or 20mm routes. These jobs were then grouped together and run separately through the simulation. The results were as follows:

SIMULATION MODEL TOTAL RUN TIME

BEHRENS MACHINE NAME	TIMES NO MOVE. NO SET.		NO MOVE. H. SET. 0.1\0.3	NO MOVE. B+H SET.	MOVE. 5 SECS. SET.	MODEL NO.
DCE A	55	HRS	75	75	75	DCE22
1.5MM	56		-	91	98	DCE26
DCE B	21		29	29	29	DCE22
LASER	21		-	33	36	DCE26
DCE C	95		118	125	125	DCE22
2.0MM	101		-	159	170	DCE26

The work in progress limit was set at 200. All other variable remained constant throughout.

When the three files are combined with the same parameters as above and no times for setting or moving the total run time is 130 hrs for DCE22. From the Lotus file the following times were obtained:

BEHRENS MACHINE NAME	1.5mm	2mm	Laser	Totals	MODEL NO
ADCE	2 hrs	50	13	65hr	DCE22
1.5MM	2	51	13	66	DCE26
BDCE	-	1	20	21	DCE22
LASER	-	1	20	21	DCE26
CDCE	95	18	1	114	DCE22
2.0MM	97	19	1	117	DCE26
	97	69	34	200	DCE22
	99	71	34	204	DCE26

Total machine hours = 200 hours - DCE22
204 hours - DCE26

It can be seen that there is a discrepancy between the Lotus and PC Model results, this is due to a queuing function. This comes about due to the interaction of parts with simulation model.

A change was implemented into programming in DCE\DCE22.MDL, there was a PULL system in operation between the Behrens\Hammerle\Welding, the jobs only move from one machine to the next, if the next machine is free. In DCE\DCE23.MDL there was a PUSH system in operation between the Behrens\Hammerle\Welding, the job moves immediately to the next machine, forming queues at these positions.

SIMULATION MODEL TOTAL RUN TIME

BEHRENS MACHINE NAME	TIMES NO MOVEMENT NO SETTING	NO MOVEMENT H. SETTING 0.1\0.3	NO MOVEMENT H\B SETTING	MOVE 5 SECS H\B SETTING
1.5MM 23.ADCE	52 HRS	74	74	75
LASER 23.BDCE	19 HRS	26	27	27
2.0MM 23CDCE	86 HRS	107	115	116

The work in progress limit was set at 200. All other variables remained constant throughout, as set previously.

From the Lotus file the following times were obtained.

DATA FILE NAME	1.5mm	2mm	Laser	Totals
23 ADCE	4.8 hrs	2	11	61
23 BDCE	1	-	18	19
23 CDCE	14	86	1	101
	63	88	30 hrs.	

TOTAL - 181 Hours

The data from the simulation model has been plotted on graphs (Graphs 3.3.4\3.3.5\3.3.6). In Graph 3.3.4, the loading of DCE22 the machines have a buffer stock and the setting does not take place after one job and before the next job can be machined. The loading of DCE26 starts at the same position, but has no second value the third and fourth values are greater than DCE22 as no job can start or have its setting done while another job is still on a machine. DCE23 follows the same logic as DCE22, but uses a PULL system not a PUSH system to achieve job transfer between machines.

The same reasoning applies to Graphs 3.3.5 and 3.3.6 where there is a greater difference between DCE 22 and DCE23 values. When comparing the values from the Lotus 123 file and the PC Model simulation, there was a difference, this was put down to the time taken to move jobs and queueing function. The time taken waiting for jobs to be completed on a machine before the next job can be started.

The next development in the programming was to move away from scheduling work, to that of having all the machines working simultaneously, necessitating a move from SEQUENTIAL reading of the Data Array to SIMULTANEOUS reading of the Data Array as is expanded upon in Chapter 4.

C H A P T E R   4

D E V E L O P M E N T   O F   T H E   M O D E L

4.0 INTRODUCTION

With the project having run for more than a year, a review of the data base was called for, the reasons and results of this are given in this chapter.

The modelling had reached the stage where the logic for the data to be read into the program was to be changed from sequential to simultaneous, so that no scheduling could effectively take place. The development and results obtained from this change in logic are expanded upon in this chapter, with results.

Again in this chapter the number of computer print-outs have been limited, with only the final program JEF\DCE54.MDL printed in its entirety in Appendix 4.2.1.

4.1 DEVELOPMENT OF THE LOTUS 123 DATA BASE.

Although the Lotus data base was being updated throughout the project, upon using it for analysis its accuracy was brought into question. The results of the analysis can be found in section 4.3. Having identified the need to review the data file, this was carried out using the small team of 4 people, on the 7th and 8th September, and 16th October 1989. A new list of quantities and actual product breakdown were obtained and the data base was examined in some detail, the following anomalies being apparent:

- Some components had an incorrect part description.
- Some components did not have the correct quantities required to meet the weekly schedule.
- Components that entered the P1 store were still listed, and in many cases, the quantities and times were doubled up.
- There were no (S.I.) Sintamatic Insertable or (S.C.) Sintamatic Cased units listed on the file.
- The components of whole units (DX7) were omitted on the original listing by the Burroughs computer and were not included in the file.
- The stainless steel jet tubes were omitted by the Burroughs computer, although the units containing them were listed.
- There was an inconsistency in weekly quantities (ie. handed parts not always had matching quantities), and\or compatible parts within an assembly did not match assembly requirements.
- It was believed that subsidiaries\spares requirements are not consistent throughout the data base.
- The components that were 'bought-out' have been included.
- Requirement quantities appear on some obsolete, and corresponding current listed components (both being listed).
- Obsolete units (Unimaster UMA 350\350V) have been included.
- The Burroughs main frame computer product breakdown was not the same when the request was repeated.

Generally it was observed that most parts for assemblies were listed and any obvious omissions were not apparent. In the data bases original form it was considered to be 60% accurate, but with the review changes having been made, it was thought that it was improved to 85% accurate.

The next major changes to the data base were the questioning of setting times, and further itemised changes to the file. These are expanded upon in Chapter 5.

4.2 DEVELOPMENT OF FACTORY MODEL

As the results became available from the computer simulation, DCE Management were initially unhappy with the long duration of various machines in activity. This occurred when the simulation was run as planned due to the simulation data input being SEQUENTIAL one line then the next. When the routes available were balanced by scheduling the week was completed in a far quicker time. But scheduling was not trusted and so a method of reading the data simultaneously was required to minimise the machinery inactivity. This lead to the development of JEF\DCE50.MDL.

The flow chart for JEF\DCE54.MDL the final program in this section can be found overleaf (Flow Chart 4.2) showing the program structure. It can be seen that the logic of the movement between th machines in the same as in the DCE\FINAL.MDL range of files.

In developing JEF\DCE50.MDL the starting point was DCE\DCE21.MDL the major difference being that the method of data being read into the simulation has been changed. In the DCE directory the data was read in from the data file SEQUENTIALLY from the start of the data file to the end. In the JEF directory the data is read in SIMULTANEOUSLY by routes. The model in the first line of the first route then finds the first line of the next route, until all the routes have been read into the model. Then repeats the operation from the start of the file.

As the length of the array, the number of columns has to be changed, the controls of the program reading these columns should also change. This is not so in all cases, so giving inaccurate results. In the utilisation statistics the first job that previously has not registered sets up all the counters and sets the various flags. This raises the number of jobs from 18 to 19. The time taken to run the simulation is now 9 hours, but with the array data being read in certain cases the results are not valid.

The program uses DCE\DCE26.OLY overlay. The program uses JEF\DCE 50.DAT, this has the similar data to previous files. But has a new column labeled PART, a consecutive number, used to identify the contents of the file. All of the jobs in the Array have the same priority of 1.

In developing JEF\DCE50.MDL to JEF\DCE52.MDL (see Appendix 4.1.1) the following revisions were made:

- LK(!Move) - Deleted, logic at start of program.
- The columns been read from the array have been changed from OBJ@2 to OBJ@3
OBJ@4 to OBJ@5
To correct the errors in the program.

By comparison with JEF\DCE50 the utilisation statistics, the time taken to complete the job drops from 9 to 8 hours. This is due to the correct data now being used.

The program uses JEF\DCE52.OLY (Appendix 4.0.1) this overlay extends the distance travelled before the jobs enter the Behrens Machines. All jobs are now read into one point on the screen.

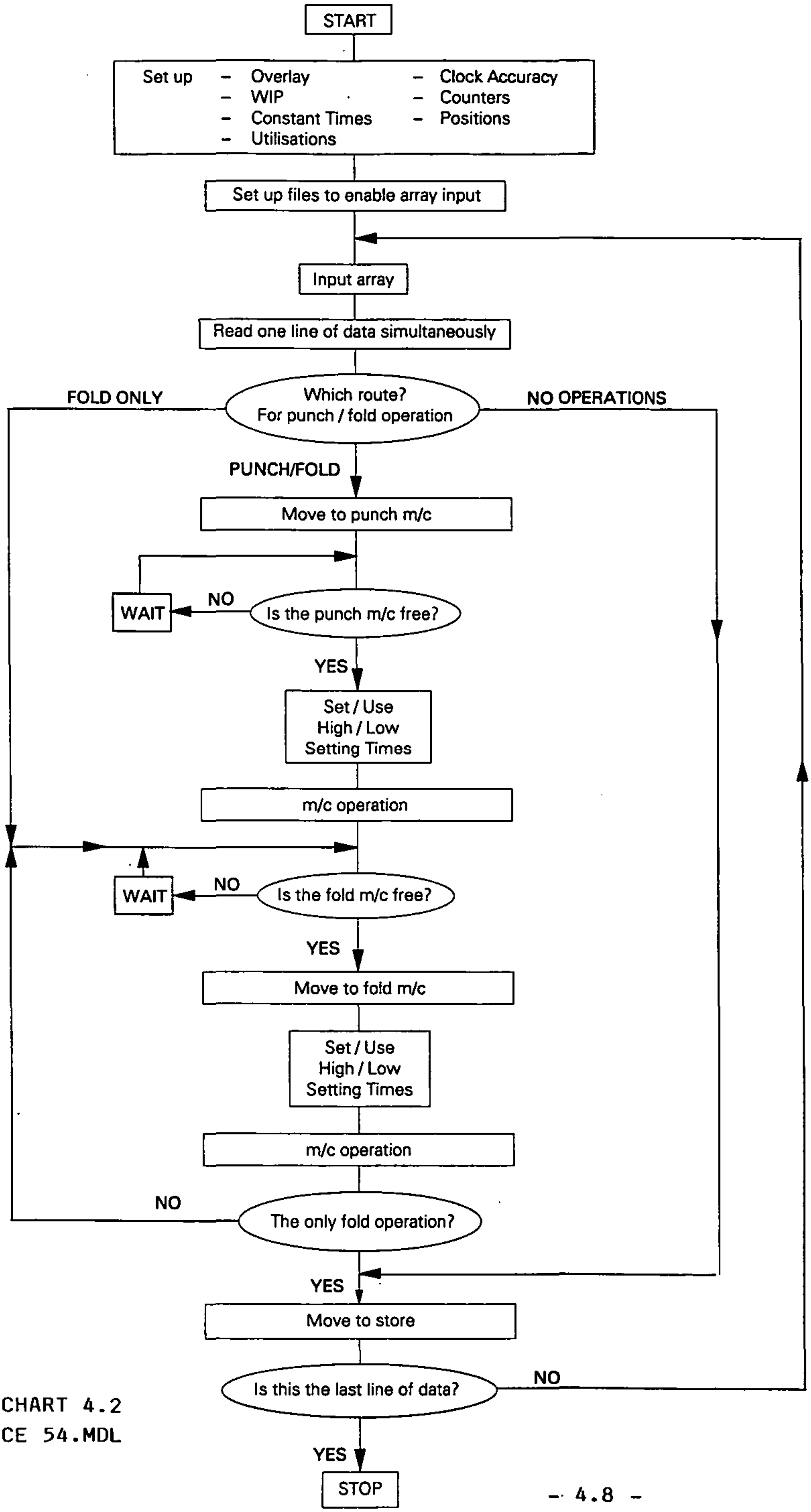
The labels on screen "No. of Batches =" and "Time Taken =" are not used in this program.

The data file used was JEF\DCE50.DAT as previously in JEF\DCE50.MDL. The program uses JEF\DCE50.DAT as did the previous program JEF\DCE50.MDL.

In developing JEF\DCE52.MDL to JEF\DCE54.MDL (see Appendix 4.2.1.) the following revisions were made:

- New counters and logic was added so that the "No. of batches =" and "Time Taken =" display a value on screen.
- A report file was created so that the machine concerned and the time the job left the machine are recorded (Appendix 4.2.3.).
- A close file; CF(DAT) was included but not needed, as opening a new file closed th previous file.
- Include statements I=(A,MIF) and I=(B,MIF) were included but not required. This statement adds new files into the current file, making the size of each file more manageable, or side stepping the 64K Byte file size limit.

These changes are show in the Appendix 4.2.1. complete print out of the file. By comparing (Appendices 4.2.3. and 4.2.2.) the utilisation statistics. The times and production rates are shown to be the same. This is because none of the movement logic or set times were changed. The program uses JEF\DCE52.OLY for the overlay, as did JEF\DCE52.MDL. The program uses JEF\DCE50.DAT as did the previous two programs.



FLOW CHART 4.2
JEF\DCE 54.MDL

4.3 RESULTS OF DEVELOPMENT OF THE MODEL

The following section gives the results obtained from the Lotus file during the development of the JEF file PC Model computer simulation models were being developed and used. The results have been presented separately for clarity.

4.3.1. RESULTS FOR 123 LOTUS DATA BASE

On the 14th August 1989, the following breakdown of the Lotus file was undertaken:

Route in Simulation	Total No of each Component	No. of different Components	No. of different Families
1	7674	453	75
2	7507	410	79
3	1155	31	6
4	1365	84	15
5	2	1	1
6	276	27	3
7	-	-	-
8	16675	67	11
TOTALS	34654	1073	190

As the 18th August 1989 the following analysis of the laser tapes and gauges of material were undertaken. The tapes falling into effectively 9 routes.

There are 593 laser tapes, of these, 367 use high gauge material and 226 use low gauge material.
On the 23rd August 1989 the following analysis of the gauges of material in the Lotus file was undertaken.

High Gauge (71.6mm)

Total number of lines = 1451

- Lines with laser operations = 367 (+4 with no punch
  Lines with punch operations = 1184 operations)
- Lines with NO operations = 22 (+41 red lines)

Low Gauge (1.6mm)

Total number of lines = 1195

- Lines with laser operations = 226 (None not
- Lines with punch operations = 873 punched)
- Lines with NO operations = 312 (+10 red lines)

Complicated (all 1.6mm)

These have also been included in the low gauge section.

Total number of lines = 44.

- Lines with laser operations = 23
- Lines with punch operations = 44
- Lines with NO operations = 0

N.B. - Red lines (Tapes) obsolete.

The total number of lines in the Lotus 123 file is 2645. The total number of different components is 1671, ie. 1.6 lines each component. On the 5th September 1989, the following times were obtained for lasered parts.

High Gauge	- Behrens	35.1 hrs	Hammerle	15.03 hrs	
Low Gauge	- Behrens	26.85 hrs	Hammerle	36.14 hrs	
Allowed Hrs	Total	61.95 hrs	Total	51.17 hrs	
Actual Hrs	Total	37 hrs		31 hrs	

These are times for components that have laser tapes, so that one operation of production is via a laser Behrens, normally the route taken. With the simulation running at 159 hrs actual hours. The above results represent one lines working for 23% of that time. To this time has been added that of setting (judgement times), which as the component will be out of step will add considerably to the total time.

4.3.2. RESULTS FOR PC MODEL

There was only one file run through the JEF\DCE series, which was SCHED3.DAT. Due to the continuing development of computer simulation model. The settings for the following simulation were:

```
L=(10)                  ;Working in progress limit = 10
C=(0)                   ;Clock accuracy = seconds
%Move=(0:00:10.0)       ;Movement = 10 seconds per move
%Set_Same=(0:06:00.00)  ;Setting time 6 minutes
%Set_Dif=(0:18:00.00)   ;Setting time 18 minutes
```

The program used was JEF\DCE54.MDL. There were no cutting times allowed. The SCHED3 DAT file was the same as the files used in the DCE\DCE.MDL series of programs.

With 1080 batches of work the time for completion of the model run was 159 hours. This was compared with the total run times for th JDCE.DAT series of files (see Graph 4.3.1). As can be seen the completion time is substantially less for SCHED3.DAT than any of the JDCE.DAT files. There was a difference in completion time of 39% between the SCHED3.DAT file and the worst JDCE.DAT file. This was because the SCHED3.DAT file is constantly at minimum usage, all the machines are being continually loaded for the data file.

By transferring the utilisations of the three Behrens machines to Lotus 123 (results for the SCHED3.DAT file) graphs were plotted, and these are shown in Graphs 4.3.2\3\4. Whilst the laser and 1.5mm Behrens are active throughout the model run. The 2mm Behrens is not used for 70 hours of the run time, 40%. Showing room for improvement. The utilisation of all the machines was expected to be relatively flat, not varying dramatically as shown in the graphs. This factor depends on the supply of jobs and the length of time that the jobs are being operated on. For the low utilisation hours jobs may be waiting in the buffer, allowing a setting time for the machine.

This brings to a close the development of the simultaneous reading of dat files it is used along with sequential reading of the date files in Chapter 5 in which the model is developed from a simple 1 machine to 3 machines to 4 machines, to take into account all the machines in that section of the shop floor, and hence termed the complex model.

C H A P T E R   5

FURTHER DEVELOPMENT OF THE MODEL

5.0 INTRODUCTION

With both the sequential and simultaneous methods of data input to the model achieved, all that could be done to bring the model as close to reality as possible was to expand the model to include all those machines that were involved with the sheet metal process. The expanding of the model takes place in this chapter.

First is the final review of the Lotus data base, followed by the development of the overlay (picture) of the process, the development of the programs to control the movement of the work on screen.

The results obtained from those programs are attached after the program, as in previous chapters 3 and 4.

5.1 REVIEW OF THE LOTUS 123 DATA BASE

Continuing from the review of the Lotus file (Section 4.1.) the judgement (setting) times were investigated by Work Study. The originals:

0.10 hrs to change from similar job to job, and
0.30 hrs to change from unlike job to job

being approximate estimates. The investigation revealed the following revised judgement times:

To change from similar job to job - ie. more work in and out, booking etc., with no tool changes.

Behrens and Hammerle 0.10 hours per occasion.

To change tools, having pre-sorted jobs!

Behrens 0.35 hours per occasion	Hammerle 0.47 hours per occasion

These values have been used in all programs in chapter 5 (JON\ series of programs). It was also noted that the time taken to move a job from one machine to the next varied greatly.

Best time - 5 minutes
Worst time - 2 days
The average time - 2 hours

This was investigated more fully in Chapter 5, Work Study.

With the PC Model computer simulation being changed, there was a requirement to amend the Lotus 123 file to represent the following 3 manufacturing lines:

- Punch\Fold - General components up to an including 1.6mm.
- Punch\Fold - General components over 1.6mm.
- Laser - Components with specific laser requirements.
  - All stainless steel components.

Notes

- There are specific routings to cater for the more specialised production of some components which have to be incorporated in the above routes. (eg. Seal Frames, Hoppers, DLMV Weather Cowls and UMA Back Panels).

- The typical week quantity is based on a full week's program for each product type, which may not be the same production week.

- The 'Family' groupings are known to contain some anomalies that affect the gauge principle adopted to minimize setting.

- 37 components that were to be sub-contracted have been removed from the Lotus file.

The only other changes that were made to the Lotus file were to add components and their data as and when required to obtain accurate modelling of the actual Salvagnini's weeks work.

5.2 DEVELOPMENT OF THE MORE COMPLETE FACTORY MODEL OVERLAY

Once a quantity of data had been run through the DCE and JEF series of programs it became apparent that what actually happened on the shop floor was not being truly represented.

Development started on the overlay with DCE\DCE2.OLY (Plate 16). This shows all the machinery used, although no high\low gauge definition is given for the O.P. Press. It was a development from DCE\DCE22.OLY, with loops to ensure the largest possible queueing room. The routes between the Behrens and Hammerle are circular as the jobs cannot pass over one another. This overlay was rejected due to the roundabout between the Behrens and the Hammerle and the lack of direct flow through the model, production lines namely the nibbler\mitre machines and the O.P. Press.

The next development was DCE\JCDCE.OLY (Plate 17). This is similar to the original overlay but with two added Hammerles. This model was discontinued as it did not represent all the machines and routes available.

DCE\JCDCE1.OLY (Plate 18) was developed from DCE\JCDCE.OLY. This showed all the machines with a production line concept. There were three additional start points after the Behrens. This would lead to confusion. Also at this time the idea of having 2 images representing one machine which was used on both the high gauge and low gauge lines was raised. This was discounted as the queueing function would be incorrect, as would the machine utilisation. As in practice it is impossible to use the machine on two different jobs at the same time, this being modelled by the simulation. This overlay was discounted as it possessed to many start points.

DCE\JCDCE2.OLY (Plate 19) was developed from DCE\JCDCE1.OLY. There are no secondary start points in this although all the routes are now shown. This model was agreed as the basic outline. With the addition of all the necessary routes required and machines (N.B.H7 to H2 link not shown).

The model manages to convey the basic production line principle required. DCE\JCDCE6.OLY (Plate 20) is the completed model with the mitre\nibbler included and all the additional routings incorporated. This is the overlay that is used in all the JON directory programs. Effectively modelling all the possible routes that jobs can undertake.

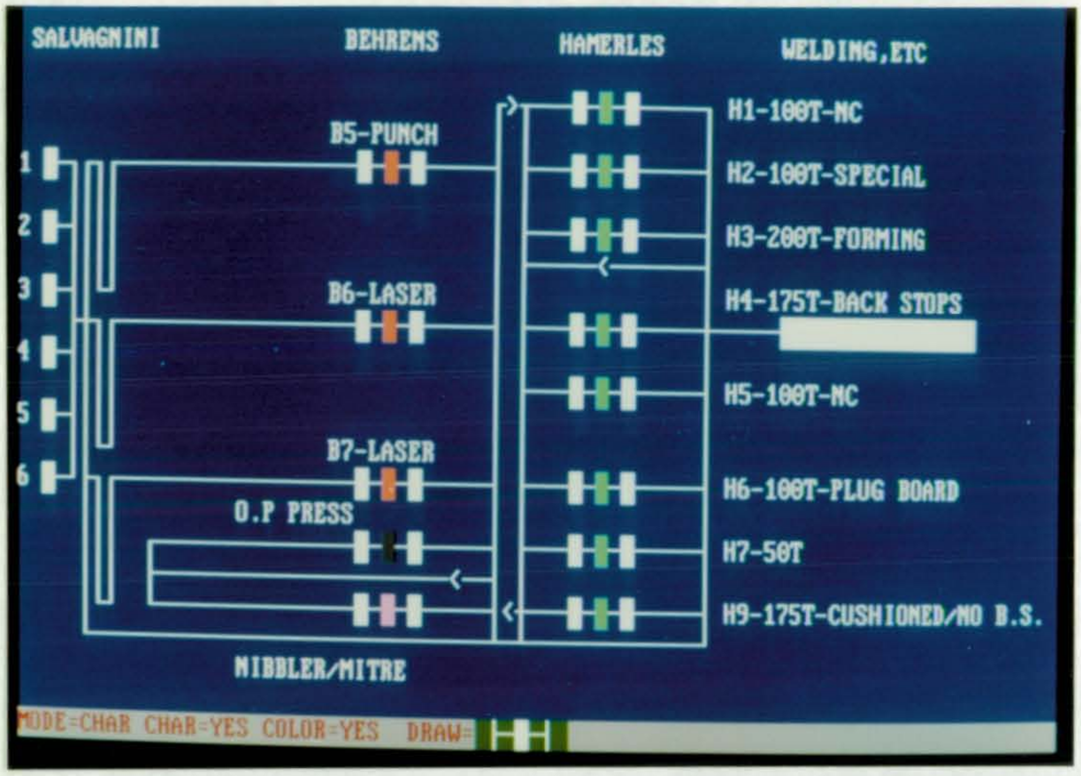


PLATE 16 - (APPENDIX 5.1.1) OVERLAY DCE\DCE2.OLY

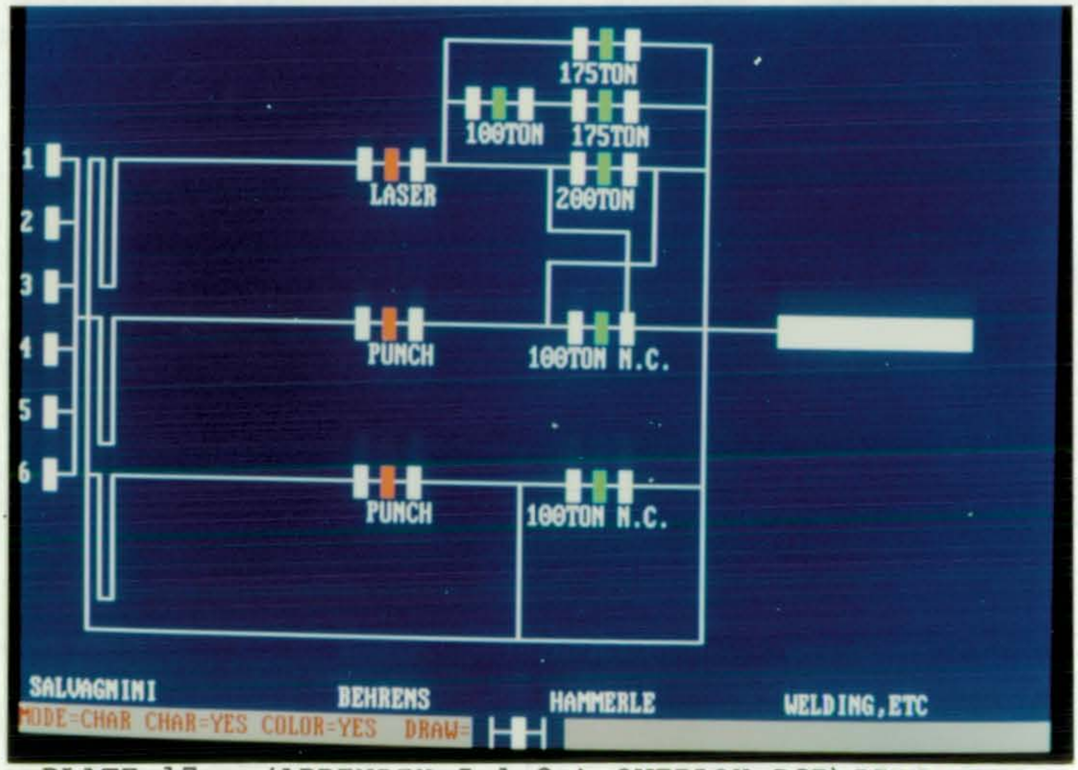


PLATE 17 - (APPENDIX 5.1.2.) OVERLAY DCE\JCDCE.OLY

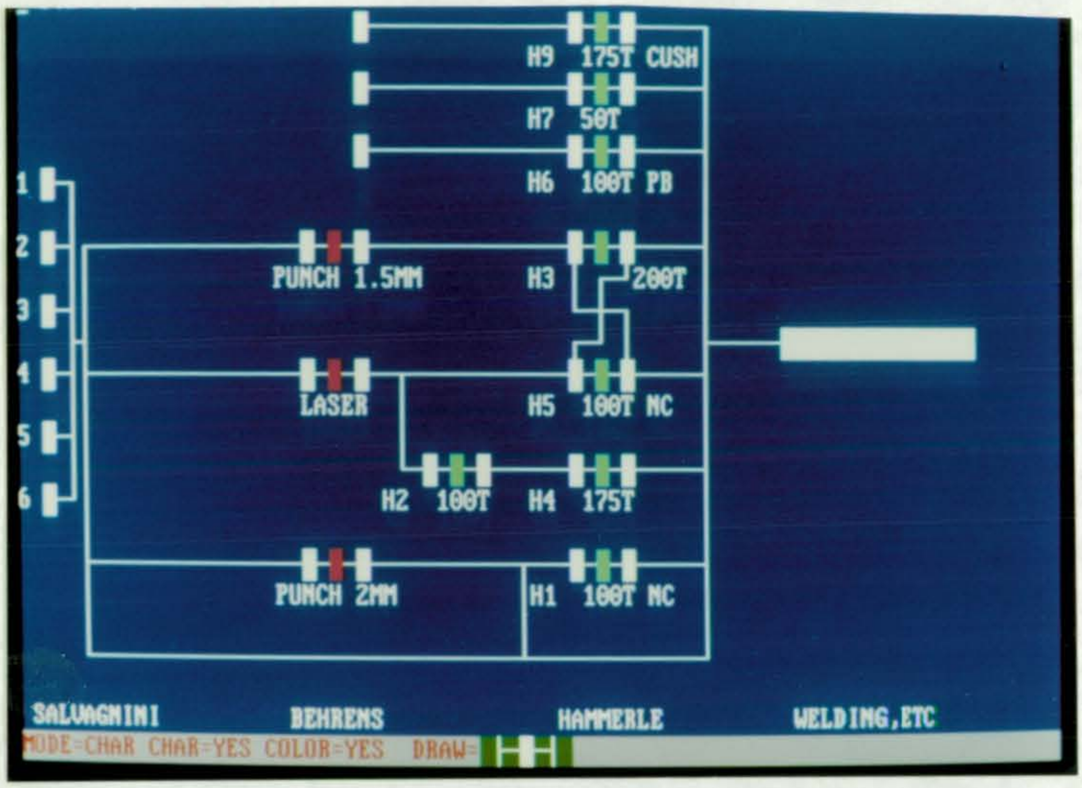


PLATE 18 - (APPENDIX 5.1.3.) OVERLAY DCE\JCDCE1.OLY

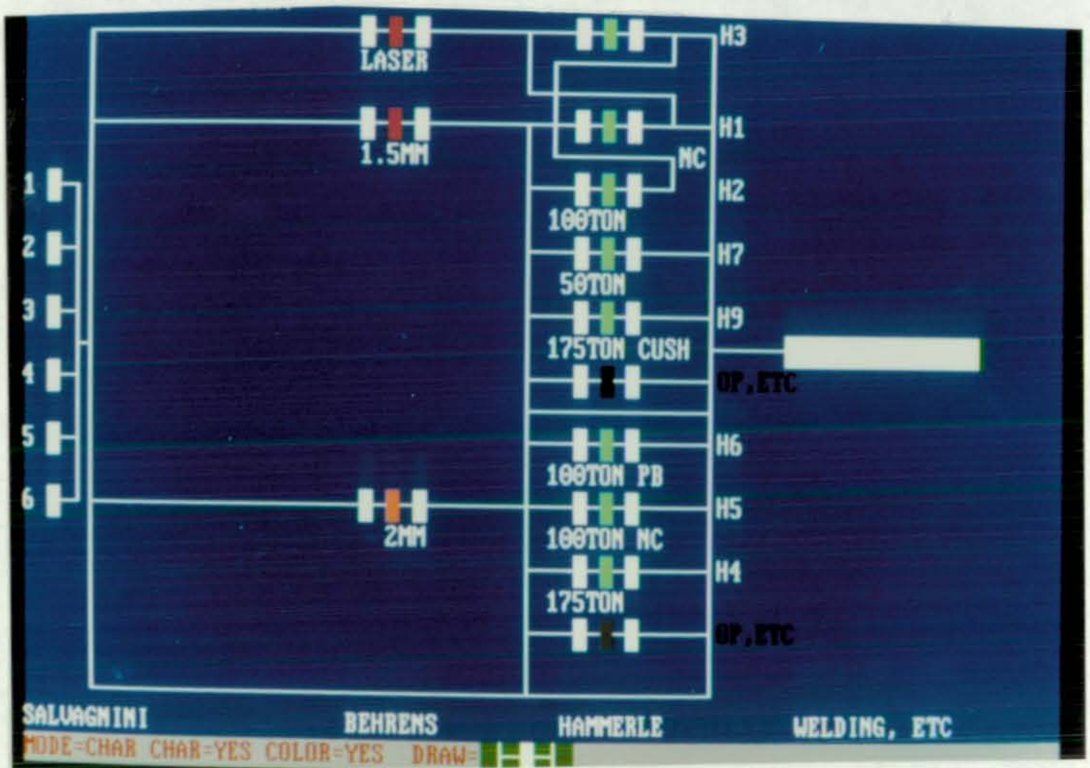


PLATE 19 - (APPENDIX 5.1.4.) OVERLAY DCE\JCDCE2.OLY

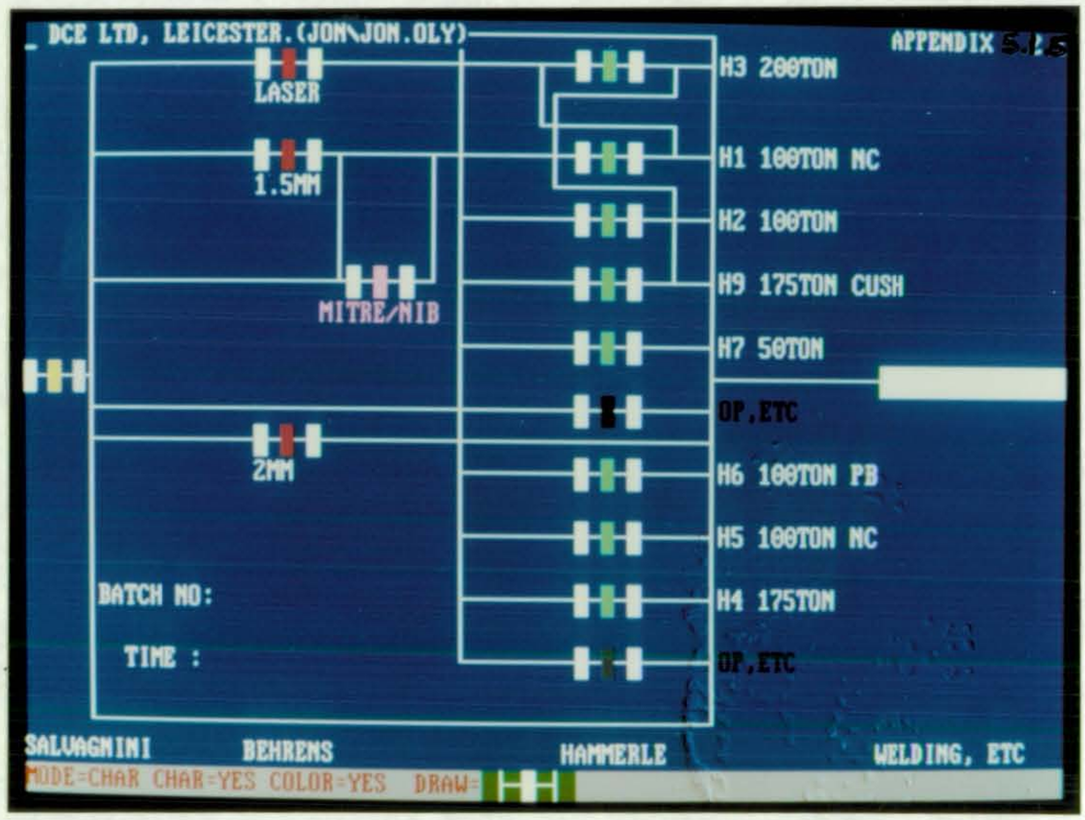


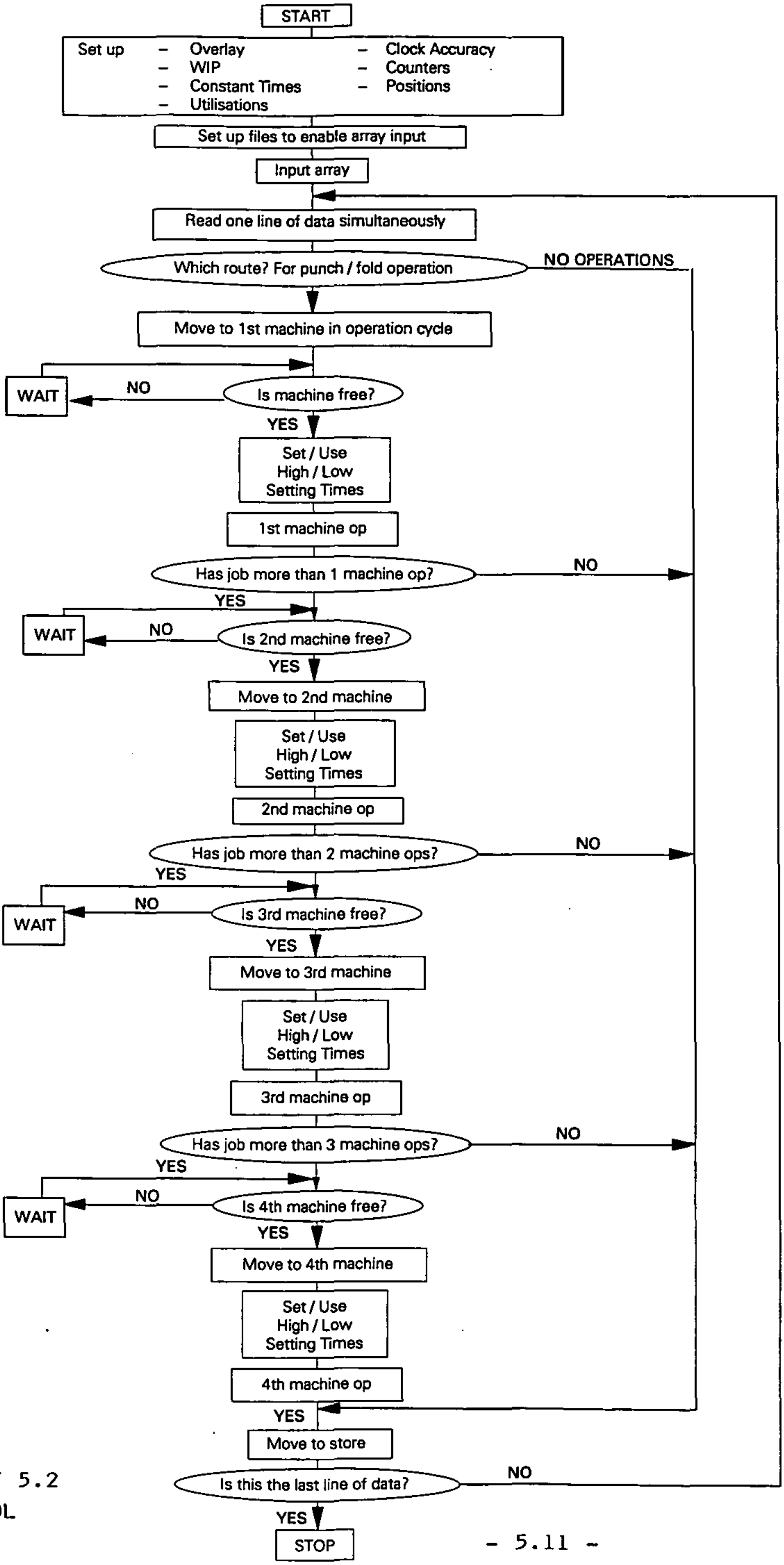
PLATE 20 - (APPENDIX 5.1.5.) OVERLAY DCE\JCDCE6.OLY
AS USED FOR THE JON DIRECTORY PROGRAMS.

5.3 DEVELOPMENT OF THE MORE COMPLETE FACTORY MODEL

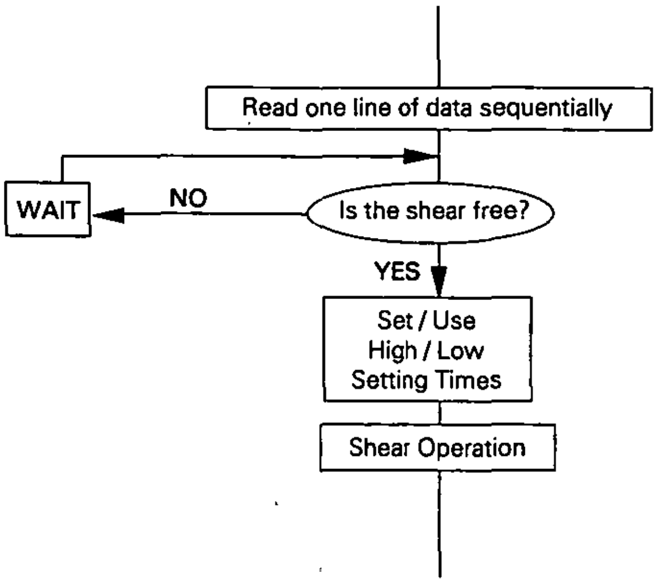
Once the JEF\DCE50.MDL series of programs had been developed it finally became apparent that with the limited number of machines (7off), the true complexities and interaction of job routes on the shop floor could not be successfully represented. So the JON \JON MDL series of programs were developed. These programs all use the basic MDL file with detail job movement changes, to represent the different methods of job transfer available.

The files JON\JON.MDL and JON\SOC.MDL read the data file simultaneously, the rest JON\DAVE.MDL, JON\KANBAN.MDL read the data file into the program sequentially, thus enabling more precise scheduling to take place. The overlay used is shown in Plate 20.

The flow chart for JON\JON.MDL the major program in this section can be found overleaf (Flow Chart 5.2). Shown also in this flow chart are where the changes necessary to obtain the logic for JON\DAVE.MDL and JON\KANBAN.MDL should be inserted. These additions can be found in Flow Chart 5.3. The routes used for these programmes are shown in Appendix 5.1. As can be seen there are far more than previously encountered, (Appendix 3.2).



FLOW CHART 5.2
JON\JON.MDL



FLOW CHART 5.3

JON\DAVE.MDL AND JON\KANBAN.MDL ADDITIONS

In developing JON\JON.MDL many sections of different programs were brought together. The simultaneous reading of the data file obtained from the JEF\DCE50.MDL series of programs. The logic for controlling the jobs movement through the machines was obtained from the DCE\FINAL.MDL series of programs. These were added to enabling true representation on the new overlay. The program JON\JON.MDL is included in its entirety in Appendix 5.2.1. with the description of the logic inserted as comments on the right hand side of the sheet.

The data file has had to be changed and is longer than before as all the 32 routes are taken into account. An extra column has been included in the file to enable the accurate routing of jobs through the simulation.

In the utilisation statistics the jobs are completed in just under 10 hours. The initial jobs that sets all the counters and controls is also included. Not all the utilisation figures from the machines have been included as they would not fit on the paper and could also be confusing. The report array shows the times the jobs pass through the relevant machines. The JON\JON .RPT file shows the sequential times in the order that they occur so that the order that machining usage can be seen. These again are the actual times from the start of the job entering the simulation.

In developing JON\JON.MDL to JON\SOC.MDL the modifications made to the program were to the job moves from the Laser Behrens to the approach of the second machine. This was originally an incremental movement, it is now an absolute movement. The basic change in logic is shown in Appendix 5.3.1.

The same overlay and data files are used as in JON\JON.MDL.

In the utilisation statistics are similar to the JON\JON.MDL, completing working in hour 10, but due to the logic changes there were some small deviations in individual figures. The report array and JON\SOC.RPT files also follow this trend. This is due to the data file only being a "proving" file so that the various routes will not become congested giving true results.

In developing JON\JON.MDL to JON\DAVE.MDL this method of loading data into the simulation was changed from simultaneous, to sequential reading off the data file. The Salvagnini has also been included enabling a cutting time to be introduced. Resulting in a more realistic simulation. The effects of different loadings on the shop floor could be seen. The data file had to be changed. One of the routings columns was deleted, the RTF column.

In the utilisation statistics, the time to complete the jobs is now 12 hours. This is longer than before and would be expected as a new operation is included that every job has to pass through. The array display and JON\DAVE.RPT files are similar to those before as the logic is effectively the same. The major difference between JON\DAVE.MDL and JON\SOC.MDL is that the data is read into the program sequentially and specific scheduling can take place.

In developing JON\DAVE.MDL to JON\KANBAN.MDL the logic controlling job transfer between the Behrens machines and the Hammerle machines. The Salvagnini shear has not been included due to the randomness of its completion of parts production. The change in logic is shown in Appendix 5.4.1.. The same data and overlays have been used as in JON\DAVE.MDL.

The utilisation statistics show that the jobs are completed in just under 12 hours. The results for this programme are again similar to JON\DAVE.MDL's. This is due to the data file being the same with only small changes in the logic.

5.4 RESULTS OF FURTHER DEVELOPMENT OF THE MODEL

The following section gives the results obtained from PC Model during the JON programs use. The results from the same set of typical week data show that there was a discrepancy between the Lotus times and the machining times in PC Model. The first simulation run with the reviewed Lotus file gave the following results:

No setting time\No movement time = 61 hrs

Setting 0.1 and 0.3 hrs\Movement 5 seconds = 150 hrs

Setting 0.1 hr for all\Movement 5 seconds = 83 hrs.

The work in progress limit was set at 25 throughout. There are 186 families processed and the number of jobs exceeds 1100. The situation results were then transferred to Lotus. The following times recorded.

	Setting	Machining
Laser Behrens	66.65 hrs	35.83 hrs
1.5mm Behrens	33.77 hrs	6.59 hrs
2.0mm Behrens	34.94 hrs	12.80 hrs
	135.36 hrs	55.22 hrs

Now comparing the tables with the Lotus file.

Lotus Setting 138.43 hrs Machining 131.61 hrs.

It was clear that the simulation machining time was low. It was concluded that not all the data file was being read by the model and the buffers were working all the time a job was waiting. Jobs with zero quantities were also being read into the simulation. The following actions were taken:

- The data file was checked for errors. By running the simulation. Entering the array file and checking for zero lines.

- The logic of the buffers was changes from JC(*H22,:CONT25) to TP(2,*H22,*H23) to ensure that setting could not start before the previous job had finished.

- The data file had all zero quantity jobs removed.

With these changes implemented the simulation was run again. The results:

Simulation run time 131 hrs. All settings as for the previous run.

	Setting	Machining
Laser Behrens	84.42 hrs	59.46 hrs
1.5mm Behrens	51.50 hrs	25.69 hrs
2.0mm Behrens	73.30 hrs	44.20 hrs
	209.22 hrs	129.35 hrs

These results are 34% and 2% respectively astray from the Lotus file results. At this time, the setting was considered to be out due to constantly changing from one family to the next and back again incurring a heavy time penalty. The 2% machining time was put down to file errors. Now examining the Hammerle times (Brake Press)

(Hammerle)	H3 -	37.14	hours
	H1 -	78.73	
	H2 -	13.20	
	H9 -	3.52	
	H7 -	3.81	
	H6 -	5.12	
	H5 -	41.74	
	H4 -	10.39	
		193.65	hours (Actual)

The Lotus file gave:

Total Machining 176.58 hours Setting 117.60 hours Actual

Implying that the model has 17 hours (9%) to many hours machining. The logic at the start of each machining sequence was changed so that there was not time delay in transporting the job from one part of the machine to the next. A new column was inserted into the array. That gave the jobs a second route number. This was used to sort the jobs for the first operation (normally the Behrens). The file was run again:

```
The total time being = 187 hours
With work in progress limit = 30
Setting times were changed to Set_Same = 18 mins.
                              Set_Diff =  6 mins.
```

The setting times were an error! The times were as follows:

```
Behrens machining = 141.12 hours
        setting   = 231.40 hours
Hammerle machining = 192.48 hours
         setting   =  85.04 hours
```

The setting times were expected to be out, but both the machining times were greater than those in the Lotus file.

The PC Model program was re-examined, errors were found in the set value statements.

```
                      SV(@LASTPUNI,OBJ@3)

Had the wrong                              This value 3 was
file name                                  occasionally 2
occasionally
```

After these changes had been made the program was run again, with the setting times corrected to 0.1 and 0.3 hours. The work in progress limit set at 32. The results being:

```
Behrens  machining = 173.45 hours
         setting   = 137.88 hours

Hammerle machining = 127.58 hours
         setting   = 137.20 hours
```

The total run time being 187 hours. The setting time for the Behrens was correct, but the rest of the times were out! Even though each route had been checked by running a test data file through it after being constructed. Now each route was again checked individually. Individually they work correctly, but when more than one route is added some times are lengthened, even though no errors can be found. Next zero travelling and zero setting times were tried. The results were.

	PC Model	Lotus	Error
Behrens	122.27 hrs	131.61 hrs	9 hrs
Hammerle	127.59 hrs	176.58 hrs	49 hrs

Upon reviewing the file the position of a set value statement was changed. This had the effect of stopping the screen from flashing the change is shown below, carried out whenever it occurred.

```
:Cont74                              ;Label
SV(@LASTBND7,OBJ@3)                  ;Set value
MA(*H73,0)                           ;Move Absolute
SV(%BEND7,%%SCHED(#BND,OBJ@2)) ;Set Value
AO(%BEND7,*,@@SCHED(#SQTY,OBJ@1)) ;Arithmetic
                                       operation.
WT(%BEND7)                            ;Wait
```

The program was again run using W.I.P. limit of 32. Only the Behrens 1.5mm machine was out, by 0.06 hours. Putting the Lotus file through simulation gave the following results:

Behrens Machining	122.27 hours
Behrens Setting	137.88 hours
Hammerle Machining	127.58 hours
Hammerle Setting	144.20 hours

The machining hours remained constant with the previous results, still a large error. The only thing now left unchecked was the errors incurred changing the Lotus file and then transferring it to PC Model. The shock was that 20 hours of machining time were deleted in the change and transfer for Hammerle.

Upon reconstructing the Lotus file ready to be transferred to PC Model, 8 hours were lost on the Behrens machining times, and 23 hours were lost on the Hammerle machining times. This occurred when using Lotus 123 version 3. Transferring data to a new sheet and multiplying the allowed hours to get seconds actual time.

Deleting the zero lines loses no time. The way that the simulation model has been constructed multiplies the 2 and 3 Hammerle operation times by 2 and 3, so that their time has to be reduced by this factor to obtain the correct time. The file was then run again, and the following results were obtained:

Behrens Machining	122	132	8	2	hrs
Behrens Setting	128	139	-	11	hrs
Hammerle Machining	151	177	23	3	hrs
Hammerle Setting	143	118	-	25	hrs

The Behrens machining error unaccounted for is 1.5%. The Hammerle machining error unaccounted for is 1.7%. An acceptable percentage error. Further investigations into how the Lotus file lost these times were undertaken. It was concluded that the hours were being lost by the Lotus Package rounding figures up\down when undertaking the arithmetical operations, although this could not be proved. The setting time differentials were put down to two causes:

- For the Behrens value being less than the Lotus file value, items that travelled down other routes, mitre and no machining operations had been included in the Lotus value.

- For the Hammerle value being greater than the Lotus file value, items intermixed before entering the machine, giving a maximum setting value. It only taking on an additional 84 maximum settings to obtain the 25 hour differential, in a file in excess of 100 jobs, very easy!

With the data and programming files accuracy validated, the number of jobs passing through each route needed to be assessed. This was achieved by adding a counter on screen during the simulation. The results were as follows:

	28	jobs, No 2nd operation
263 jobs Laser Behrens	11	jobs, H3
	283	jobs, H1 (N.C.)
304 jobs, 1.5mm Behrens	50	jobs, H2
	34	jobs, H9
3 jobs - 20 jobs mitre\nibbler	28	jobs, H7
	61	jobs, OP 1.5mm.
31 jobs, No 1st operation	56	jobs, No 2nd operation
	50	jobs, H6
355 jobs, 2mm Behrens	297	jobs, H5 (N.C.)
	79	jobs, H4
4 jobs, No operations	21	jobs, OP 2mm.

These numbers are reflected in the Lotus file. At this time for further identification the colours of the job were changed as soon as the first operation had been completed. With the program validated, 3 files were run through the simulation.

File 1 - All stainless steel and families going through laser route, with complex parts.

File 2 - Only stainless steel and lasered work through the laser Behrens.

File 3 - Only stainless steel and lasered work through the laser Behrens, but all work going through the simulation in family order.

The work in progress limit was set at 10 jobs throughout. The setting times used were those recommended by the Work Study Department. The same data file was used throughout with only the routes changed as and when necessary. From the results it can be seen that changing the routings of the material through the laser and changing the order of the material flow has an effect on the total run time of the program. This shows up the bottle-neck between the laser Behrens and the Hammerle machines.

To try and overcome this the JON\SOC.MDL file was created. In this file the movement of jobs from the laser Behrens to the Hammerle is absolute with no delay. With a work in progress limit of 10, there is no change in the overall time (198 hours) increasing the W.I.P limit to 1500 gives a saving of 10 hours (SOC-1B). But when applied to the original file the time drops to 168 hours (FILE-3B). These results are shown sectionally on Graph 5.3.1. The next investigation carried out was 5 weeks actual program of work. These results are shown below:

1)	Week 51 1989	Family 32 hours As MCD 25 hours KANBAN 30 hours	57% of items on program are in data file.
2)	Week 01 1990	Family 38 hours As MCD 36 hours KANBAN 38 hours	46% of items on program are in data file.
3)	Week 05 1990	Family 54 hours As MCD 42 hours KANBAN 41 hours	44% of items on program are in data file.
4)	Week 06 1990	Family 59 hours As MCD 69 hours KANBAN 78 hours	52% of items on program are in data file.
5)	Week 07 1990	Family 28 hours As MCD 35 hours KANBAN 41 hours	54% of items on program are in data file.

The totals of weeks 1\2\3\4\5

Family 182 hours	37 hrs/week
As MCD 180 hours	36 hrs/week
KANBAN 212 hours	43 hrs/week

The typical week

Family 216 hours
As MCD 216 hours
KANBAN 243 hours

All settings remained constant throughout the simulation runs. The method of programming was:

- Family, in job order as on the typical week data file.

- As MCD, in job order as processed by the Salvagnini Shear.

- KANBAN, in job order being pulled by the last machine, no W.I.P. between the Hammerle and Behrens machines.

The percentage of jobs appearing in the data file have been shown to the right of the listing. The results are also shown in Graphs 5.3.2. and 5.3.3. The graphs show that there is not a great difference in the overall completion time between the different processing methods (max. 24% week 06).

Even though the files are the same per method for each week, effectively only 50% of the jobs appear on the data file. The "KANBAN" and "As MCD" files have the Salvagnini Shearing times included, implying that the family data would take that much longer to process, bringing down the difference to 24%. From these figures to optimum method would appear to be processing the jobs as the Salvagnini produces them, balancing the shearing programme by using high\low\high gauge of materials and then the numbers of different components.

Checking each component individually to balance the setting time would be unproductive as the Salvagnini produces the parts in a random order and the time involved would be prohibitive - Approx 3095 parts x 3 minutes each implies 15.25 hours. Scheduling each week - 2 days (8 hour day)

The family system has a high work in progress, as a whole weeks work is stored behind the Salvagnini and then the system\process are flooded whilst the Salvagnini generates work for the next week. In real life the factory if now working to a very strict regime would soon become clogged with W.I.P. and the financial cost would be prohibitive.

The KANBAN method is a "pull" system not "push" as in family system. The difference in philosophy is that a push system puts all the work at the start and then pushes it along the system. A pull system starts at the end of a machine is free goes to the next up the line, and pulls the work down, incurring a lot less work in progress (see Diagram 5.3.1.).

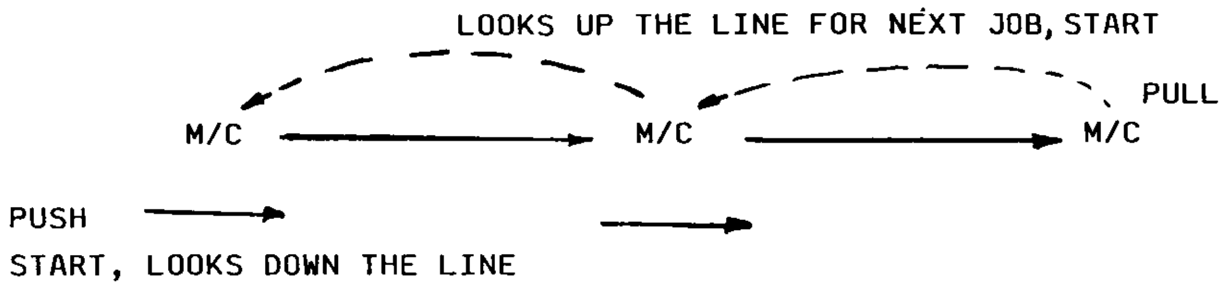


DIAGRAM 5.3.1. SHOWING THE DIFFERENCE BETWEEN A PULL AND PUSH SYSTEM

The KANBAN system has a W.I.P. build-up at the start of the system, after the Salvagnini Shear. From there is it regulated at one job at a time basis, so that there is a minimum of W.I.P. throughout the rest of the system. This system works at the speed of the slowest process or longest machining time.

The AS MCD system accepts a small build up of W.I.P., but tries to process it as fast as possible, by balancing the routes using the principle "get it in, get it processed, get it out". In 5 out of the 7 data files examined, it was the quickest method for the application it also seemed to be the most logical.

The worrying factor in the results, was that if the longest time (78 hour week 06) is doubled it still falls considerably short of the 216 hours total run time for the typical week. During the course of the project batch quantities and work held in stores have been reduced so that manufacturing is now primarily biased towards on line shortages. Following the difference in results obtained above, the data file was completed for week 11, 1990 to tie in with the results obtained from the Work Study investigation, enabling a direct comparison to be made. Two different low setting times were used, 6 minutes and 12 minutes the high setting times remained as recommended by Work Study.

Total run time = (6 min) 98 hours (12 min) 106 hours

AS MCD Times on machine

Salvagnini	=	(6 min)	68 hours	(12 min)	69 hours
Behrens	=	"	72 hours	"	78 hours
Hammerle	=	"	98 hours	"	106 hours

Family Method

Total run time = (6 min) 87 hours (12 min) 94 hours

The 12 minute small setting time was recommended by the Section Leaders to take into account all velocities as the new parts added to the data file were not put into families as "product knowledge" and time did not allow.

The production rate for the simulation shown against the actual production rate for week 11 can be seen in graph 6. The family time is not truly a reflection of what would happen on the shop floor as no Salvagnini times are included, this would minimise the 11% advantage in completion time. Two more weeks were processed having added all the jobs required to the data file.

Week 12, 1990 12 minute start setting time.

Salvagnini	97 hours
Behrens	114 hours
Hammerle	120 hours
AS MCD Time	120 hours
Family Time	108 hours

Again the family time is quicker as the items are not correctly placed into families, but placed by gauge and then by what family is available effectively sorting the Salvagnini shearing program. This weeks work had an excessive amount of high gauge material, but still only took 12% longer in total run time to complete the program. With improved program logic, it would have been possible to process work down a different route to minimise the overall run time. The spare time on machines might be required for maintenance and the rules of the simulation and the real life control would be changed and made more complex. The final week to be programmed was:

Week 49, 1989 12 minutes short setting time.

Salvagnini	71 hours
Behrens	77 hours
Hammerle	93 hours
AS MCD Time	93 hours
Family Time	86 hours

Between these lost three completion times there is a variation of 24% with an average completion time of 106 hours. The family times are never more than 13 hours less than the As Machined method which also includes cutting the steel and not incurring W.I.P. storage\handling\purchase costs.

N.B. All other settings remained constant throughout the preceding simulation runs, so that the results can be accurately compared.

As mentioned earlier in this chapter, a direct comparison was made between work for one week on the computer model, and the same work on the factory floor. What was found, the improvements to the factory recommended, and the method study evaluation can be found in the next chapter, 6, Work Study.

CHAPTER 6

WORK STUDY EXAMINATION

6.0 INTRODUCTION

With the computer simulation model complete and generating results, the validity of these results were brought into question as there was no actual data available from the shop floor to measure the results against. Another requirement was stated, that assuming the PC Model simulation was better than was actually happening on the shop floor, what improvements could be made to the shop floor to bring them into line with the simulation. In effect, where was the production process falling down? To answer these questions an analysis of the front end of the shop floor was undertaken (Method Study).

At this time the company placed job packs onto the shop floor and allowed 4 weeks for the completion of the job. It was not known if this was achieved and if so, by what percentage of job packs passing through the shop floor. The 4 weeks being an historical value. To obtain the necessary comparative results, the following studies were undertaken:

- Factory layout
- Process charts
- Operation process chart
- Flow process chart
- String diagram
- Critical analysis
- Inter-relationships of machinery
- Job movement

6.1 CURRENT FACTORY LAYOUT

It was found that although drawings of the factory layout existed they were out of date. The first job was therefore to draw up the current factory floor plan. With the help of the Work Study Department, a tape measure, the old factory plan and basement plans were 'marked-up' and passed onto Production Engineering for modifications to be incorporated into the factory and basement plans on the CAD system. At the same time drawings of the available racking at the Salvagnini\Behrens\Hammerle sections (front end of the shop floor) were constructed with the utilisations shown on them.

This was done to obtain a judgement of what racking was actually required. From which it was hoped to remove some racking, this was achieved. The racking next to the "guillotine" was judged to be superfluous to requirements, and never used. This racking was disposed of during the time of the project. A layout drawing is shown in Appendix 1, photographs of the factory floor are shown in Plates 21, 22, 23 to give a sense of size and proportion.

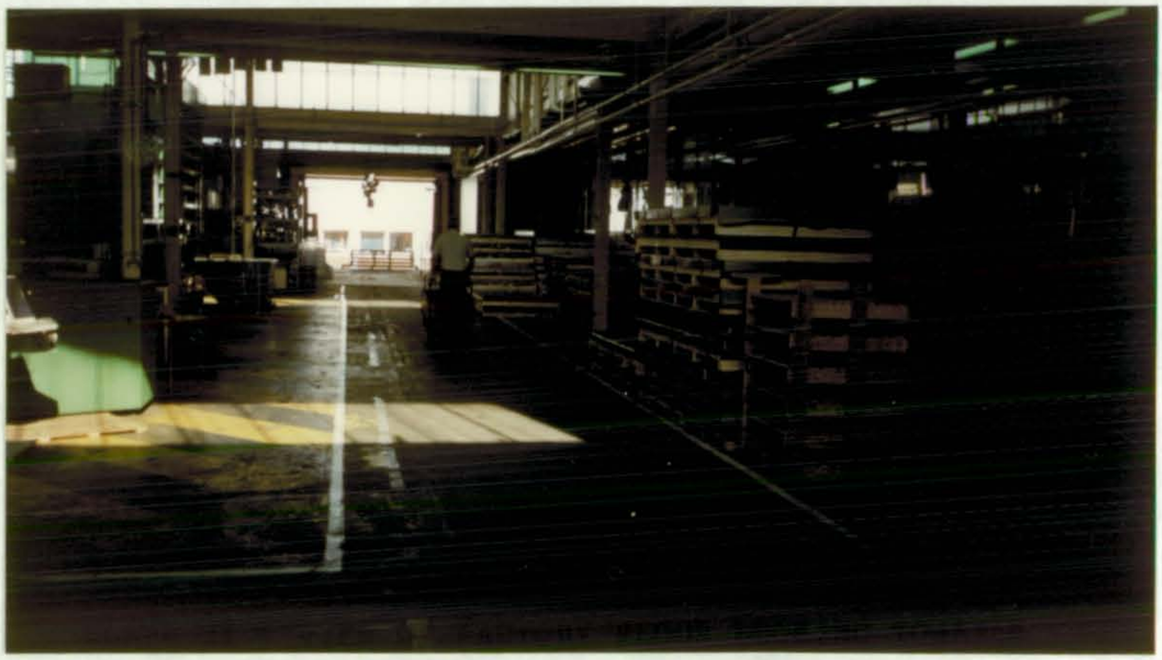


PLATE 21 - VIEW OF FACTORY FLOOR LOOKING TOWARDS LOADING AREA (SHEAR ON LEFT OF PICTURE).

PLATE 22 - VIEW ACROSS FACTORY FLOOR, TOWARDS THE FOLDING SECTION FROM THE PUNCH SECTION.

PLATE 23 - VIEW OF FACTORY FLOOR, LOOKING UP THE FOLDING SECTION FROM WELDING.

6.2 THE PROCESS CHARTS

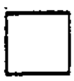
The first method of recording and analysing the work flow was the Process Chart. Using the International Labour Offices Definition:

"A process chart is a graphic representation of the sequence of events or steps that occur in the work method or procedure, classifying them by symbols according to the nature of the event. It is a device for visualising a procedure for the purpose of improving it".

The symbols used are as follows:

 - Operation

This takes place at each machine when the job is 'operated' on. Also whenever the job has its characteristics intentionally changed, eg. prepared for transportation.

 - Inspection

This takes place when the job is checked for identification, quantity or quality.

 - Transport

This takes place when the job is moved from one place to another.

 - Delay

This takes place then the job is stationary awaiting the next operation to take place, when not actually in storage.

 - Storage

This takes place when a job has to be signed for from an official area. It cannot just be picked up and moved.

These symbols are also known as ASME code (The American Society of Mechanical Engineers) for the description of processes.

6.2.1 THE OPERATION PROCESS CHART

The first process chart to be constructed was an 'Operational Process Chart' (Appendix 6.2.1.). This shows an overview of the front end of the factory floor. Drawn onto a factory floor layout for a clear reference and orientation. It immediately shows the number and complexity of movements that the jobs undergo.

At the bottom of this chart is the vertical movement that the 'standard job', that undergoes Cut\Punch\Fold operations is itemised and is tabulated below in Table 6.2.1. The vertical movement has been itemised to see what is currently occurring and to help highlight areas of improvement to smooth the flow of work through the factory. Moving work through the vertical plane achieves nothing other than wasting time. To minimise this, work should enter the process at one height, and remain at this height throughout. It was found that on the factory floor the total distance being 11.4 metres (up and down), assuming that the fork lift carries the job at a height of 0.6M above the ground, 16 different lifts being undertaken during this time.

6.2.2. THE FLOW PROCESS CHART

This chart examined the movement of the Behrens tapes, tools, (punch and die sets) and the special tooling for the Hammerle Brake Presses and Overhead Power Presses (O.P. presses) to see if the positioning of tools or tapes could be repositioned to obtain a faster change-over, reducing the distance travelled. The chart was again laid out on a factory floor plan for clarity and to give an idea of the distances involved in each movement, see appendix 6.2.2.

It can be seen that whilst the Behrens tooling has a relatively straightforward route, the special tooling for the Hammerle and O.P. Presses is long and tourcherous, travelling through the most congested part of the factory floor. The Salvagnini effectively blocking off the old and most direct route.

VERTICAL MOVEMENTS +VE METRES	VERTICAL MOVEMENTS -VE METRES	ACTION
0.6		Forklift from store
0.3		lift onto Salvagnini
	0.9	Output from Salvagnini
0.6		Forklift from Salvagnini
	0.6	Forklift to Store
0.6		Forklift from Store
	0.6	Forklift to Store
0.6		Forklift from Store
0.3		Forklift onto Behrens
	0.3	Forklift off Behrens
1.8		Forklift onto Rack
	1.8	Forklift off Rack
0.3		Forklift onto Hammerle
	0.9	Manual off Hammerle
0.6		Forklift moving job.
	0.6	Forklift to Welding\Paint Line etc.
TOTALS 5.7	5.7	16 Different Lifts

Total vertical distance moved 11.4 metres.

Allowing forklift to carry jobs at 0.6 metres off the ground.

Date of investigation - 06/02/90

JOB VERTICAL MOVEMENTS IN UNDERGOING CUT\PUNCH\FOLD OPERATIONS

TABLE 6.2.1

6.2.3. THE STRING DIAGRAM

Using the International Labour Offices definition (14).

"A string diagram is a scale diagram on which is plotted, usually by means of a continuous thread, movement within a given area and over a given period of time for the purpose of showing the frequency of movement between various points and of determining the distance covered".

The string diagram was constructed using different coloured thread to show the different jobs.

Black	-	Cut\Punch\Fold\Weld etc.
Green	-	Cut or Direct to O.P. press
White	-	On Line Shortage

The time for the investigation was 1 week. The percentages of work that flowed through each machine are shown below in Table 6.2.3. The diagram was laid out on a copy of the factory floor plan (Appendix 6.2.3). It can be seen immediately that 2 gangways are not used. The first between the Shear and the Rack is not wide enough for a forklift truck. The second next to the Punches at the other end of the drawing is not used as this leads into the welding area and when in welding is not wide enough for a forklift truck.

To the left of the 'Shear' looking down the factory, the gangway has a high utilisation with all three varieties of work moving through this area, 2 out of 3 in both directions. The high utilisation continues across the factory towards the Brake Presses and OP Press. Through this section and into the Welding Section. The punch section gangway is progressively less utilised the further down the factory the routings are examined (For Cut\Punch\Fold~1/5 mile). The diagram shows the large distances moved by jobs and the intermingling of the routes through the factory. Showing proportionally, by the number of strings used, just how many jobs pass one point in a week.

Behrens Loading

No. 7 (Laser)	30%	
No. 6 (Laser)	30%	
No. 5 (Non-Laser)	40%	of work

No. of jobs per shift per machine - 10 off

Hammerle Loading

No. 1 (100 Ton N.C.)	25%	
No. 2 (100 Ton)	15%	
No. 3 (200 Ton)	15%	
No. 4 (175 Ton)	10%	
No. 5 (100 Ton N.C.)	25%	
No. 6 (100 Ton Plug Board)	5%	
No. 7 (50 Ton)	2%	
No. 9 (175 Ton cushioned)	3%	of work

No. of jobs per shift per machine - 30 off for N.C.

Overhead Power Press (O.P.)

200 Ton	30%	
200 Ton	30%	
75 Ton	15%	
75 Ton	15%	
10 Ton	15%	of work

No. of jobs per shift per machine - 1 off.

Date of investigation - 03/01/90.

When implemented onto the string diagram, 1 string = 5% of work.

DATA USED TO CONSTRUCT STRING DIAGRAM Table 6.2.3.

6.3. CRITICAL ANALYSIS

This stage of the examination was carried out to justify why each of the operations was undertaken. It has been split into three sections.

- Cut\Punch\Fold
- O.P. Press
- On line shortages

The questions asked at each section were:
(Included in Appendix 6.3.2. for reference)

-Purpose	:	What is actually done?
	:	Why is the activity necessary at all?
-Place	:	Where is it being done?
	:	Why is it done at that particular place?
-Sequence	:	When is it done?
	:	Why is it done at that particular time?
-Person	:	Who is doing it?
	:	Could it be done better by someone else?
-Means	:	How is it being done?
	:	Is there any possibility of doing it more economically in some other way?

In answering the question"what is actually done?" for each section, the ASME code symbols were drawn in for each operation (machining, movement, storage, etc.). The number and type of these operations were then compared with the minimum number of operations required to complete the job, highlighting what improvements could be made. It was found that in the case of:

- Cut\Punch\Fold - 8 operations out of 24 needed to be deleted.
- O.P. Press - 2 operations out of 13 needed to be deleted.
- On Line Shortage - 8 operations out of 22 needed to be deleted.

The examination shows what is being done and why it is being done, compared with the ideal. As can be seen from the appendices, there are a lot of unnecessary operations taking place. So that combining new methods with the movement of machinery (if justified) an optimum method of production can be reached, with the minimum distance moved by the work.

6.3.1 INTER-RELATIONSHIPS OF MACHINERY

Through careful consultation with the Section Leaders of the various machines, the following relationships became apparent (Appendix 6.3.4), when discussing the idea of "production lines".

Of the 2 laser Behrens one would have to go onto the specialised low gauge line, the other would go onto the high gauge line, so minimising the cross over of work, not to mention the excessive change over times on machines due to the gauge change. The non-laser Behrens would punch all the rest of the low gauge material as there is more of this material used. Thus trying to balance the routings.

The Hammerle machines lining up with the respective Behrens being used for one gauge of material. There are two machines that it is possible to dispose of.

- -Hammerle No. 7 — 50 ton capacity
- -Hammerle No. 4 — 175 ton capacity

As the work done by these machines can be done on other machines. The links with the other sections are also noted; namely

- -The O.P. Press Section
- -The Paint Section
- -The Welding Section

6.4 INVESTIGATION INTO JOB MOVEMENT THROUGH THE FIRST 4 SECTIONS OF THE PRODUCTION FACILITY

With the Work Study showing the overall picture, the actual times for jobs to pass through the Production facility were required to help quantify the amount of delay time, the accuracy of the time allowed for production, and validate the computer simulation.

A card was added (Appendix 6.4.0.) to each job pack to be manufactured in week 11 (16th March 1990, start date). 305 jobs were to be manufactured in this week, of the cards 70% (214) were returned. The product breakdown to be produced that week is shown in Table 6.4.1. This is also compared with the product breakdown of the "Typical Week". The comparison is close, the typical week requiring 40 more units to be produced at a total of 195 units. Examining the data obtained from the cards:

The last job passed through the Salvagnini Shear on 5th April 1990, some 3 weeks from the start. The graph of the waiting times in front of the Shear shows the amount and distribution of waiting that takes place (Graph 6.4.1.). The graph of the waiting time between loading and cutting operations shows clearly the effect of "On line shortages" (urgent jobs) quantities of jobs varying between 0 and 23 in adjacent days. The on line shortage jobs taking precedent.

The last job passed through the Behrens Punch\Lasers on 11th April, 1990, some 4 weeks from the start. The graph of the waiting times in front of the Punches shows the amount and distribution of waiting that takes place (Graph 6.4.2). The graph of the waiting time between cut and punch operations shows the section leaders aim at punching the work as soon as it is cut, normally within 2 days. The majority done in this time, the vast majority of the rest was completed within the week.

The last job did not pass through either Hammerle\OP presses or the Welding Section before the end of the investigation on 1st May 1990, some 6 weeks from the start. The graphs of waiting times in front of these sections shows the amount and distribution of waiting that takes place (Graph 6.4.3. and 6.4.4).

The graph of the waiting time between punch and fold operations shows the change in Section Leaders aim.

He only folds what can be welded or painted. The times, although tailing off show an excessive waiting period. The graph of the waiting time between fold and weld operations, continues the section leaders reasoning, as the vast majority of weld\painting is done the same day.

The following graphs were also constructed:

- Total times for Cut\Punch\Fold Operations (Graph 6.4.5.). This graph shown how many jobs were completed in a set time. No jobs took less than four days. The peak number of jobs being placed around 12 day duration. The graph dropping off.

- Return of cards for Cut\Punch\Fold Operations (Graph 6.4.6.). This graph closely mirrors Graph 6.4.5. and the cumulative comparison is shown in Graph 6.4.7. Again a significant number of jobs (22) were completed in over 30 days from the beginning of the project.

- Return of cards for Cut\Punch\Fold Operations cumulative (Graph 6.4.7). The return of cards and time of jobs closely match each other. They both tail off at 26 days into the project.

	Difference
50 Jobs completed in 10 days	
100 Jobs completed in 13 days	3 days
150 Jobs completed in 21 days	8 days
200 Jobs completed in 37 days	16 days

  To complete each extra 50 jobs it effectively takes double the time.

- Job completion rate, actual and simulation (Graph 6.4.8.). This graph shows how dramatically quicker the computer simulation finishes the jobs. Although as might be expected the production rate from the simulation is not constant as the jobs have different processing times.

It has to be remembered that the simulation does 'ramp-up' and then 'ramp-down', even excluding the end reading productivity is significantly up on the actual work being produced.

Examining the 50 and 90 percentile values for completion,

Actual jobs	50%	completed after 17 days.
Simulation jobs	50%	completed after 9 days.
Actual jobs	90%	completed after 38 days.
Simulation jobs	90%	completed after 15 days.

For the same week these figures show that the claimed\allowed production cycle time of 4 weeks was greatly exceeded. This is due in part to the way in which jobs are processed through the sections and "on line shortages" (urgent jobs) take priority as and when they occur.

The following results were obtained from a computer simulation run using the work programmed on the production facility for week 11, 19th March 1990. The setting times used were as agreed with the Work Study Department and the Relevant Section Leaders. The results were as follows:

- SALVAGNINI finished working after 69 hours.
- BEHRENS finished working after 78 hours.
- HAMMERLE finished working after 106 hours.

Now applying these computer run times to the times worked by the sections.

- The SALVAGNINI section works for 63 hrs/wk, implying work completion of approx. 1 week.

- The BEHRENS section works for 73 hrs/wk, implying work completion of approx. 1 week, shortly after the SALVAGNINI.

- The HAMMERLE section works for 40 hrs/wk, implying work completion of approx. 2.5 weeks.

- Leaving 1.5 weeks of the production cycle for the job to be welded\painted\placed into stores for usage.

Please note: that these times have been taken to the nearest hour. The method of loading being alternately high gauge\low gauge\high gauge and the jobs followed defined work routes unlike current shop floor practices, where the jobs are loaded at the Section Leaders discretion. When comparing these actual results with those generated by the computer simulation for the same week. The following points need to be remembered.

- The computer simulation, starts and ends with an empty process, this is not true in practice.

- The computer simulation does not take into account the "on line shortages".

- The computer simulation, used times obtained from the Production Control computer which were on occasion questionable.

These points may account for the experimental error between the actual and simulation times.The difference at 59% found when comparing times to completion of actual work (6 weeks) and simulation (2.5 weeks) cannot be explained away in this manner.

TABLE 6.4.1.

PRODUCT BREAKDOWN FOR WEEK 11

For week 11 the production of the following units was required:

UNITS	NO. OFF	TOTAL PER RANGE	TYPICAL WEEK VALUE
ADT	0	0	6
DX7	3	3	2
DLM 1\4\10	1		
DLM 2\4\10	1	4	7
DLM 3\3\10	2		
DU 202 F6	2		
DU 304 F10	6	8	10
SU 243	1		
SU 323	1		
SU 484	4	13	14
SU 80	7		
UMA 40	5		
UMA 70	1		
UMA 100	10		
UMA 150	33	76	99
UMA 250	17		
UMA 450	10		
DLMV 4	2		
DLMV 6	11		
DLMV 7	2		
DLMV 8	1		
DLMV 9	1		
DLMV 10	2		
DLMV 12	13	51	57
DLMV 14	2		

UNITS	NO. OFF	TOTAL PER RANGE	TYPICAL WEEK VALUE
DLMV 15	3		
DLMV 20	4		
DLMV 30	3		
DLMV 45	6		
DLMV 60	1		

The total number of units produced this week was 155.
The total number of units produced in a typical week was 195.

6.5 RESULTS

6.5.1 OPERATIONS TO BE REMOVED

From the section "Critical Analysis of Production Routes", (Appendix 6.3.2) a number of superfluous operations were found to exist, which needed to be eliminated. In doing so the time to completion for the production process would reduce with less operations to the job.

On the major Cut\Punch\Fold route it is possible to delete 8 operations (Delay\Movement\Storage).

On the O.P. Presses route, it is possible to delete 2 operation (Storage\Movement).

On the on line shortages route, it is possible to delete 8 operations (Delay\Movement\Storage).

6.5.2 PROPOSED RE-ORGANISATION OF MACHINERY AND ROUTES

Before machinery can be moved and routes etc. changed the cost has to be considered and the effects to the machine mechanisms. The costs of machinery movement are included in Appendix 6.5.2. The cost of moving all the machinery would be in the region of £50K. With a new transport system (£20K) the total cost rises to £70K.

Between 55% to 77% of work passes through 2 Hammerle Brake presses. To move these adjacent to the relevant Behrens would only cost £5K. Minimum expenditure for maximum improvement. The other changes required would be a small re-arrangement of a storage area. The rack adjacent to the Salvagnini can be moved to the Hammerle No.5 Brake Press position and the Section Leaders "office" can be moved to the Hammerle No. 1 Brake Press position. This enables forklift trucks to once again use the gangway, whilst minimizing the distances moved by people\tools\jobs. The positioning of new machinery as and when it is purchased is also then not restricted as the cost of positioning the new machinery will far out-weigh the cost of moving the machinery using this plan.

To move more machinery other constraints have to be considered:

- The investment in new machinery at a later date, to obtain maximum advantage this will have to be linked with an automatic job transfer mechanism to and from the machine with a minimum distance between machines. This may cut across current machinery and production routes.

- In moving the machinery, would any irreparable damage occur to it. This is possible with the 2 Laser Behrens. These are old machines and have settled in the one position. Thus to move them, whilst taking time to reset the lasers may also upset the mechanism within them.

With the aid of expanding foam being cut into the shape of the machinery to be moved, various alternative layouts were tried. The one proving to be the best (Appendix 6.2.3.\4\5) involved the following machine movements.

- The guillotine moved to the temporary storage position alongside the Salvagnini shear.

- The Hammerle 1 moved to face the Behrens machines, placed between No. 5 and 6.

- The Hammerle 2 moved to face the Behrens machines, adjacent to Hammerle 1.

- The Hammerle 3 moved to behind Hammerle 2.

- The Hammerle 7 moved adjacent to Hammerle 3.

- The four OP Presses are to be moved to a new section.

- The proposed new machinery, another Behrens\Hammerle and OP Press have been positioned to show how it is expected that the factory will develop.

- The racking adjacent to the Salvagnini shear that holds the tooling for the Hammerles and OP Presses is to be moved adjacent to the new position of Hammerle No. 3.

The achievements will be to open up the gangway through the factory next to the Hammerles. Open up the gangway past the Behrens into Welding. Leave room adjacent to the Salvagnini shear so that new machinery, as and when purchased can be placed to obtain direct links with the Salvagnini. Minimise the distance moved by work through the manufacturing process by the jobs (Appendix 6.2.4\5 for a comparison of current and proposed job routes).

Re-positioning of machinery require capital expenditure. From the section investigation into job movement through the first four sections of the production facility a number of improvements to work plan became apparent. These are included in Appendix 6.5.3. The 15 improvements itemised will require the minimum of expenditure whilst reaping considerable rewards if instigated with the current system. But will be a necessity as and when the MRPII system comes on line as the batches will be smaller and the production times will be more critical. It is concluded that by ordering the job flow through the production process time and financial savings can be made.

C H A P T E R 7

D I S C U S S I O N O F R E S U L T S

DISCUSSION OF RESULTS

DCE's product is good, their market share of 32% in the U.K. (perceived) reflects this. But they have problems within their production facility with a large WIP and inventory. Work has to start on orders before they are placed, in order to meet delivery deadlines. If the market ever goes into recession, this situation will leave DCE at risk financially, with customers also not being satisfied with the delivery times, they will tend to look at other dust control equipment leading to a reduction in DCE's market share.

Examining the market criteria of "what does the customer want, when does he want it, and how much does he want to pay for it", DCE has what the customer wants, but not when he wants it, or the price that he wants to pay for it, because of the inadequacies of the production process. This is highlighted by the fact that currently 15% of DCE (U.K.) annual turnover is held as stock. When viewed from above, the factory floor resembles a warehouse with operatives assembling and working on jobs in odd corners and the space not taken up by stock. Even though all the machinery on the factory is not new it is serviceable and efficient at what it does. Having perceived that change was required to the manufacturing process to remain competitive what change was required and how could it be ratified before it was implemented, to prove that the change would show an improvement, and by how much. It should be noted that there was also a continued investment in new technology but, the financial justification (Section 1.8) was becoming more difficult due to the short payback period allowed. The application and advantages of the machine in use was required to be demonstrated. By using computer modelling this could be achieved very dramatically before the machinery was purchased.

One of the tasks set to a standing committee at DCE (the M.O.R. Committee) was to construct a data base, linking all data available about a part produced into a single file. This was a great undertaking with over 2500 different parts being manufactured. It was hoped that with this information more precise figures could be quoted when different parts were compared. It also helped to demonstrate that different parts could be grouped together and so reduce the setting times on the machines speeding up the rate of production. The Lotus 123 spreadsheet computer software was chosen and the computers memory had to be extended to hold all the information. Throughout the course of the project the data base was being up-dated and checked, so that the information held on it would be correct.

With a data base available, the possibilities of accurate modelling of any proposed changes to the manufacturing system were possible, but what modelling should be done and how. The most relevant seemed to be computer simulation modelling, giving a quick result, being inter-active and possible to change. The results were to be accurate as the actual times and inter-relationships of the machines were programmed into the model. The results being displayed graphically and the process being shown on an animated screen. The computer model chosen was PC MODEL by Simcon Ltd., as it appeared to meet the requirement and have the best back-up available.

By referring to the data base, DCE Management saw a method of improving the speed of manufacturing, as natural gauge brakes and certain groups of components linked together (Families). They proposed that a weeks worth of work should be constantly in storage behind the first machine, the shear, and all work should be issued from there. This immediately meant that an extra 40 tonnes of steel at a cost of £15K was being introduced into a new storage area on the shop floor. This was the first process to be modelled.

The simulation started as a simple model of the Cut\Punch\Fold process. With eight out of the eighteen machines shown on it. The model started out with a sequential reading of the data input from the data base held on LOTUS 123. The DCE series of programmes. Then as scheduling was not wanted it changed to simultaneous data input to keep all the machines busy. The JEF series of programmes. Trying to improve on the performance at this stage it was perceived that the simplification of the factory floor was not giving true results so a more complex model was constructed showing all the machines and routes possible. The JON series of programs. A comparison of the results is shown below.

Lotus Data Base	- Total Machine 135 hrs	-	3.5 weeks
DCE\FINAL.MDL (J3DCE.DAT) Sequential	- Total Machine 260 hrs	-	6.5 weeks
JEF\DCE54.MDL (SCHED3.DAT) Simultaneous	- Total run time 159 hrs	-	4 weeks
JON\JON.MDL (FILE3.DAT) Simultaneous	- Total run time 198 hrs	-	5 weeks

Quoted production cycle time (historical) - 4 weeks.

These figures are based on one file, the 'Typical Weeks' on historical value generated from past production when the maximum production requirement for each product was achieved. Examining these figures:

- Lotus data base shows no inter-relationships between components being machines. No travel or queueing times, and therefore must be assumed as the best time achievable.

- DCE\FINAL.MDL shows the simplified models results running through the Cut\Punch\Fold operations. As all the possible routes are not shown there is an excessive amount of queueing taking place giving an inaccurate result.

- JEF\DCE54.MDL shows the same machine but with a simultaneous reading of the data file so that all the machines are kept constantly supplied with work, but there is no time allowed for the cutting of work, implying that the necessary work is being called out of a store at the start of the simulation.

- JON\JON.MDL shows all the routes\machine operations possible and the cutting operation is included.

When compared to the actual production time allowed, 4 weeks, all these times fail as there are 2 additional operations not taken into account, welding\painting. The settings for the simulation can be changed to increase the work in progress, decrease the travel times, changing the setting times on the machines. These will reduce the completion time for the weeks work, but when compared to the factory as is, are artificial improvements. It has to be remembered that these results are for 1 week only, so that the process starts empty and ends empty. This does not happen in real life, 1 week leading on from the next, which would give a longer completion time. It was shown in the DCE\FINAL.MDL file that by scheduling the work throughput the total run time could be improved (Graph 4.3.1.).

These results show that the planned store and "family part" method would be a retrograde step if implemented in the production facility, so that the computer modelling, whilst not showing an improvement, prevented a worse method of production being implemented on the shop floor.

At this time, partly due to Management changeovers, the philosophy to production changed to that relating to MRP II, and making demand, not to stock. The changes meant that the data used was no longer historical, but current ie. what was actually being processed on the shop floor now. To this end five weeks of production work were modelled. The data preparation in Lotus proving to be long and tedious, but results were achieved and even though all the data required was not available, the results showed that considerable time saving could be made, but these had to be ratified.

To ratify the results of the computer model one weeks jobs were traced through the production process, the relevant data being extracted. This same weeks worth of work was then passed through the computer model. The results showed that on improvement on the historical four week production cycle could be made with the actual production on the shop floor currently taking more than four weeks. Three methods of loading the model were tried, simultaneous, sequential, and Kanban. The simultaneous proved to be the worst one, a large amount of WIP was required to keep all the machines busy. The sequential in the current circumstances proved to be the best as work could be scheduled through the process giving a minimum throuput timw and low WIP levels. The Kanban method would prove to be the best method if the individual machines set up times could be reduced. The time was worse than the sequential method, but with a minimum of WIP. Whilst the weeks jobs were being traced through the manufacturing process, a method study of the process was carried out, to help highlight areas of improvement, these are itemised in Appendix 6.5.3. This also looked into the more long term developments of the factory layout and implementation of new machinery.

The computer modelling demonstrating the best method of placing the work onto the shop floor could be improved to help this speeding up of production.

The project developed from a proving exercise of a proposed manufacturing method using a computer simulation model, to review the whole process and all aspects, showing what improvements are possible to DCE's manufacturing process to maintain its position in BTR's companies performance league and in rediness for 1992 and the single European Market, but meeting customer requirements.

C H A P T E R 8

C O N C L U S I O N S

CONCLUSIONS

In section 1.9 the project objectives were itemised; Have they been achieved? and with what result?

- Construct a component data base, using the Lotus 123 spreadsheet computer software package, for the sheet metal work processed by the factory.

  This was achieved, all the necessary data was input and the file reached an accuracy of 85%.

- Construct a computer model of the Cut\Punch\Fold process carried out on the sheet metal work, using PC MODEL by Simcon Ltd.

  This was achieved several model were constructed from the initial simple model with 8 machines to the complex with all 18 machines represented.

- Create a link between the Lotus 123 and PC MODEL software, so that the data base can be used to provide the data by the computer model. This was achieved, two different methods of data loading into the model, by sequential and simultaneous reading of the data file.

- Verify the computer model, by comparing predicted results with results obtained from the factory.

  This was achieved, initially be comparing the models results to the historical production cycle time and then finally to actually compare the time taken to process the same work on the factory floor, to the time taken to process that work though the computer model.

- Demonstrate what and where improvements could be made to the sheet metal manufacturing process, using method study techniques.

This was achieved. It was found that work travelled through excessive horizontal and vertical distances whilst being processed. The routes that the work flowed through were congested and random, leading to specific improvements being outlined.

Reviewing the project more generally, the computer modelling, it was shown that although the Family method of processing the parts was efficient it was not the most effective. The most effective being by alternating gauge (High\Low\High) with the parts where possible in their Family, but producing what is required production, not based on historic values.

The project moved on to review the other aspects of production, the machine layout, a more efficient layout with allowances for new investments in machinery have been proposed. Whilst the results from the simulation were confirmed, by data obtained from the shop floor. Other immediate improvements to the method of processing the work through the shop floor were recommended. To obtain\maintain improvements on the shop floor three items are required:

- People to use the system
- Software to run the system.
- Hardware to optimise the system.

The workforce is not currently achieving their maximum output of what is required when it is required. Due in part to the software, but more importantly to their method of payment, a piece part bonus scheme. Britain being one of the few developed countries to persist in this method of payment.

The software controls the whole of the production system. This is currently inaccurate and unreliable. When the move to MRPII is made a drastic improvement in stock levels and amount of work being processed should be seen.

The hardware is the production system, all the machines that are on the shop floor are operations, there is investment in new machinery planned for the shop floor, to improve setting and transport times.

To obtain the most from the Company all three items have to be considered, when one is not considered the other two fail.

DCE is in the position to make vast improvements to its manufacturing facility in the next 2\3 years, but the leadership of Senior Management will be required, as the drive and power to implement changes does not well up from the shop floor!

CHAPTER 9

RECOMMENDATIONS

RECOMMENDATIONS

- Implementation of recommendations in Appendix 6.5.3. to help order and improve productivity in readiness for the introduction of MRPII and B.S.5750.

- Company wide education re:the changes about to take place, why they are to take place, and the perceived results.

- Implementation of MRPII - taking place - This has to be successful to bring W.I.P. down and achieve successful manufacturing planning.

- Implementation of B.S.5750 - taking place - This has to be successful as more customers require this standard to be met. It also brings a responsibility to deliver what the customer requires.

- Capacity Planning - to take place after MRPII is installed.

- Shop Floor data collection - to take place after MRP II is installed.

- Standardisation of parts\gauge where possible, to help ease production.

- Redesign products to achieve variety through reduction part count, pushing for a reduction in end products.

- Payment scheme (company wide), a standard wage with monthly bonus related to the numbers of units supplied to customers and who are satisfied with them.

- Purchasing of materials - a balance is required so that enough raw material is on site to produce the requirement component.

- Movement\Optimising of the machines on the shop floor to minimise the time wasted, and maximise the usage of the shop floor area available. Speeding up production.

- Review of the processes to find new ways to improve the productivity of the factory.

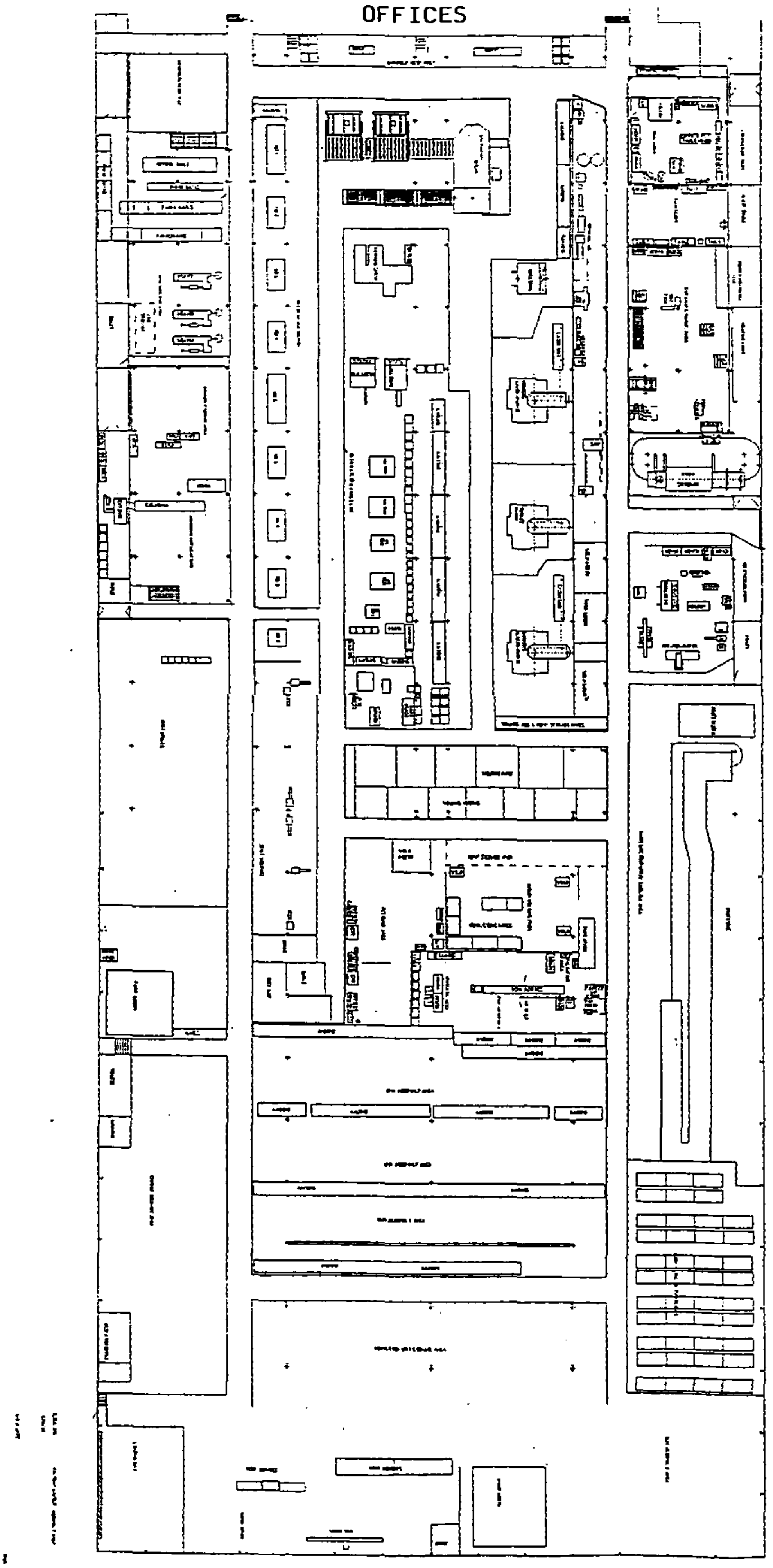
- Investment in new machines - Behrens\Hammerle type - on going. Moving with technology will enable greater productivity as advances are made in machine speed\setting\transfer between machines.

CHAPTER 10

APPENDICES AND GRAPHS

APPENDIX 1

LAYOUT DRAWING OF FACTORY FLOOR



SCALE - 1.600

DATE - 1988

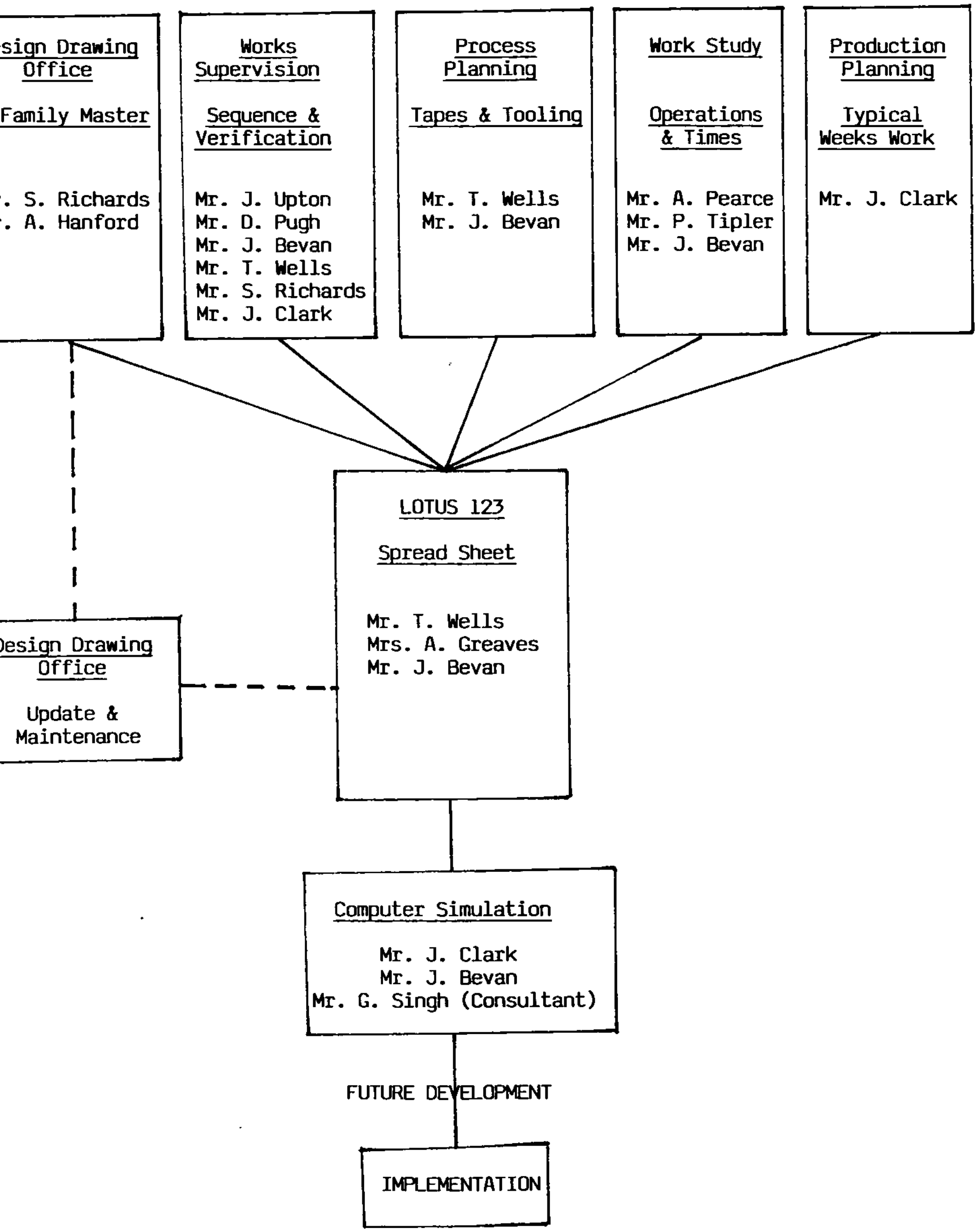
APPENDIX 2

COMPARISON BETWEEN VS6 AND PC MODEL COMPUTER SIMULATION PACKAGES

SYSTEM:	VS6	PC MODEL
UNIVERSITY:	L.U.T.	COVENTRY POLY.
COST:	£900 + VAT CONSULTATION £700 + VAT SOFTWARE	? £10,000
COMPUTER:	IBM COMPATABLES MIN, 512K RAM	IBM COMPUTER
FACILITIES AVAILABLE:	GRAPHICS NO ROUTE BETWEEN M/C'S PARTS MOVE FROM ONE POSITION TO THE NEXT.	GRAPHICS ROUTE IS MODELED, FLOW OF MOVEMENT OF PARTS IS SHOWN BETWEEN M/C'S GRAPHICAL REPRESENTATION SUPRIOR TO VS6.
	NUMERICAL O/P AVAILABLE GRAPHICAL REPRESENTATION OF QUEUES, UTILISATION ETC. AVAILABLE.	NUMERICAL O/P AVAILABLE GRAPHICAL REPRESENTATION OF QUEUES, UTILISAITON ETC. AVAILABLE.
LIMITS ON CAPABILITY:	INITIAL MODELING OF SHEAR, BEHRENS, PRESSES THEN INTRODUCTION OF WELDING AND PAINT LINE/ ASSEMBLY WILL NOT BE ABLE TO BE MODIFIED.	PUT FORWARD AS BEING ABLE TO MODEL THE WHOLE FACTORY.
ADVANTAGES:	CHEAP (£).	CAN COMMUNICATE WITH LOTUS 123. V.FAST DATA I/P FROM CURRENT DATA ON LOTUS. MORE USER FRIENDLY THAN VS6??
TIME REQUIRED TO MODEL PLANT:	43 DAYS + DATA I/P	?
USER COMPATIBLE:	GRAPH/DIAGRAM OF ROUTE HAS TO BE DRAWN FIRST. THEN I/P TO VS6. ALL DATA I/P IS MANNUAL.	SHOWN TO BE DIRECT I/P OF ROUTE / M/C'S ETC. THEN DIRECT LINK TO DATA.

APPENDIX 3.1.1.

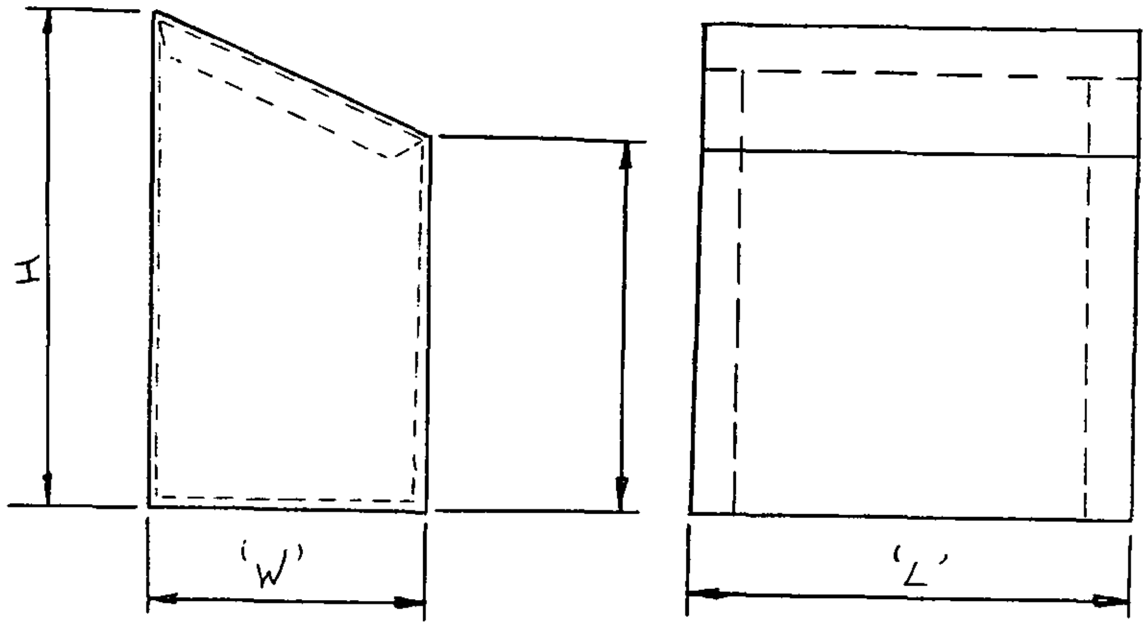
DIAGRAM SHOWING ROUTE OF DATA COLLECTING AND THE PEOPLE INVOLVED



APPENDIX 3.1.2.

ONE PAGE FROM THE "FAMILY PART FOLDER"

DLM-V WEATHER COWL



	TYP WK QT	WORKS NO.	DRG. NO.	DESCRIPTION	DIM H1	DIM H2	DIM 'W'	DIM 'L'
	N/A	1.HW.1	345-2438	V4/6 1MM THK 001	320	256	110	280
	1	1.HW.2	345-2439	1MM THK 002	365	266	172	340
FOLD	2	1.HW.3	345-2441	V30/15 1MM THK 003	390	278	196	380
		1.HW.4	345-2442	V21/30/45 1.2MM THK	548	395	265	590
	2	7.40MF.1	640-2913	UMA 40 DISCHARGE COWL 1.2 005	215	170.5	80	335

APPENDIX 3.1.3.

PRINT OUT FROM BURROUGHS COMPUTER/CANDE SOFTWARE (1 PAGE ONLY)

PROGRAM WORK · DCE GROUP 'TMS' REPORTS 14.03.89 TIME: 11.38.36

OPERATION BATCH SIZES
****

**UMA DOOR PANEL	***PART NOT FOUND***						
	UMA FAN CHAMBER DOOR ASSY	10.00	A	10	0.2500	0.1330	CUT TO SIZE:441.5 X 546 X 1.5 FROM 1250 X 2500. MAKES 8
			B2	20	0.0000	0.1210	PUNCH COMPLETE TAPE NO.202(2) BROWN
			J	30	0.0700	0.2020	FOLD X 4 - 4 STROKES
			I	40	0.1500	0.2500	WELD 4 CORNERS OF DOOR PANEL
4 100 2	UMA 100/150 FAN CHAMBER DOOR ASSY	10.00	A	10	0.2500	0.1330	CUT TO SIZE: 523 X 736 X 1.5
			J	30	0.700	0.2170	FOLD X 4 - 4 STROKES P.P.
			L	40	0.1500	0.2570	WELD 4 CORNERS

BEHREN'S HAMMERLE

WEEKLY LOADING (THE LOTUS 123 SPREADSHEET)

7

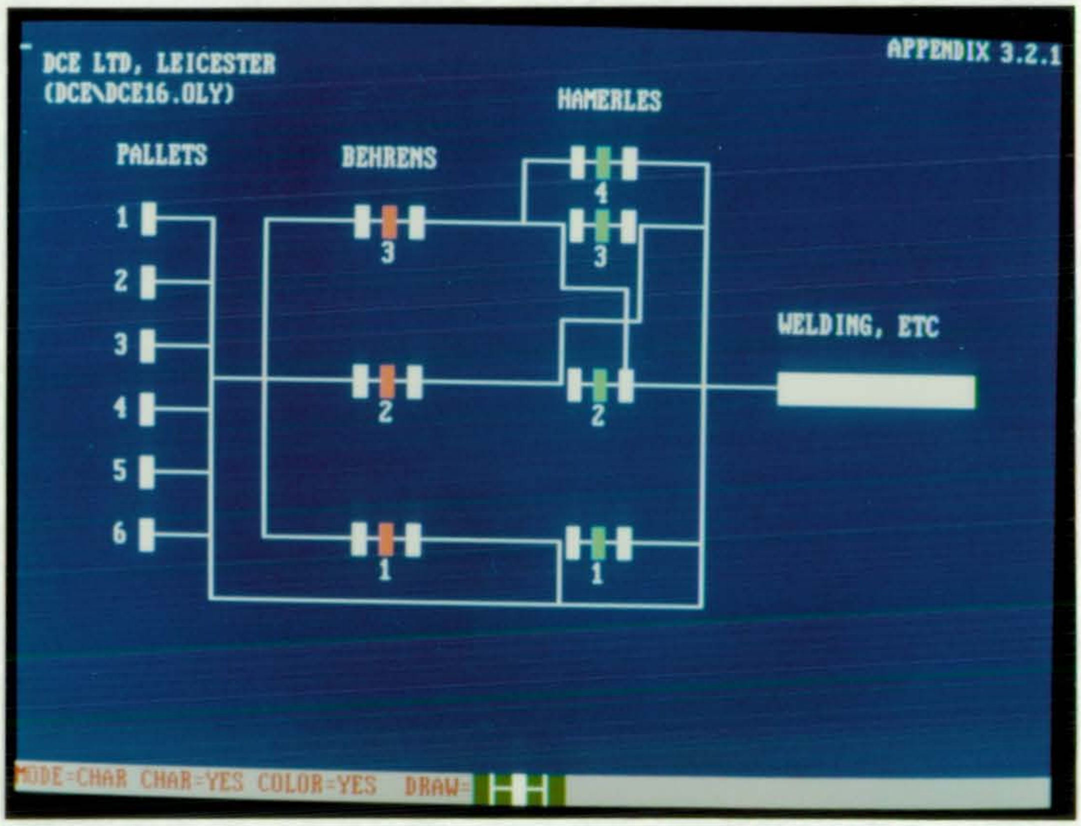
APPENDIX 3.1.4.

PART NO.	DESCRIPTION	DRG.NO.	DIMS OF BLAN	TAPE NO.	PUNCHES RND.	SQU.	SLOT	OB.RND	M.M. GAUGE	LOW/ HIGH DIE	TOOL QTY.	QTY /WK	BEHRENS ALLOWED HRS EA.	TOTAL HRS/WK.	"J" TIME	FAMILY NAME	HAMMERLE ALLOWED HRS.EA.	TOTAL HRS/W	"J" TIME	ROUTINGS /REMARKS	MODEL RANGE
1 HW 1	DLM V WEATHER COWL	345 2438	405*552*1	LBLK-1915B(4)				12X8	1.00	L	1	0	0.026		0.30	F01 0010 001	0.064		0.30	S/B7/H1	DLM-V
1 HW 2	DLM V WEATHER COWL	345 2439	493.5*736*1	LBLK-1914B(2)				20X10	1.00	L	1	1	0.029	0.03	0.10	F01 0010 002	0.064	0.06	0.10	S/B7/H1	DLM-V
1 HW 3	DLM V WEATHER COWL	345 2441	533*829*1	LBLK-1913B(2)				20X10	1.00	L	1	2	0.037	0.07	0.10	F01 0010 003	0.064	0.13	0.10	S/B7/H1/H3	DLM-V
1 HW 4	DLM V WEATHER COWL	345 2442		LBLK-1912C(3)			20X10		1.20	L	2	0				F01 0010 004				S/B7/H1/H3	DLM-V
1 HW 4	DLM V WEATHER COWL	345 2442	743*1201*1.2	LBLK-1912C(3)			20X12		1.20	L	2	0	0.033		0.10	F01 0010 004	0.079		0.10	S/B7/H1/H3	DLM-V
7 40MF 1	UMA 40 DISCHARGE COWL	640 2913	278*528*1.2	LBLK-2480(1)	6				1.20	L	1	2	0.023	0.05	0.10	F01 0010 005	0.045	0.09	0.10	S/B7/H1	UMA
1 B 3B	UMA 70 INLET SPIGOT TOP	640 3003/2		BRN-2577		30	6x10		1.20	L	2	2	0.001	0.00	0.20	F01 0020 001	0.016	0.03	0.20	S/B7/H1	UMA
9 B 3B	UMA 100/150 INLET SPIG.TOP	640 3004/2		BRN-2579		30	6x10		1.20	L	2	24	0.002	0.05	0.10	F01 0020 002	0.016	0.38	0.10	S/B7/H1	UMA
19 B 3B	UMA 250 INLET SPIGOT TOP	640 3005/2		BRN-2581		30	6x10		1.20	L	2	40	0.002	0.08	0.10	F01 0020 003	0.016	0.64	0.10	S/B7/H1	UMA
32 40 1B	UMA 40 INLET SPIGOT TOP	640 3319/2		GRN-3080(2)	30		6X10		1.20	L	2	4	0.002	0.01	0.10	F01 0020 004	0.016	0.06	0.10	S/B7/H1	UMA
2 B 2	UMA S/GEAR RETAINING RING		1250*1250*1	BLK-2050E(3)			6X10		1.00	L	1	70	0.002	0.14	0.10	F01 0020 005	0.029	2.03	0.10	S/B7/H1	UMA
1 B 3A	UMA 70 INLET SPIGOT BODY	640 3003/1		LBLK-2575	40		5X50		1.20	L	4	0	0.000	0.00		F01 0030 001	0.000	0.00		S/B7/H1	UMA
1 B 3A	UMA 70 INLET SPIGOT BODY	640 3003/1		LBLK-2575	6		6X10		1.20	L	4	1	0.015	0.02	0.30	F01 0030 001	0.042	0.04	0.30	S/B7/H1	UMA
1 B 3AX	UMA 70 BIN BAL.IN.SPIG.BODY			LBLK-2575(3)	6		6X10		1.20	L	4	1	0.013	0.01	0.10	F01 0030 002	0.042	0.04	0.10	S/B7/H1	UMA
1 B 3AX	UMA 70 BIN BAL.IN.SPIG.BODY			LBLK-2575(3)	40		5X50		1.20	L	4	0	0.000	0.00		F01 0030 002	0.000	0.00		S/B7/H1	UMA
9 B 3A	UMA 100/150 IN.SPIG.BODY	640 3004/1		LBLK-2578	6		6X10		1.20	L	3	21	0.015	0.32	0.10	F01 0030 003	0.042	0.88	0.10	S/B7/H1	UMA
9 B 3A	UMA 100/150 IN.SPIG.BODY	640 3004/1		LBLK-2578	40				1.20	L	3	0	0.000	0.00		F01 0030 003	0.000	0.00		S/B7/H1	UMA
9 B 3AX	UMA 100/150 BIN BAL.IN.SPIG.			LBLK-2578	40				1.20	L	3	0	0.000	0.00		F01 0030 004	0.000	0.00		S/B7/H1	UMA
9 B 3AX	UMA 100/150 BIN BAL.IN.SPIG.			LBLK-2578	6		6X10		1.20	L	3	3	0.015	0.05	0.10	F01 0030 004	0.042	0.13	0.10	S/B7/H1	UMA
19 B 3A	UMA 250 INLET SPIG. BODY	640 3005/1	1240*1400*1.	LBLK-2580(2)	40		6X10		1.20	L	3	0	0.000	0.00		F01 0030 005	0.000	0.00		S/B7/H1	UMA
19 B 3A	UMA 250 INLET SPIG. BODY	640 3005/1	1240*1400*1.	LBLK-2580(2)	6				1.20	L	3	37	0.019	0.70	0.10	F01 0030 005	0.042	1.55	0.10	S/B7/H1	UMA
19 B 3AX	UMA 250 BIN BAL.IN.SPIG.BODY		1240*1400*1.	LBLK-2580(2)	40				1.20	L	3	0	0.000	0.00		F01 0030 006	0.000	0.00		S/B7/H1	UMA
19 B 3AX	UMA 250 BIN BAL.IN.SPIG.BODY		1240*1400*1.	LBLK-2580(2)	6		6X10		1.20	L	3	3	0.019	0.06	0.10	F01 0030 006	0.042	0.13	0.10	S/B7/H1	UMA
32 40 1A	UMA INLET SPIGOT BODY	640 3319/1		LBLK-3079(2)	40				1.20	L	3	0	0.000	0.00		F01 0030 007	0.000	0.00		S/B7/H1	UMA
32 40 1A	UMA INLET SPIGOT BODY	640 3319/1		LBLK-3079(2)	6		6X10		1.20	L	3	4	0.014	0.06	0.10	F01 0030 007	0.042	0.17	0.10	S/B7/H1	UMA

APPENDIX 3.2

WORK ROUTES AVAILABLE IN DCE AND JEF PROGRAMS

ROUTE	SALVAGNINI	BEHRENS	HAMMERLE	WELDING
1	"	1	1	"
2	"	2	2	"
3	"	3	3	"
4	"	3	4	"
5	"	2	2 3	"
6	"	3	3 2	"
7	"	-	1	"
8	"	-	-	"
9	"	3	3 23	"



APPENDIX 3.2.1.

(PLATE 15)

```
O=(DCE\DCE16.OLY)              ;DESIGNATES OVERLAY

@CUTTER=(0)                    ;SET COUNTERS TO ZERO
@THISFAM=(0)
@THISBAT=(0)
@THISSTR=(0)
@BESTBAT=(0)
@BESTPTY=(0)
@R1PROD=(0)
@LASTCUT=(0)
@LASTPUN1=(0)
@LASTPUN2=(0)
@LASTPUN3=(0)
@LASTBND1=(0)
@LASTBND2=(0)
@LASTBND3=(0)

#EOF_FLAG=(999)                ;SET END COUNTERS
#HIGH=(999)

%MOVE=(0:00:10.00)             ;SET CONSTANT TIMES
%CUTTER=(0:01:10.00)
%PUNCH=(0:01:10.0)
%BEND=(0:01:10.0)

%CUT_SAME=(0:00:05.00)
%CUT_DIFF=(0:00:15.00)

%SETCUT_SAME=(0:06:00.00)
%SETCUT_DIFF=(0:18:00.00)

%CLOCK=(00:00:00.00)
%WORK=(00:00:00.00)

@@FAMILY=(3,100)               ;SET UP ARRY SIZE
#FFLG=(0)                      ;SET UP ARRY CONTENTS
#FPRT=(1)
#FSTR=(2)
#FBAT=(3)

@@SCHED=(5,100)
#SFLG=(0)
#SFAM=(1)
#SQTY=(2)
#SRTE=(3)
#SPTY=(4)
#SASC=(5)

%XSCHED=(3,100)
#SCUT=(1)
#SPUN=(2)
#SBND=(3)

+CUTTER0=(XY(10,4))            ;SET START POSITIONS
+CUTTER=(XY(0,0))
+START2=(XY(6,6))

J=(1,I,1,0,0,0,1)              ;SET JOB QUANTITIES
J=(2,A,2,0,0,0,100)
                               ;BEGIN INITIALISING
BR(1,XY(0,1),0)                ;BEGIN ROUTE
   SV(@THISFAM,1)              ;SET START OF TEST POINTERS
   SV(@THISSTR,1)
```

```
   OF(DCE\JDCE.DAT)                   ;OPEN FILE
:R1/10                                 ;LOAD IN DATA FROM FILE
   GD(@R1PROD)                         ;GET FIRST ENTRY IN LINE
   IF(@R1PROD,EQ,#EOF_FLAG,:R1/50)     ;QUIT IF END OF FILE FLAG
   SV(@@SCHED(#SFLG,@R1PROD),1)        ;SET FLAG TO 1=LINE USED
   GD(@@SCHED(#SFAM,@R1PROD))          ;LOAD REST OF LINE ..
   GD(@@SCHED(#SQTY,@R1PROD))
   GD(@@SCHED(#SRTE,@R1PROD))
   GD(@@SCHED(#SPTY,@R1PROD))
   GD(%%SCHED(#SCUT,@R1PROD))
   GD(%%SCHED(#SPUN,@R1PROD))
   GD(%%SCHED(#SBND,@R1PROD))
   GD(@@SCHED(#SASC,@R1PROD))
   JP(:R1/10)                          ;NEXT LINE (EOF_FLAG STOPS LOOP)
:R1/50
ER                                     ;END OF ROUTE

BL(!CUTTER)                            ;FIND FREE CUTTER
   :CT/10
      SV(@CUTTER,0)                    ;SET LOOP COUNT TO ZERO
      SV(*CUTTER,*CUTTER0)             ;SET ZERO POSITION
   :CT/20
      AO(*CUTTER,+,320)                ;ADD TWO LINES (2LINES*80COLS*2BYTES)
      JC(*CUTTER,:CT/30)               ;JUMP IF POSITION CLEAR
      IV(@CUTTER)                      ;ELSE INCREMENT LOOP COUNT
      IF(@CUTTER,LT,6,:CT/20)          ;TRY NEXT IF NOT TRIED ALL
      WE                               ;ELSE WAIT EVENT..
      JP(:CT/10)                       ;..AND RESTART
   :CT/30
      MA(*CUTTER,0)                    ;MOVE TO VARIABLE POSITION
      IF(OBJ@2,EQ,@LASTCUT,:CT/40)     ;TEST IF THIS EQ LAST
      WT(%CUT_DIFF)                    ;IF NO, WAIT DIFF SET TIME
      JP(:CT/50)                       ;CONTINUE
   :CT/40
      WT(%CUT_SAME)                    ;ELSE WAIT SAME SET TIME
   :CT/50
      SV(%CUTTER,%%SCHED(#SCUT,OBJ@1))     ;GET UNIT CUT TIME
      AO(%CUTTER,*,@@SCHED(#SQTY,OBJ@1))   ;CALC TOTAL TIME
      WT(%CUTTER)                          ;WAIT CUT TIME
      SV(@LASTCUT,OBJ@2)                   ;SAVE LAST FAMILY TYPE
EL

BR(2,XY(0,2),0)                        ;RELEASE LOGIC
:R2/10
   SV(@BESTPTY,#HIGH)                  ;SET BEST PRIORITY TO HIGH
   SV(@BESTBAT,0)                      ;ASSUME NO BEST LINE
   SV(@THISBAT,@THISSTR)               ;GET START BATCH OF CURRENT FAMILY
   DV(@THISBAT)                        ;DECREMENT TO MAKE NICE TEST LOOP
:R2/20
   IV(@THISBAT)                              ;INCREMENT TO NEXT BATCH
   IF(@@SCHED(#SFAM,@THISBAT),NE,@THISFAM,:R2/50)     ;QUIT LOOP IF
                                             ;THIS BAT NOT THIS FAMILY
   IF(@@SCHED(#SFLG,@THISBAT),NE,1,:R2/20) ;NEXT IF THIS BAT HAS
                                             ;BEEN TAKEN
   IF(@@SCHED(#SQTY,@THISBAT),EQ,0,:R2/20) ;NEXT IF THIS QTY IS ZERO
   IF(@@SCHED(#SPTY,@THISBAT),GE,@BESTPTY,:R2/20)     ;NEXT IF THIS
                                             ;PRIORITY HIGHER THAN BEST
   SV(@BESTPTY,@@SCHED(#SPTY,@THISBAT))      ;ELSE GET NEW BEST PRIORITY
   SV(@BESTBAT,@THISBAT)                     ;AND NEW BEST BATCH
   JP(:R2/20)

:R2/50
  IF(@BESTBAT,GT,0,:R2/100)             ;JUMP IF VALID BEST BATCH
  SV(@THISFAM,@@SCHED(#SFAM,@THISBAT)) ;ELSE GET NEXT FAMILY
             ;(THIS BAT ENDS POINTING TO NEXT FAMILY START)
  IF(@THISFAM,EQ,0,WAIT)                ;STOP FLOW IF NO NEXT FAMILY
```

```
    SV(@THISSTR,@THISBAT)              ;SAVE START BATCH FOR FAMILY
    JP(:R2/10)

  :R2/100
    SV(OBJ@1,@BESTBAT)                 ;SET BATCH
    SV(@@SCHED(#SFLG,OBJ@1),2)         ;SET FLAG TO BATCH STARTED
    SV(OBJ@2,@@SCHED(#SFAM,OBJ@1))     ;GET FAMILY
    SV(OBJ@3,@@SCHED(#SQTY,OBJ@1))     ;GET QTY
    SV(OBJ@4,@@SCHED(#SRTE,OBJ@1))     ;GET ROUTE
    SV(OBJ@ID,@@SCHED(#SASC,OBJ@1))    ;GET ASCII VALUE OF ID

  ;MOVE LOGIC
  ;==========

      MA(*START2,%MOVE)                ;MOVE ABSOLUTE
      SV(OBJ%1,CLOCK)                  ;SET VALUE
      LK(!CUTTER)                      ;LINK
      MR(5,%MOVE)                      ;MOVE RIGHT
      MA(XY(15,11),%MOVE)
      MR(4,%MOVE)

      IF(OBJ@4,EQ,1,:ROUTE1)           ;DESIGNATE JOB ROUTES
      IF(OBJ@4,EQ,2,:ROUTE2)
      IF(OBJ@4,EQ,3,:ROUTE3)
      IF(OBJ@4,EQ,4,:ROUTE3)

:ROUTE1
      MD(5,%MOVE)
      MR(5,%MOVE)
      JC(XY(25,16),:CONT1)             ;JUMP CONDITIONAL
:CONT1
      MA(XY(26,16),%MOVE)
      IF(OBJ@2,EQ,@LASTPUN1,:SET1)
      WT(%SETCUT_DIFF)                 ;WAIT
      JP(:CONT11)                      ;JUMP
:SET1
      WT(%SETCUT_SAME)

:CONT11
      TP(XY(28,16))                    ;TEST POSITION
      MA(XY(28,16),%MOVE)
      SV(%PUNCH,%%SCHED(#SPUN,OBJ@1))
      AO(%PUNCH,*,@@SCHED(#SQTY,OBJ@1)) ;ARITHMETRIC OPERATION
      WT(%PUNCH)
      SV(@LASTPUN1,OBJ@2)
      JP(:CONT12)

:CONT12
      TP(XY(30,16))
      MR(2,%MOVE)
      WT(00:30:00.0)
      SV(%WORK,CLOCK)
      AO(%WORK,-,OBJ%ST)
      SV(OBJ%2,%WORK)
      TP(XY(42,16))
      JP(:CONT13)

:CONT13
      TP(XY(44,16))
      MA(XY(42,16),%MOVE)
      IF(OBJ@2,EQ,@LASTBND1,:SET11)
      WT(%SETCUT_DIFF)
      JP(:CONT14)
:SET11
      WT(%SETCUT_SAME)
```

```
:CONT14
        MA(XY(44,16),%MOVE)
        SV(%BEND,%%SCHED(#SBND,OBJ@1))
        AO(%BEND,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND)
        SV(@LASTBND1,OBJ@2)
        MR(2,%MOVE)
        MR(1,%MOVE)
        MR(5,%MOVE)
        MU(5,%MOVE)                         ;MOVE UP
        MA(XY(72,11),%MOVE)
        WT(01:00:00.0)
        JP(:END)

:ROUTE2
        MR(5,%MOVE)
        JC(XY(25,11),:CONT2)
:CONT2
        MA(XY(26,11),%MOVE)
        IF(OBJ@2,EQ,@LASTPUN2,:SET2)
        WT(%SETCUT_DIFF)
        JP(:CONT21)
:SET2
        WT(%SETCUT_SAME)

:CONT21
        TP(XY(28,11))
        MA(XY(28,11),%MOVE)
        SV(%PUNCH,%%SCHED(#SPUN,OBJ@1))
        AO(%PUNCH,*,@@SCHED(#SQTY,OBJ@1))
        WT(%PUNCH)
        SV(@LASTPUN2,OBJ@2)
        JP(:END)

:ROUTE3
        MU(5,%MOVE)
        MR(5,%MOVE)
        JC(XY(25,6),:CONT3)
:CONT3
        MA(XY(26,6),%MOVE)
        IF(OBJ@2,EQ,@LASTPUN3,:SET3)
        WT(%SETCUT_DIFF)
        JP(:CONT31)
:SET3
        WT(%SETCUT_SAME)

:CONT31
        TP(XY(28,6))
        MA(XY(28,6),%MOVE)
        SV(%PUNCH,%%SCHED(#SPUN,OBJ@1))
        AO(%PUNCH,*,@@SCHED(#SQTY,OBJ@1))
        WT(%PUNCH)
        SV(@LASTPUN3,OBJ@2)
        JP(:END)

   ;END LOGIC
:END
      SV(@@SCHED(#SFLG,OBJ@1),3)                ;SET FLAG TO BATCH ENDED
   ER
```

;DCE\DCE16.DAT

;PROD	FAMILY	QTY	ROUTE	PRTY	CUT	PUNCH	BEND	ASCII
1	1	98	1	3	0:00:30.00	0:01:35.00	0:02:20.00	65
2	1	196	1	2	0:00:04.00	0:00:15.00	0:00:10.00	65
3	1	64	1	1	0:00:05.00	0:01:25.00	0:03:00.00	65
4	1	64	1	4	0:00:08.00	0:00:04.00	0:01:22.00	65
5	2	0	2	1	0:00:05.00	0:00:30.00	0:00:20.00	66
6	2	20	2	2	0:00:05.00	0:00:30.00	0:00:20.00	66
7	2	20	2	3	0:00:05.00	0:00:30.00	0:00:20.00	66
8	2	20	2	4	0:00:05.00	0:00:30.00	0:00:20.00	66
9	2	20	2	5	0:00:05.00	0:00:30.00	0:00:20.00	66
10	3	10	3	1	0:00:00.00	0:00:45.00	0:00:30.00	67
11	3	10	3	2	0:00:00.00	0:00:45.00	0:00:30.00	67
12	3	10	3	3	0:00:00.00	0:00:45.00	0:00:30.00	67
13	3	10	3	4	0:00:00.00	0:00:45.00	0:00:30.00	67
14	3	10	3	5	0:00:00.00	0:00:45.00	0:00:30.00	67
15	3	10	3	6	0:00:00.00	0:00:45.00	0:00:30.00	67
16	4	30	4	2	0:00:05.00	0:00:30.00	0:00:10.00	68
17	4	30	4	1	0:00:05.00	0:00:30.00	0:00:10.00	68
18	5	15	1	1	0:00:05.00	0:00:35.00	0:00:25.00	69
999								

PCModel Statistics File Template APPENDIX 3.2.4

******PRESS ALT-I TO READ IN STATISTICS FILE

Total Hours: 20 Total Tools: 1

	THRUPUT
MEAN	0.90
STD-D	1.76
MAX	6.00
MIN	0.00

Hour	THRUPUT
1	6.00
2	6.00
3	1.00
4	0.00
5	0.00
6	0.00
7	0.00
8	1.00
9	1.00
10	1.00
11	0.00
12	0.00
13	0.00
14	0.00
15	1.00
16	1.00
17	0.00
18	0.00
19	0.00
20	0.00

Statistics from PCModel session converted by PCMLOTUS

PCModel Statistics File Template

APPENDIX 3.3.4

******PRESS ALT-I TO READ IN STATISTICS FILE

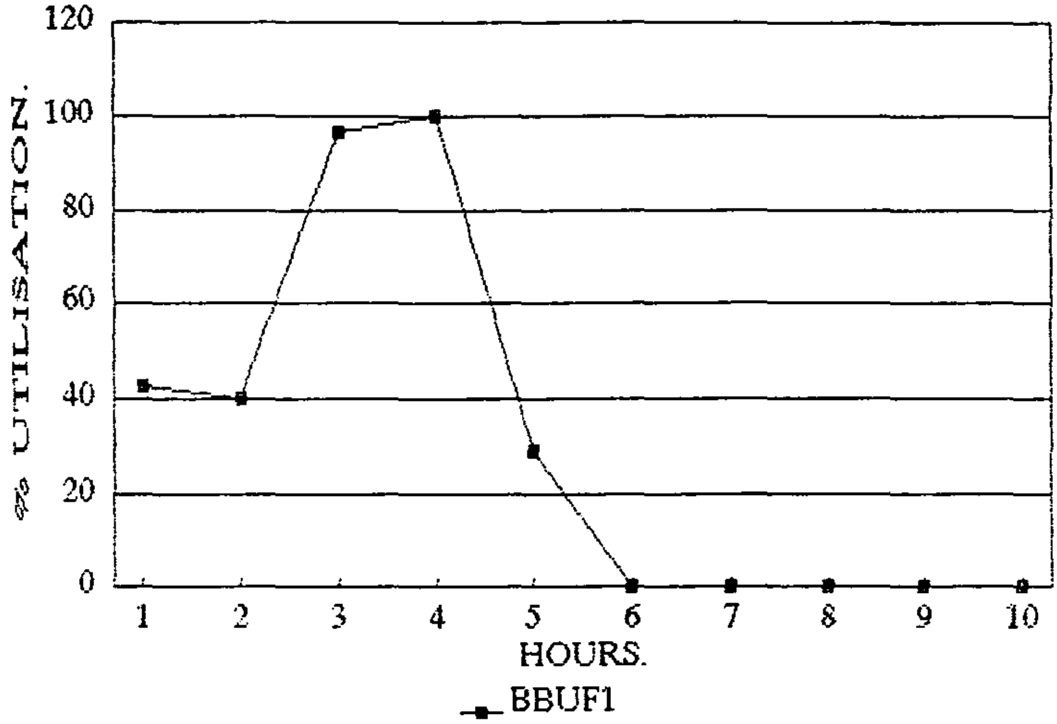
Total Hours: 10 Total Tools: 16

	BBUF1	BEHR1	BBUF2	BEHR2	BBUF3	BEHR3	HBUF1	HAME1	HBUF2	HAME2	HBUF3	HAME3	HBUF4	HAME4	STORE	THRUPUT
MEAN	30.84	38.10	24.22	18.85	15.14	7.64	55.72	47.26	47.17	50.35	13.14	4.86	6.06	2.28	0.47	1.80
STD-D	37.54	43.32	38.89	32.73	31.07	15.37	41.58	40.28	45.16	42.90	27.80	8.87	14.95	6.83	0.66	2.32
MAX	100.00	100.00	100.00	100.00	91.25	42.08	100.00	100.00	100.00	100.00	91.38	28.61	49.86	22.77	1.94	7.00
MIN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Hour	BBUF1	BEHR1	BBUF2	BEHR2	BBUF3	BEHR3	HBUF1	HAME1	HBUF2	HAME2	HBUF3	HAME3	HBUF4	HAME4	STORE	THRUPUT
1	42.63	41.97	92.08	30.97	91.25	34.30	0.00	0.00	44.58	8.61	32.36	11.38	10.69	0.00	0.00	1.00
2	39.97	100.00	100.00	100.00	60.13	42.08	59.75	29.47	25.55	11.38	91.38	28.61	49.86	22.77	1.94	7.00
3	96.66	100.00	50.13	57.52	0.00	0.00	55.52	42.47	100.00	35.97	7.63	8.61	0.00	0.00	1.38	5.00
4	100.00	100.00	0.00	0.00	0.00	0.00	0.86	0.00	100.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00
5	29.13	39.00	0.00	0.00	0.00	0.00	100.00	70.58	100.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00
6	0.00	0.00	0.00	0.00	0.00	0.00	100.00	100.00	100.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.00	0.00	0.00	0.00	0.00	0.00	100.00	100.00	1.52	100.00	0.00	0.00	0.00	0.00	0.27	1.00
8	0.00	0.00	0.00	0.00	0.00	0.00	100.00	100.00	0.00	47.58	0.00	0.00	0.00	0.00	0.27	1.00
9	0.00	0.00	0.00	0.00	0.00	0.00	41.08	30.11	0.00	0.00	0.00	0.00	0.00	0.00	0.83	3.00
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Statistics from PCModel session converted by PCMLOTUS

UTILISATION OF BUFFER 1.



```
      ;BL(!STORE)
      ;MR(5,%MOVE)
      ;MR(14,%MOVE)
      ;WT(30)
      ;EL

      ;LK(!STORE)                    ;HAS BEEN CHANGED TO

       TP(*STORE)
       MA(*STORE,%MOVE)
       MA(*END,%MOVE)                ;ON EACH OCCASSION THAT
                                     ;IT IS REQUIRED
       JP(:END)
```

```
KBCIN  FN=DCE\DCE18.MDL                    SZ=10625  L=0358  C=033  Help=F1
```

```
;MOVE LOGIC
;==========

       MA(*START2,0)
       LK(!CUTTER)
       SV(OBJ%1,CLOCK)

;      SV(%%PROD_TIMES(#PFLG,OBJ@1),CLOCK)
;      SV(%%PROD_TIMES(#PFLG,OBJ@1),OBJ%1)  IF WANT TO START
;                                           IMMEDIATELY AT SHEAR
;      BOTH HAVE BEEN OMMITTED

       LK(!MOVE)
```

```
KBCIN  FN=DCE\DCE18.MDL                    SZ=10625  L=0254  C=033  Help=F1
```

```
:CONT12
       TP(*B14)
       MA(*B14,%MOVE)

      ;SV(%WORK1,CLOCK)
      ;AO(%WORK1,-,OBJ@1)
      ;SV(OBJ%2,%WORK1)
      ;SV(%%PROD_TIMES(#PBEH,OBJ@1),%WORK1)      ;HAVE BEEN OMMITTED
                                                 ;THROUGHOUT THE PROGRAM
       JC(*H12,:CONT13)
```

```
  @@SCHED=(6,200)              ;EXTRA COLUMN 5-6
  #SFLG=(0)
  #SFAM=(1)
  #SQTY=(2)
  #SRTE=(3)
  #SPTY=(4)
  #SASC=(5)
  #SAID=(6)                    ;LINE IDENTIFICATION

KBCIN  FN=DCE\DCE19.MDL              SZ=10618  L=0071  C=001  Help=F1
```

```
:ROUTE1
       MD(5,%MOVE)
       MR(5,%MOVE)

;      JC(*B12,:CONT1)         ;CONDITIONAL JUMP

       TP(*B12,*B13)           ;REPLACED BY TEST POSITION

;:CONT1                        ;LABEL ALSO REMOVED

       MA(*B12,%MOVE)
       IF(OBJ@2,EQ,@LASTPUN1,:SET1)
       WT(%SET_DIFF)
       JP(:CONT11)

KBCIN  FN=DCE\DCE19.MDL              SZ=10618  L=0286  C=001  Help=F1
```

```
;SETTING TIMES                                               ;APPENDIX 3.7.1

  %SET_SAMEB=(0:06:00.00)              ;SET SAME BEHRENS
  %SET_SAMEH=(0:06:00.00)              ;SET SAME HAMMERLE
  %CUT_SAME=(0:00:05.00)               ;SET SAME SALVAGNINI
  %SET_DIFFB=(0:18:00.00)              ;SET DIFFERENT BEHRENS
  %SET_DIFFH=(0:18:00.00)              ;SET DIFFERENT HAMMERLE
  %CUT_DIFF=(0:00:05.00)               ;SET DIFFERENT SALVAGNINI

KBCIN  FN=DCE\DCE22.MDL                        SZ=12034  L=0060  C=075  Help=F1
```

```
;UTILISATION OF BUFFERS AND MACHINES
;
U=(1,BUF20,*B12)                       ;THE NAME LABELS HAVE BEEN CHANGED
U=(2,B-20M,*B13)

U=(3,BUF15,*B22)
U=(4,B-15M,*B23)

U=(5,BBUFL,*B32)
U=(6,B-LAS,*B33)

U=(7,HBUF1,*H12)
U=(8,HAME1,*H13)

U=(9,HBUF5,*H22)
U=(10,HAME5,*H23)

U=(11,HBU2/3,*H32)
U=(12,HAM2/3,*H33)

U=(13,HBUF4,*H42)
U=(14,HAME4,*H43)

U=(15,STORE,*STORE)
KBCIN  FN=DCE\DCE22.MDL                        SZ=12034  L=0151  C=075  Help=F1
```

```
  BL(!MOVE)
       MR(2,%MOVE)
       MA(XY(4,11),%MOVE)             ;THIS DISTANCE HAS CHANGED
       MR(1,%MOVE)                    ;DUE TO THE CHANGES MADE TO
  EL                                  ;THE OVERLAY

  BL(!STORE)
       MR(6,%MOVE)
       MR(14,%MOVE)
       WT(30)
  EL
```

```
KBCIN  FN=DCE\DCE22.MDL                    SZ=12034  L=0227  C=075  Help=F1
```

```
:ROUTE1
       MD(5,%MOVE)                    ;CHANGES IN DISTANCES DUE
       MR(1,%MOVE)                    ;TO CHANGES MADE TO THE
       MD(4,%MOVE)                    ;OVERLAY
       MR(1,%MOVE)
       MU(4,%MOVE)
       MR(16,%MOVE)
       JC(*B12,:CONT1)

:CONT1
       MA(*B12,%MOVE)
       IF(OBJ@2,EQ,@LASTPUN1,:SET1)
       WT(%SET_DIFFB)
       JP(:CONT11)

:SET1
       WT(%SET_SAMEB)

:CONT11
       TP(*B13)
       MA(*B13,%MOVE)
       SV(%PUNCH1,%%SCHED(#SPUN,OBJ@1))
       AO(%PUNCH1,*,@@SCHED(#SQTY,OBJ@1))
KBCIN  FN=DCE\DCE22.MDL                    SZ=12034  L=0308  C=075  Help=F1
```

```
    :CT/50                                              ;APPENDIX 3.8.1
     IF(@COUNTER,EQ,1,:A)
     IF(@COUNTER,EQ,2,:B)
     IF(@COUNTER,EQ,3,:C)
     IF(@COUNTER,EQ,4,:D)
     :A                            ;COUNTER THAT SETS THE
     SC(7,0)                       ;COLOURS OF THE JOBS
     JP(:N)                        ;MOVING AROUND THE
     :B                            ;OVERLAY
     SC(7,1)
     JP(:N)
     :C
     SC(7,2)
     JP(:N)
     :D
     SC(7,4)
     SV(@COUNTER,0)
     JP(:N)
     :N
      SV(%CUTTER,%%SCHED(#SCUT,OBJ@1))
      AO(%CUTTER,*,@@SCHED(#SQTY,OBJ@1))
      WT(%CUTTER)
      SV(@LASTCUT,OBJ@2)
  EL
KBCIN  FN=DCE\DCE23.MDL                     SZ=12417  L=0213  C=001  Help=F1
```

```
;END LOGIC
:END
     SV(@@SCHED(#SFLG,OBJ@1),3)              ;SET FLAG TO BATCH ENDED
     SV(%END,CLOCK)                          ;NEW END LOGIC ADDED
     AO(%END,/,10:00:00.00)
     AO(%END,*,10:00:00.00)
     AO(%END,+,10:00:00.00)
     WC(%END)
     PM(+MESSAGE,"MODEL ENDED")
     OX
     IF(@THISFAM,EQ,0,WAIT)
     ER

KBCIN  FN=DCE\DCE23.MDL                     SZ=12417  L=0582  C=001  Help=F1
```

```
                                                    ;APPENDIX 3.9.1
:ROUTE1
       MO(5,%MOVE)
       MR(5,%MOVE)
       JC(*B12,*B13,:CONT1)          ;POSITION CHECK ADDED TO
                                     ;ENSURE THAT CORRECT SETTING
:CONT1                               ;TIMES ARE INCURED
       TP(*B12,*B13)                 ;POSITION CHECK ADDED
       MA(*B12,%MOVE)
       IF(OBJ@2,EQ,@LASTPUN1,:SET1)
       WT(%SET_DIFF)
```

```
L=(1000)                          ;WORK IN PROGRESS
C=(0)                             ;CLOCK ACCURACY

O=(DCE\DCE22.OLY)                 ;OVERLAY TO BE USED

@COUNTER=(0)                      ;SET COUNTERS TO ZERO
@CUTTER=(0)
@THISFAM=(0)
@THISBAT=(0)
@THISSTR=(0)
@BESTBAT=(0)
@BESTPTY=(0)
@R1PROD=(0)
@LASTCUT=(0)
@LASTPUN1=(0)
@LASTPUN2=(0)
@LASTPUN3=(0)
@LASTBND1=(0)
@LASTBND2=(0)
@LASTBND3=(0)
@LASTBND4=(0)

#EOF_FLAG=(9999)                  ;SET END COUNTERS
#HIGH=(9999)

%MOVE=(0:00:10.00)                ;SET CONSTANT TIMES

%CUTTER=(0:00:00.00)
%PUNCH1=(0:00:00.00)
%PUNCH2=(0:00:00.00)
%PUNCH3=(0:00:00.00)
%BEND1=(0:00:00.00)
%BEND2=(0:00:00.00)
%BEND3=(0:00:00.00)
%BEND4=(0:00:00.00)

%SET_SAMEB=(0:06:00.00)
%SET_SAMEH=(0:06:00.00)
%CUT_SAME=(0:00:05.00)
%SET_DIFFB=(0:18:00.00)
%SET_DIFFH=(0:18:00.00)
%CUT_DIFF=(0:00:05.00)
                                  ;COUNTERS FOR REPORT ARRAY
%WORK0=(00:00:00.00)              ;NEW TIME COUNTER  ;SALV
%WORK1=(00:00:00.00)                                 ;B1
%WORK2=(00:00:00.00)                                 ;B2
%WORK3=(00:00:00.00)                                 ;B3
%WORK4=(00:00:00.00)                                 ;H1
%WORK5=(00:00:00.00)              ;NEW TIME COUNTER  ;H2
%WORK6=(00:00:00.00)              ;NEW TIME COUNTER  ;H3
%WORK7=(00:00:00.00)              ;NEW TIME COUNTER  ;H4
%WORK8=(00:00:00.00)                                 ;STORE

@@SCHED=(5,3000)                  ;SET UP ARRAY SIZE
#SFLG=(0)                         ;SET UP ARRAY CONTENTS
#SFAM=(1)                         ;DATA ARRAY
#SQTY=(2)                         ;FIGURES
#SRTE=(3)
#SPTY=(4)
#SASC=(5)

%%SCHED=(3,3000)                  ;DATA ARRAY
#SCUT=(1)                         ;ACTUAL TIMES
#SPUN=(2)
```

```
   #SBND=(3)

   %%PROD_TIMES=(3,3000)              ;REPORT ARRAY
   #PFLG=(0)                          ;ACTUAL PRODUCTION TIMES
   #PBEH=(1)
   #PHAM=(2)
   #PPROD=(3)

   *CUTTER0=(XY(2,4))                 ;SET START POSITIONS
   *CUTTER=(XY(0,0))
   *START2=(XY(2,6))

      *B32=(XY(26,6))                 ;BUFFER,MACHINE LOCATIONS
      *B33=(XY(28,6))                 ;BEHRENS
      *B34=(XY(30,6))

      *B22=(XY(26,11))
      *B23=(XY(28,11))
      *B24=(XY(30,11))

      *B12=(XY(26,16))
      *B13=(XY(28,16))
      *B14=(XY(30,16))

      *H42=(XY(42,4))                 ;HAMMERLE
      *H43=(XY(44,4))
      *H44=(XY(46,4))

      *H32=(XY(42,6))
      *H33=(XY(44,6))
      *H34=(XY(46,6))

      *H22=(XY(42,11))
      *H23=(XY(44,11))
      *H24=(XY(46,11))

      *H12=(XY(42,16))
      *H13=(XY(44,16))
      *H14=(XY(46,16))

      *STORE=(XY(58,11))              ;WELDING STORE
      *END=(XY(72,11))

U=(1,BUF20,*B12)                      ;UTILISATION
U=(2,B-2MM,*B13)                      ;BUFFERS,MACHINES

U=(3,BUF15,*B22)
U=(4,B-15M,*B23)

U=(5,BBUFL,*B32)
U=(6,B-LAS,*B33)

U=(7,HBUF1,*H12)
U=(8,HAME1,*H13)

U=(9,HBUF5,*H22)
U=(10,HAME5,*H23)

U=(11,HBU2/3,*H32)
U=(12,HAM2/3,*H33)

U=(13,HBUF4,*H42)
U=(14,HAME4,*H43)

U=(15,STORE,*STORE)
```

```
J=(1,I,1,0,0,0,1)                   ;SET JOB QUANTITIES
J=(2,A,2,0,0,0,3000)
                                    ;BEGIN INITIALISING
BR(1,XY(0,1),0)                     ;BEGIN ROUTE
   SV(@THISFAM,1)                   ;SET START TEST POINTERS
   SV(@THISSTR,1)

   OF(DCE\SALTYPWE.DAT)                 ;OPEN FILE
:R1/10                              ;LOAD DATA FROM FILE
   GD(@R1PROD)                      ;GET FIRST ENTRY IN LINE
   IF(@R1PROD,EQ,#EOF_FLAG,:R1/50)  ;QUIT IF END OF FLAG
   SV(@@SCHED(#SFLG,@R1PROD),1)     ;SET FLAG TO=1LINE USED
   GD(@@SCHED(#SFAM,@R1PROD))       ;LOAD REST OF LINE..
   GD(@@SCHED(#SQTY,@R1PROD))
   GD(@@SCHED(#SRTE,@R1PROD))
   GD(@@SCHED(#SPTY,@R1PROD))
   GD(%%SCHED(#SCUT,@R1PROD))
   GD(%%SCHED(#SPUN,@R1PROD))
   GD(%%SCHED(#SBND,@R1PROD))
   GD(@@SCHED(#SASC,@R1PROD))
   JP(:R1/10)                       ;NEXT LINE (EOF_FLAG STOPS LOOP)
:R1/50
ER                                  ;END OF ROUTE

BL(!CUTTER)                         ;FIND FREE CUTTER
   :CT/10
      SV(@CUTTER,0)                 ;SET LOOP COUNT TO ZERO
      SV(*CUTTER,*CUTTER0)          ;GET ZERO POSITION
   :CT/20
     AD(*CUTTER,+,320)              ;ADD 2 LINES(2LINES*80COLS*2BYTES)
     JC(*CUTTER,:CT/30)             ;JUMP IF POITION CLEAR
     IV(@CUTTER)                    ;ELSE INCREMENT LOOP COUNT
     IF(@CUTTER,LT,6,:CT/20)        ;TRY NEXT IF NOT TRIED ALL
     WE                             ;ELSE WAIT EVENT..
     JP(:CT/10)                     ;..AND RESTART
   :CT/30
     MA(*CUTTER,0)                  ;MOVE TO VARIABLE POSITION

                                    ;NEW LINE MOVED FROM MOVE LOGIC
     SV(OBJ%1,CLOCK)                ;SET OBJ%1 TO CLOCK VALUE
                                    ;FOR REPORT ARRAY
     IF(OBJ@2,EQ,@LASTCUT,:CT/40)   ;TEST IF THIS EQUAL TO LAST
     WT(%CUT_DIFF)                  ;IF NO,WAIT DIFF SET TIME
     IV(@COUNTER)                   ;INCREMENT COUNTER
     JP(:CT/50)                     ;CONTINUE
   :CT/40
      WT(%CUT_SAME)                 ;ELSE WAIT CUT SAME

   :CT/50
     SV(%CUTTER,%%SCHED(#SCUT,OBJ@1))     ;GET UNIT CUT TIME
     AD(%CUTTER,*,@@SCHED(#SQTY,OBJ@1))   ;CALC TOTAL TIME
     WT(%CUTTER)                          ;WAIT CUT TIME
     SV(@LASTCUT,OBJ@2)                   ;SAVE LAST FAMILY TYPE
EL

BL(!MOVE)                           ;RELEASE LOGIC
     MR(2,%MOVE)
     MA(XY(4,11),%MOVE)
     MR(1,%MOVE)
EL

BL(!STORE)                          ;BEGIN LINK
     MR(6,%MOVE)
     MR(14,%MOVE)
```

```
        WT(30)
    EL                                  ;END LINK

    BR(2,XY(0,2),0)                     ;BEGIN JOB
    :R2/10                              ;RELEASE LOGIC
        SV(@BESTPTY,#HIGH)              ;SET BEST PRIORITY TO HIGH
        SV(@BESTBAT,0)                  ;ASSUME NO BEST LINE
        SV(@THISBAT,@THISSTR)           ;GET START BATCH OF CURRENT FAMILY
        DV(@THISBAT)                    ;DECREMENT TO MAKE NICE TEST LOOP

    :R2/20
        IV(@THISBAT)                    ;INCREMENT TO NEXT BATCH
        IF(@@SCHED(#SFAM,@THISBAT),NE,@THISFAM,:R2/50) ;QUIT LOOP IF
                                        ;THIS BAT NOT THIS FAMILY
        IF(@@SCHED(#SFLG,@THISBAT),NE,1,:R2/20)        ;NEXT IF THIS BAT
                                                       ;HAS BEEN TAKEN
        IF(@@SCHED(#SQTY,@THISBAT),EQ,0,:R2/20)        ;NEXT IF THIS QTY ZERO
        IF(@@SCHED(#SPTY,@THISBAT),GE,@BESTPTY,:R2/20) ;NEXT IF THIS PRIORITY
                                                       ;HIGHER THAN BEST
        SV(@BESTPTY,@@SCHED(#SPTY,@THISBAT))           ;ELSE GET NEW BEST
        SV(@BESTBAT,@THISBAT)                          ;BATCH
        JP(:R2/20)

    :R2/50
        IF(@BESTBAT,GT,0,:R2/100)              ;JUMP IF VALID BEST BATCH
        SV(@THISFAM,@@SCHED(#SFAM,@THISBAT))   ;ELSE GET NEXT FAMILY
                ;(THIS BAT ENDS POINTING TO NEXT START)
        IF(@THISFAM,EQ,0,WAIT)                 ;STOP FLOW IF NO NEXT FAMILY
        SV(@THISSTR,@THISBAT)                  ;SAVE START BATCH FOR FAMILY
        JP(:R2/10)

    :R2/100
        SV(OBJ@1,@BESTBAT)                  ;SET BATCH
        SV(@@SCHED(#SFLG,OBJ@1),2)          ;SET FLAG TO BATCH STARTED
        SV(OBJ@2,@@SCHED(#SFAM,OBJ@1))      ;GET FAMILY
        SV(OBJ@3,@@SCHED(#SQTY,OBJ@1))      ;GET QTY
        SV(OBJ@4,@@SCHED(#SRTE,OBJ@1))      ;GET ROUTE
        SV(OBJ@ID,@@SCHED(#SASC,OBJ@1))     ;GET ASCII VALUE OF ID


;MOVE LOGIC
;==========

        MA(#START2,0)                   ;MOVE ABSOLUTE
        LK(!CUTTER)                     ;LINK
                                        ;IMMEDIATELY AFTER CUTTING
        SV(%WORKO,CLOCK)                          ;NEW LINE
        AO(%WORKO,-,OBJ%1)                        ;NEW LINE
        SV(OBJ%2,%WORKO)                          ;NEW LINE
      . SV(%%PROD_TIMES(#PFLG,OBJ@1),%WORKO)      ;WRITE #PFLG TO
                                                  ;REPORT ARRAY
;       SV(%%PROD_TIMES(#PFLG,OBJ@1),OBJ%1)       IF WANT TIME TO START
;                                                 IMMEDIATELY AT SHEAR

        LK(!MOVE)                       ;LINK TO INITIAL MOVEMENT

        IF(OBJ@4,EQ,1,:ROUTE1)          ;DESIGNATE JOB ROUTES
        IF(OBJ@4,EQ,2,:ROUTE2)
        IF(OBJ@4,EQ,3,:ROUTE3)
        IF(OBJ@4,EQ,4,:ROUTE3)
        IF(OBJ@4,EQ,5,:ROUTE2)
        IF(OBJ@4,EQ,6,:ROUTE3)
        IF(OBJ@4,EQ,7,:ROUTE7)
        IF(OBJ@4,EQ,8,:ROUTE6)
        IF(OBJ@4,EQ,9,:ROUTE3)
```

```
:ROUTE1
        MD(5,%MOVE)                     ;MOVE DOWN
        MR(1,%MOVE)                     ;MOVE RIGHT
        MD(4,%MOVE)
        MR(1,%MOVE)
        MU(4,%MOVE)                     ;MOVE UP
        MR(16,%MOVE)
        JC(*B12,*B13,:CONT1)            ;JUMP CONDITIONAL

:CONT1
        TP(*B12,*B13)                   ;TEST POSITIONS
        MA(*B12,%MOVE)
        IF(OBJ@2,EQ,@LASTPUN1,:SET1)    ;IF THEN
        WT(%SET_DIFFB)                  ;WAIT SETTING TIME
        JP(:CONT11)                     ;JUMP

:SET1
        WT(%SET_SAMEB)

:CONT11
        MA(*B13,%MOVE)
        SV(%PUNCH1,%%SCHED(#SPUN,OBJ@1))
        AO(%PUNCH1,*,@@SCHED(#SQTY,OBJ@1))  ;ARITHMETIC OPERATION
        WT(%PUNCH1)
        SV(@LASTPUN1,OBJ@2)
        JP(:CONT12)

:CONT12
        TP(*B14)
        MA(*B14,%MOVE)
        SV(%WORK1,CLOCK)                ;SET %WORK1 TO CLOCK VALUE
        AO(%WORK1,-,OBJ%1)              ;%WORK1 MINUS START TIME AFTER
        SV(OBJ%2,%WORK1)                ;SHEAR
        SV(%%PROD_TIMES(#PBEH,OBJ@1),%WORK1)    ;WRITE #PBEH TO
        MR(7,%MOVE)                             ;REPORT ARRAY
        JC(*H12,*H13,:CONT13)

:CONT13
        TP(*H12,*H13)
        MA(*H12,%MOVE)
        IF(OBJ@2,EQ,@LASTBND1,:SET11)
        WT(%SET_DIFFH)
        JP(:CONT14)

:SET11
        WT(%SET_SAMEH)

:CONT14
        MA(*H13,%MOVE)
        SV(%BEND1,%%SCHED(#SBND,OBJ@1))
        AO(%BEND1,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND1)
        SV(@LASTBND1,OBJ@2)
        MA(*H14,%MOVE)
        SV(%WORK4,CLOCK)
        AO(%WORK4,-,OBJ%1)
        SV(OBJ%2,%WORK4)
        SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK4)    ;WRITE #PHAM TO
        MR(6,%MOVE)                             ;REPORT ARRAY
        MU(5,%MOVE)
        LK(!STORE)
        JP(:END)

:ROUTE2
        MR(1,%MOVE)
        MD(4,%MOVE)
```

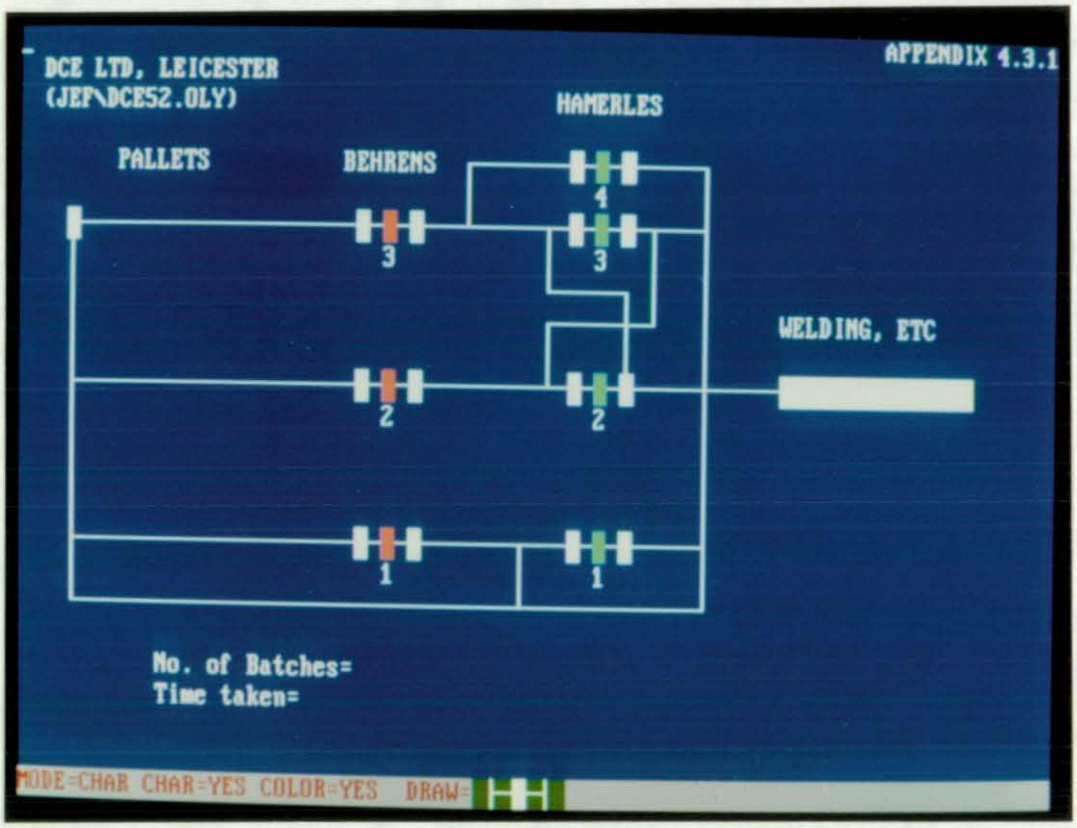
```
        MR(1,%MOVE)
        MU(4,%MOVE)
        MR(16,%MOVE)
        JC(*B22,*B23,:CONT2)

:CONT2
        TP(*B22,*B23)
        MA(*B22,%MOVE)
        IF(OBJ@2,EQ,@LASTPUN2,:SET2)
        WT(%SET_DIFFB)
        JP(:CONT21)

:SET2
        WT(%SET_SAMEB)

:CONT21
        MA(*B23,%MOVE)
        SV(%PUNCH2,%%SCHED(#SPUN,OBJ@1))
        AO(%PUNCH2,*,@@SCHED(#SQTY,OBJ@1))
        WT(%PUNCH2)
        SV(@LASTPUN2,OBJ@2)

:CONT22
        TP(*B24)
        MA(*B24,%MOVE)
        SV(%WORK2,CLOCK)
        AO(%WORK2,-,OBJ%1)
        SV(OBJ%2,%WORK2)
        SV(%%PROD_TIMES(#PBEH,OBJ@1),%WORK2)      ;WRITE #PBEH TO
        MR(7,%MOVE)                               ;REPORT ARRAY
        JC(*H22,*H23,:CONT23)

:CONT23
        TP(*H22,*H23)
        MA(*H22,%MOVE)
        IF(OBJ@2,EQ,@LASTBND2,:SET21)
        WT(%SET_DIFFH)
        JP(:CONT24)

:SET21
        WT(%SET_SAMEH)

:CONT24
        MA(*H23,%MOVE)
        SV(%BEND2,%%SCHED(#SBND,OBJ@1))
        AO(%BEND2,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND2)
        SV(@LASTBND2,OBJ@2)
        MA(*H24,%MOVE)

        IF(OBJ@4,EQ,5,:ROUTE5)
        IF(OBJ@4,EQ,9,:ROUTE9)

        SV(%WORK5,CLOCK)
        AO(%WORK5,-,OBJ%1)
        SV(OBJ%2,%WORK5)
        SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK5)      ;WRITE #PHAM TO
        MR(5,%MOVE)                               ;REPORT ARRAY
        LK(!STORE)
        JP(:END)

:ROUTE3                           ;MOVEMENT TO BEHRENS
        MU(5,%MOVE)
        MR(1,%MOVE)
        MD(4,%MOVE)
```

```
        MR(1,%MOVE)
        MU(4,%MOVE)
        MR(16,%MOVE)
        JC(*B32,*B33,:CONT3)
                                        ;BEHRENS OPERATION
:CONT3
        TP(*B32,*B33)
        MA(*B32,%MOVE)
        IF(OBJ@2,EQ,@LASTPUN3,:SET3)
        WT(%SET_DIFFB)
        JP(:CONT31)

:SET3
        WT(%SET_SAMEB)

:CONT31
        MA(*B33,%MOVE)
        SV(%PUNCH3,%%SCHED(#SPUN,OBJ@1))
        AO(%PUNCH3,*,@@SCHED(#SQTY,OBJ@1))
        WT(%PUNCH3)
        SV(@LASTPUN3,OBJ@2)

:CONT32
        TP(*B34)
        MA(*B34,%MOVE)
        SV(%WORK3,CLOCK)
        AO(%WORK3,-,OBJ%1)
        SV(OBJ%2,%WORK3)
        SV(%%PROD_TIMES(#PBEH,OBJ@1),%WORK3)     ;WRITE #PBEH TO
                                                 ;REPORT ARRAY
        IF(OBJ@4,EQ,4,:ROUTE4)

        MR(7,%MOVE)
        JC(*H32,*H33,:CONT33)

:CONT33                                 ;HAMMERLE OPERATION
        TP(*H32,*H33)
        MA(*H32,%MOVE)
        IF(OBJ@2,EQ,@LASTBND3,:SET31)
        WT(%SET_DIFFH)
        JP(:CONT34)

:SET31
        WT(%SET_SAMEH)

:CONT34
        MA(*H33,%MOVE)
        SV(%BEND3,%%SCHED(#SBND,OBJ@1))
        AO(%BEND3,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND3)
        SV(@LASTBND3,OBJ@2)
        MA(*H34,%MOVE)

        IF(OBJ@4,EQ,6,:ROUTE6)
        IF(OBJ@4,EQ,9,:ROUTE6)

        SV(%WORK6,CLOCK)
        AO(%WORK6,-,OBJ%1)
        SV(OBJ%2,%WORK6)
        SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK6)     ;WRITE #PHAM TO
        MR(6,%MOVE)                              ;REPORT ARRAY
        MD(5,%MOVE)
        LK(!STORE)
        JP(:END)
```

```
:ROUTE4
        MR(8,%MOVE)
        MU(2,%MOVE)
        MR(1,%MOVE)
        JC(*H42,*H43,:CONT43)

:CONT43
        TP(*H42,*H43)
        MA(*H42,%MOVE)
        IF(OBJ@2,EQ,@LASTBND4,:SET41)
        WT(%SET_DIFFH)
        JP(:CONT44)

:SET41
        WT(%SET_SAMEH)

:CONT44
        MA(*H43,%MOVE)
        SV(%BEND4,%%SCHED(#SBND,OBJ@1))
        AO(%BEND4,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND4)
        SV(@LASTBND4,OBJ@2)
        MA(*H44,%MOVE)
        SV(%WORK7,CLOCK)
        AO(%WORK7,-,OBJ%1)
        SV(OBJ%2,%WORK7)
        SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK7)     ;WRITE #PHAM TIME TO
        MR(6,%MOVE)                              ;REPORT ARRAY
        MD(7,%MOVE)
        LK(!STORE)
        JP(:END)

:ROUTE5
        MU(3,%MOVE)
        ML(6,%MOVE)
        MU(2,%MOVE)
        JP(:CONT33)

:ROUTE6
        MR(2,%MOVE)
        MD(3,%MOVE)
        ML(8,%MOVE)
        MD(2,%MOVE)
        JP(:CONT23)

:ROUTE7
        MD(10,%MOVE)
        MR(33,%MOVE)
        MU(5,%MOVE)
        JP(:CONT13)

:ROUTE8                             ;NO MACHINING
        MD(10,%MOVE)
        MR(47,%MOVE)
        MU(10,%MOVE)
;       SV(%WORK8,CLOCK)            ;DELETED THE FOLLOWING LINES
;       AO(%WORK8,-,OBJ%1)          ;AS NOT REQUIRED
;       SV(OBJ%2,%WORK8)
;       SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK8)     ;WRITE #PHAM TIME TO
        LK(!STORE)                               ;REPORT ARRAY
        JP(:END)

:ROUTE9
       MU(3,%MOVE)
       ML(6,%MOVE)
```

```
      MU(2,%MOVE)
      TP(*H32,*H33)
      MA(*H32,%MOVE)
      IF(OBJ@2,EQ,@LASTBND3,:SET51)
      WT(%SET_DIFFH)
      JP(:CONT54)
:SET51
      WT(%SET_SAMEH)

:CONT54
      MA(*H33,%MOVE)
      SV(%BEND3,%%SCHED(#SBND,OBJ@1))
      AO(%BEND3,*,@@SCHED(#SQTY,OBJ@1))
      WT(%BEND3)
      SV(@LASTBND3,OBJ@2)
      MA(*H34,%MOVE)
      SV(%WORK6,CLOCK)
      AO(%WORK6,-,OBJ%1)
      SV(OBJ%2,%WORK6)
      SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK6)      ;WTITE #PHAM TO
      MR(6,%MOVE)                               ;REPORT ARRAY
      MD(5,%MOVE)
      LK(!STORE)
      JP(:END)

;END LOGIC
:END
      SV(%WORK8,CLOCK)                          ;NEW LOGIC
      AO(%WORK8,-,OBJ%1)                        ;FOR OPERATION TIMES
      SV(OBJ%2,%WORK8)
      SV(%%PROD_TIMES(#PPROD,OBJ@1),%WORK8)     ;WRITE #PPROD TO
                                                ;REPORT ARRAY
      SV(@@SCHED(#SFLG,OBJ@1),3)                ;SET FLAG TO BATCH ENDED
      ER
```

APPENDIX 4.0.1.



OVERLAY USED IN JEF\DCE52.MDL
AND JEF\DCE54.MDL PROGRAMS

```
:ROUTE1
        MD(10,%MOVE)
        MR(20,%MOVE)
        JC(*B12,:CONT1)

:CONT1
        MA(*B12,%MOVE)

        IF(OBJ@3,EQ,@LASTPUN1,:SET1)
;       IF(OBJ@2,EQ,@LASTPUN1,:SET1)   ;HAS BEEN CHANGED
```

```
KBCIN  FN=JEF\DCE52.MDL                    SZ=09921  L=0222  C=075  Help=F1
```

```
:CONT11
        TP(*B13)
        MA(*B13,%MOVE)
        SV(%PUNCH1,%%SCHED(#SPUN,OBJ@1))
        AD(%PUNCH1,*,@@SCHED(#SQTY,OBJ@1))
        WT(%PUNCH1)

        SV(@LASTPUN1,OBJ@3)
;       SV(@LASTPUN1,OBJ@2)            ;HAS BEEN CHANGED
```

```
KBCIN  FN=JEF\DCE52.MDL                    SZ=09921  L=0274  C=075  Help=F1
:CONT24
        MA(*H23,%MOVE)
        SV(%BEND2,%%SCHED(#SBND,OBJ@1))
        AD(%BEND2,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND2)
        SV(@LASTBND2,OBJ@3)
        MA(*H24,%MOVE)

        IF(OBJ@5,EQ,5,:ROUTE5)
;       IF(OBJ@4,EQ,4,:ROUTE4)         ;HAS BEEN CHANGED
```

```
;JEF\DCE54.MDL                                   ;PROGRAM USES NEW OVERLAY
                                                 ;RUNS ROUTES TOGETHER
    L=(10)          ;WORK IN PROGRESS            ;STARTING AT THE BEHRENS
    C=(0)           ;CLOCK ACCURACY

    O=(JEF\DCE52.OLY)                            ;OVERLAY TO BE USED
                                                 ;SET COUNTERS TO ZERO
    @COUNT=(0)                                   ;NEW, COUNTER(VARIABLE VALUES)
    @THISFAM=(0)
    @THISBAT=(0)
    @THISSTR=(0)
    @BESTBAT=(0)
    @BESTPTY=(0)
    @R1PROD=(0)
    @LASTPUN1=(0)
    @LASTPUN2=(0)
    @LASTPUN3=(0)
    @LASTBND1=(0)
    @LASTBND2=(0)
    @LASTBND3=(0)
    @LASTBND4=(0)
    @RTE=(0)
    @ORD=(0)
    @START=(0)

    #EOF_FLAG=(9999)                             ;SET END COUNTERS
    #HIGH=(9999)                                 ;(CONSTANT VALUES)

    %MOVE=(0:00:10.00)                           ;SET CONSTANT TIMES
                                                 ;(CLOCK VALUES)
    %PUNCH1=(0:00:00.0)
    %PUNCH2=(0:00:00.0)
    %PUNCH3=(0:00:00.0)
    %BEND1=(0:00:00.0)
    %BEND2=(0:00:00.0)
    %BEND3=(0:00:00.0)
    %BEND4=(0:00:00.0)

    %SET_SAME=(0:06:00.00)
    %SET_DIFF=(0:18:00.00)

;   %WORK0=(00:00:00.00)                         ;COUNTERS FOR REPORT ARRAY
    %WORK1=(00:00:00.00)                         ;%WORK0 NOT REQUIRED AS
    %WORK2=(00:00:00.00)                         ;THERE IS NO CUTTING TIME ALLOWED
    %WORK3=(00:00:00.00)
    %WORK4=(00:00:00.00)
    %WORK5=(00:00:00.00)
    %WORK6=(00:00:00.00)
    %WORK7=(00:00:00.00)
    %WORK8=(00:00:00.00)
                                                 ;SET UP ARRAY SIZE
    @@SCHED=(6,3000)                             ;SET UP ARRAY CONTENTS
    #SFLG=(0)
    #PART=(1)
    #SFAM=(2)
    #SQTY=(3)
    #SRTE=(4)
    #SPTY=(5)
    #SASC=(6)

    %%SCHED=(3,3000)
    #SCUT=(1)
    #SPUN=(2)
```

```
   $SBND=(3)

   %%PROD_TIMES=(3,3000)              ;REPORT ARRAY
   $PFLG=(0)                          ;JOB COMPLETION TIMES
   $PBEH=(1)
   $PHAM=(2)
   $PPROD=(3)

   *START2=(XY(4,6))                  ;START POSITION SET

        *B32=(XY(26,6))               ;BUFFERS,MACHINES
        *B33=(XY(28,6))               ;BEHRENS LOCATIONS
        *B34=(XY(30,6))

        *B22=(XY(26,11))
        *B23=(XY(28,11))
        *B24=(XY(30,11))

        *B12=(XY(26,16))
        *B13=(XY(28,16))
        *B14=(XY(30,16))

        *H42=(XY(42,4))               ;HAMMERLE LOCATIONS
        *H43=(XY(44,4))
        *H44=(XY(46,4))

        *H32=(XY(42,6))
        *H33=(XY(44,6))
        *H34=(XY(46,6))

        *H22=(XY(42,11))
        *H23=(XY(44,11))
        *H24=(XY(46,11))

        *H12=(XY(42,16))
        *H13=(XY(44,16))
        *H14=(XY(46,16))

        *STORE=(XY(58,11))            ;WELDING STORE LOCATIONS
        *END=(XY(72,11))

        *BATCH=(XY(26,20))            ;NEW, PRINT MESSAGES,BATCH NUMBERS
        *TIME=(XY(22,21))             ;NEW, TIME TAKEN

U=(1,BBUF1,*B12)                      ;UTILISATION
U=(2,BEHR1,*B13)                      ;BUFFERS,MACHINES

U=(3,BBUF2,*B22)
U=(4,BEHR2,*B23)

U=(5,BBUF3,*B32)
U=(6,BEHR3,*B33)

U=(7,HBUF1,*H12)
U=(8,HAME1,*H13)

U=(9,HBUF2,*H22)
U=(10,HAME2,*H23)

U=(11,HBUF3,*H32)
U=(12,HAME3,*H33)

U=(13,HBUF4,*H42)
U=(14,HAME4,*H43)

U=(15,STORE,*STORE)
```

```
   J=(1,I,1,0,0,0,1)                    ;SET JOB QUANTITIES
   J=(2,A,2,0,0,1,3000)

                                        ;BEGIN INITIALISING
   BR(1,XY(0,1),0)                      ;BEGIN ROUTE
      SV(@THISFAM,1)                    ;SET START TEST POINTERS
      SV(@THISSTR,1)
      SV(@RTE,8)
                                        ;OPEN ARRAY FILE NAME
      OF(JEF\SCHED3.DAT)               ;TO BE READ INTO THE PROGRAM
;     I=(A.MIF)                         ;NEW, INCLUDE  FILE A.MIF
;     I=(B.MIF)                         ;NEW, INCLUDE  FILE B.MIF

   :R1/10                               ;LOAD DATA FROM FILE
      GD(@R1PROD)                       ;GET FIRST ENTRY IN LINE
      IF(@R1PROD,EQ,#EOF_FLAG,:R1/50)   ;QUIT IF END OF FLAG
      SV(@@SCHED(#SFLG,@R1PROD),1)      ;SET FLAG TO=1LINE USED
      GD(@@SCHED(#PART,@R1PROD))        ;LOAD REST OF LINE..
      GD(@@SCHED(#SFAM,@R1PROD))
      GD(@@SCHED(#SQTY,@R1PROD))
      GD(@@SCHED(#SRTE,@R1PROD))
      GD(@@SCHED(#SPTY,@R1PROD))
      GD(XXSCHED(#SCUT,@R1PROD))
      GD(XXSCHED(#SPUN,@R1PROD))
      GD(XXSCHED(#SBND,@R1PROD))
      GD(@@SCHED(#SASC,@R1PROD))
      JP(:R1/10)                        ;NEXT LINE (EOF_FLAG STOPS LOOP)
   :R1/50
;   CF(DAT)                             ;NEW, CLOSE FILE
    OF(=.RPT)                           ;NEW, OPEN FILE
    PM(F,XXPROD_TIMES)                  ;NEW, PRINT MESSAGE IN FILE
   ER                                   ;END OF ROUTE

;  BL(!MOVE)                            ;DELETED LINK MOVE
;        MR(5,XMOVE)
;        MA(XY(15,11),XMOVE)
;        MR(4,XMOVE)
;  EL

   BL(!STORE)                           ;BEGIN LINK STORE
        MR(6,XMOVE)
        MR(14,XMOVE)
   EL                                   ;END LINK

   BR(2,XY(0,2),0)                      ;BEGIN JOB

                                        ;RELEASE LOGIC
      SV(@START,@RTE)                   ;SET VALUE @START TO @RTE
   :LAB5                                ;JUMP LABEL
      IV(@RTE)                          ;INCREMENT COUNT
      IF(@RTE,LE,8,:LAB10)              ;IF..THEN
      SV(@RTE,1)                        ;SET VALUE
   :LAB10
      SV(@ORD,0)                        ;SET VALUE
   :LAB20
      IV(@ORD)                                      ;INCREMENT COUNT
      IF(@@SCHED(#SFLG,@ORD),EQ,0,:LAB50)           ;IF..THEN
      IF(@@SCHED(#SFLG,@ORD),NE,1,:LAB20)           ;IF..THEN
      IF(@@SCHED(#SRTE,@ORD),NE,@RTE,:LAB20)        ;IF..THEN
      JP(:LAB100)                                   ;JUMP TO :LAB100
   :LAB50
      IF(@RTE,NE,@START,:LAB5)
      IF(CLOCK,GT,0,WAIT)
   :LAB100
```

```
     SV(@@SCHED(#SFLG,@ORD),2)                ;SET FLAG
     SV(OBJ@1,@ORD)                           ;SET ORDER
     SV(OBJ@2,@@SCHED(#PART,@ORD))            ;GET PART
     SV(OBJ@3,@@SCHED(#SFAM,@ORD))            ;GET FAMILY
     SV(OBJ@4,@@SCHED(#SQTY,@ORD))            ;GET QUANTITY
     SV(OBJ@5,@@SCHED(#SRTE,@ORD))            ;GET ROUTE
     SV(OBJ@ID,@@SCHED(#SASC,@ORD))           ;GET ASCII VALUE OF ID

;MOVE LOGIC
;==========

        MA(*START2,0)                   ;MOVE ABSOLUTE
        IV(@COUNT)                      ;NEW, INCREMENT COUNT
        SV(OBJ%1,CLOCK)
;       SV(%WORK0,CLOCK)                ;DELETED AS TIME IS
;       AO(%WORK0,-,OBJ%1)              ;ALWAYS ZERO
;       SV(OBJ%2,%WORK0)                ;DELETED AS NOT REQUIRED
        SV(%%PROD_TIMES(#PFLG,OBJ@1),CLOCK)      ;TIME SET TO JOB START
;       SV(%%PROD_TIMES(#PFLG,OBJ@1),OBJ%1)      ;DELETED, IF WANT TO START
                                                 ;IMMEDIATELY AT SHEAR
        IF(OBJ@5,EQ,1,:ROUTE1)          ;DESIGNATE JOB ROUTES
        IF(OBJ@5,EQ,2,:ROUTE2)          ;ONLY 8 JOB ROUTES
        IF(OBJ@5,EQ,3,:ROUTE3)
        IF(OBJ@5,EQ,4,:ROUTE3)
        IF(OBJ@5,EQ,5,:ROUTE2)
        IF(OBJ@5,EQ,6,:ROUTE3)
        IF(OBJ@5,EQ,7,:ROUTE7)
        IF(OBJ@5,EQ,8,:ROUTE8)

:ROUTE1
        MD(10,%MOVE)                    ;MOVE DOWN
        MR(20,%MOVE)                    ;MOVE RIGHT
        JC(*B12,:CONT1)                 ;JUMP CONITIONAL

:CONT1
        MA(*B12,%MOVE)
        IF(OBJ@3,EQ,@LASTPUN1,:SET1)    ;IF..THEN
        WT(%SET_DIFF)                   ;WAIT SET TIME
        JP(:CONT11)

:SET1
        WT(%SET_SAME)

:CONT11
        TP(*B13)
        MA(*B13,%MOVE)
        SV(%PUNCH1,%%SCHED(#SPUN,OBJ@1))
        AO(%PUNCH1,*,@@SCHED(#SQTY,OBJ@1))     ;ARITHMETIC OPERATION
        WT(%PUNCH1)
        SV(@LASTPUN1,OBJ@3)
        JP(:CONT12)

:CONT12
        TP(*B14)
        MA(*B14,%MOVE)
        SV(%WORK1,CLOCK)
        AO(%WORK1,-,OBJ%1)
;       SV(OBJ@2,%WORK1)                ;DELETED AS NOT REQUIRED
        SV(%%PROD_TIMES(#PBEH,OBJ@1),%WORK1)
        PM(F,BEHRENS-1)                 ;NEW, PRINT MESSAGE TO FILE
        PV(F,%WORK1)                    ;NEW, PRINT VALUE TO FILE

        JC(*H12,:CONT13)
```


```
:CONT13
```

```
        MA(*H12,%MOVE)
        IF(OBJ@3,EQ,@LASTBND1,:SET11)
        WT(%SET_DIFF)
        JP(:CONT14)

:SET11
        WT(%SET_SAME)

:CONT14
        MA(*H13,%MOVE)
        SV(%BEND1,%%SCHED(#SBND,OBJ@1))
        AO(%BEND1,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND1)
        SV(@LASTBND1,OBJ@3)
        MA(*H14,%MOVE)
        SV(%WORK4,CLOCK)
        AO(%WORK4,-,OBJ%1)
;       SV(OBJ%2,%WORK4)                ;DELETED AS NOT REQUIRED
        SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK4)
        PM(F,HAMMERLE-1)                ;NEW, PRINT MESSAGE TO FILE
        PV(F,%WORK4)                    ;NEW, PRINT VALUE TO FILE
        MR(6,%MOVE)
        MU(5,%MOVE)
        LK(!STORE)
        JP(:END)

:ROUTE2
        MD(5,%MOVE)
        MR(20,%MOVE)
        JC(*B22,:CONT2)

:CONT2
        MA(*B22,%MOVE)
        IF(OBJ@3,EQ,@LASTPUN2,:SET2)
        WT(%SET_DIFF)
        JP(:CONT21)

:SET2
        WT(%SET_SAME)

:CONT21
        TP(*B23)
        MA(*B23,%MOVE)
        SV(%PUNCH2,%%SCHED(#SPUN,OBJ@1))
        AO(%PUNCH2,*,@@SCHED(#SQTY,OBJ@1))
        WT(%PUNCH2)
        SV(@LASTPUN2,OBJ@3)
        JP(:CONT22)

:CONT22
        TP(*B24)
        MA(*B24,%MOVE)
        SV(%WORK2,CLOCK)
        AO(%WORK2,-,OBJ%1)
;       SV(OBJ%2,%WORK2)                ;DELETED AS NOT REQUIRED
        SV(%%PROD_TIMES(#PBEH,OBJ@1),%WORK2)
        PM(F,BEHRENS-2)                 ;NEW, PRINT MESSAGE TO FILE
        PV(F,%WORK2)                    ;NEW, PRINT VALUE TO FILE
        JC(*H22,:CONT23)

:CONT23
        MA(*H22,%MOVE)
        IF(OBJ@3,EQ,@LASTBND2,:SET21)
        WT(%SET_DIFF)
        JP(:CONT24)
```

```
:SET21
        WT(%SET_SAME)

:CONT24
        MA(*H23,%MOVE)
        SV(%BEND2,%%SCHED(#SBND,OBJ@1))
        AO(%BEND2,+,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND2)
        SV(@LASTBND2,OBJ@3)
        MA(*H24,%MOVE)

        IF(OBJ@5,EQ,5,:ROUTE5)

        SV(%WORK5,CLOCK)
        AO(%WORK5,-,OBJ%1)
;       SV(OBJ%2,%WORK5)              ;DELETED AS NOT REQUIRED
        SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK5)
        PM(F,HAMMERLE-2)              ;NEW, PRINT MESSAGE TO FILE
        PV(F,%WORK5)                  ;NEW, PRINT VALUE TO FILE
        MR(5,%MOVE)
        LK(!STORE)
        JP(:END)

:ROUTE3
        MR(20,%MOVE)                  ;MOVEMENT TO BEHRENS
        JC(*B32,:CONT3)

:CONT3
        MA(*B32,%MOVE)                ;BEHRENS OPERATION
        IF(OBJ@3,EQ,@LASTPUN3,:SET3)
        WT(%SET_DIFF)
        JP(:CONT31)

:SET3
        WT(%SET_SAME)

:CONT31
        MA(*B33,%MOVE)
        SV(%PUNCH3,%%SCHED(#SPUN,OBJ@1))
        AO(%PUNCH3,+,@@SCHED(#SQTY,OBJ@1))
        WT(%PUNCH3)
        SV(@LASTPUN3,OBJ@3)

:CONT32
        TP(*B34)
        MA(*B34,%MOVE)
        SV(%WORK3,CLOCK)
        AO(%WORK3,-,OBJ%1)
;       SV(OBJ%2,%WORK3)              ;DELETED AS NOT REQUIRED
        SV(%%PROD_TIMES(#PBEH,OBJ@1),%WORK3)
        PM(F,BEHRENS-3)               ;NEW, PRINT MESSAGE TO FILE
        PV(F,%WORK3)                  ;NEW, PRINT VALUE TO FILE

        IF(OBJ@5,EQ,4,:ROUTE4)

        JC(*H32,:CONT33)

:CONT33
        MA(*H32,%MOVE)                ;HAMMERLE OPERATION
        IF(OBJ@3,EQ,@LASTBND3,:SET31)
        WT(%SET_DIFF)
        JP(:CONT34)
```



```
:SET31
```

```
        WT(%SET_SAME)

:CONT34
        MA(*H33,%MOVE)
        SV(%BEND3,%%SCHED(#SBND,OBJ@1))
        AO(%BEND3,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND3)
        SV(@LASTBND3,OBJ@3)
        MA(*H34,%MOVE)

        IF(OBJ@5,EQ,6,:ROUTE6)

        SV(%WORK6,CLOCK)
        AO(%WORK6,-,OBJ%1)
;       SV(OBJ%2,%WORK6)                ;DELETED AS NOT REQUIRED
        SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK6)
        PM(F,HAMMERLE-3)                ;NEW, PRINT MESSAGE TO FILE
        PV(F,%WORK6)                    ;NEW, PRINT VALUE TO FILE
        MR(6,%MOVE)
        MD(5,%MOVE)
        LK(!STORE)
        JP(:END)

:ROUTE4
        MR(4,%MOVE)
        MU(2,%MOVE)
        MR(6,%MOVE)                     ;MOVE RIGHT
        JC(*H42,:CONT43)

:CONT43
        MA(*H42,%MOVE)
        IF(OBJ@3,EQ,@LASTBND4,:SET41)
        WT(%SET_DIFF)
        JP(:CONT44)

:SET41
        WT(%SET_SAME)

:CONT44
        MA(*H43,%MOVE)
        SV(%BEND4,%%SCHED(#SBND,OBJ@1))
        AO(%BEND4,*,@@SCHED(#SQTY,OBJ@1))
        WT(%BEND4)
        SV(@LASTBND4,OBJ@3)
        MA(*H44,%MOVE)
        SV(%WORK7,CLOCK)
        AO(%WORK7,-,OBJ%1)
;       SV(OBJ%2,%WORK7)                ;DELETED AS NOT REQUIRED
        SV(%%PROD_TIMES(#PHAM,OBJ@1),%WORK7)
        PM(F,HAMMERLE-4)                ;NEW, PRINT MESSAGE TO FILE
        PV(F,%WORK7)                    ;NEW, PRINT VALUE TO FILE
        MR(6,%MOVE)
        MD(7,%MOVE)
        LK(!STORE)
        JP(:END)

:ROUTE5
        MU(3,%MOVE)
        ML(6,%MOVE)
        MU(2,%MOVE)
        JP(:CONT33)

:ROUTE6
        MR(2,%MOVE)
        MD(3,%MOVE)
```

```
        ML(8,%MOVE)
        MD(2,%MOVE)
        JP(:CONT23)

:ROUTE7
        MD(12,%MOVE)
        MR(34,%MOVE)
        MU(2,%MOVE)
        JP(:CONT13)

:ROUTE8                               ;NO MACHINING
        MD(12,%MOVE)
        MR(48,%MOVE)
        MU(7,%MOVE)
        PM(F,NO-MACHINING)            ;NEW, PRINT MESSAGE TO FILE
        PV(F,%WORK8)                  ;NEW, PRINT VALUE TO FILE
        LK(!STORE)
        JP(:END)

;END LOGIC
:END
        SV(%WORK8,CLOCK)
        AO(%WORK8,-,OBJ%1)
;       SV(OBJ%2,%WORK8)              ;DELETED AS NO LONGER REQUIRED
        SV(%%PROD_TIMES(#PPROD,OBJ@1),%WORK8)
        SV(@@SCHED(#SFLG,OBJ@1),3)    ;SET FLAG TO BATCH ENDED
        PV(*BATCH,@COUNT)             ;PRINT VALUE  BATCH QUANTITY
        PV(*TIME,CLOCK)               ;PRINT VALUE  TIME
        ER                            ;END OF ROUTE
```

PCModel Statistics File Template
(C) 1985, Simulation Software Systems

APPENDIX 4.2.2

******PRESS ALT-I TO READ IN STATISTICS FILE

Total Hours: 10 Total Tools: 16

	BBUF1	BEHR1	BBUF2	BEHR2	BBUF3	BEHR3	HBUF1	HAME1	HBUF2	HAME2	HBUF3	HAME3	HBUF4	HAME4	STORE	THRUPUT
MEAN	39.64	67.08	21.22	50.22	18.17	7.67	46.55	47.26	56.86	50.35	14.16	4.89	4.05	2.28	0.50	1.90
STD-D	47.66	43.42	38.05	45.01	36.36	17.18	41.51	44.64	44.85	46.28	15.79	6.63	8.73	6.83	0.59	2.21
MAX	100.00	100.00	100.00	100.00	94.16	55.83	100.00	100.00	100.00	100.00	40.55	20.00	27.50	22.77	1.94	7.00
MIN	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Hour	BBUF1	BEHR1	BBUF2	BEHR2	BBUF3	BEHR3	HBUF1	HAME1	HBUF2	HAME2	HBUF3	HAME3	HBUF4	HAME4	STORE	THRUPUT
1	91.66	61.38	92.77	62.50	94.16	20.55	30.27	8.61	0.00	0.00	30.27	0.27	13.05	0.00	0.83	4.00
2	100.00	100.00	100.00	100.00	87.50	55.83	10.27	1.66	41.11	8.61	40.55	20.00	27.50	22.77	0.83	3.00
3	100.00	100.00	19.44	100.00	0.00	0.27	2.22	6.94	100.00	80.55	30.27	8.61	0.00	0.00	0.83	3.00
4	100.00	100.00	0.00	100.00	0.00	0.00	48.33	71.94	100.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00
5	4.72	100.00	0.00	100.00	0.00	0.00	100.00	100.00	100.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00
6	0.00	100.00	0.00	39.72	0.00	0.00	100.00	100.00	100.00	100.00	30.27	11.38	0.00	0.00	0.27	1.00
7	0.00	100.00	0.00	0.00	0.00	0.00	100.00	100.00	100.00	100.00	0.00	0.00	0.00	0.00	0.27	1.00
8	0.00	9.44	0.00	0.00	0.00	0.00	74.44	83.47	27.44	14.38	10.27	8.61	0.00	0.00	1.94	7.00
9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

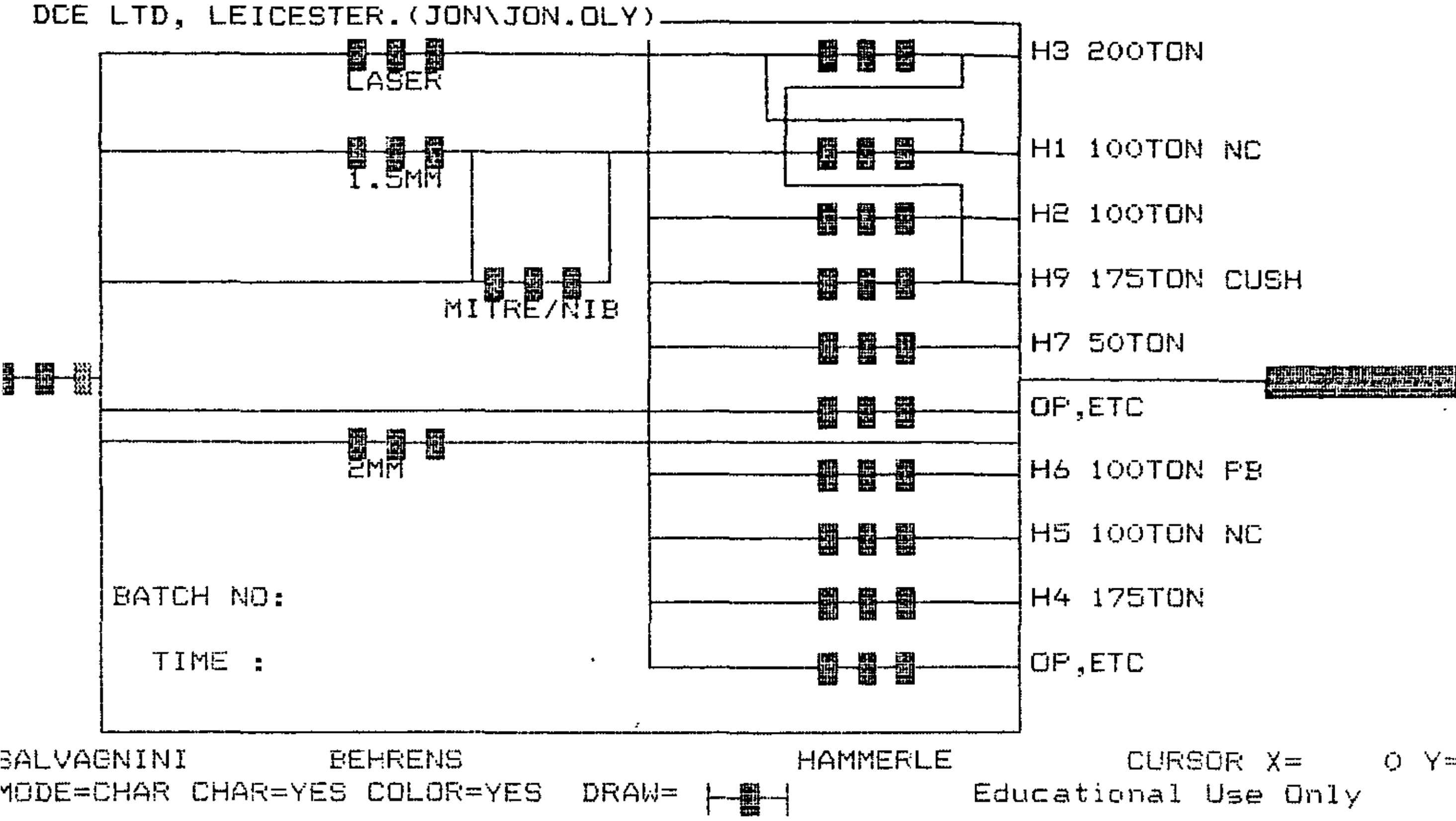
Statistics from PCModel session converted by PCMLOTUS

```
XXPROD_TIMES
NO-MACHINING
0000:00:00.00
BEHRENS-3
0000:21:50.00
HAMMERLE-1
0000:31:30.00
HAMMERLE-3
0000:40:20.00
BEHRENS-3
0000:50:00.00
BEHRENS-3
0001:05:30.00
HAMMERLE-4
0001:17:10.00
BEHRENS-3
0001:11:20.00
HAMMERLE-3
0001:24:30.00
BEHRENS-3
0001:07:30.00
BEHRENS-2
0001:53:30.00
HAMMERLE-2
0001:55:00.00
HAMMERLE-4
0001:22:40.00
BEHRENS-3
0000:37:30.00
BEHRENS-2
0002:03:20.00
HAMMERLE-1
0000:19:30.00
NO-MACHINING
0000:24:40.00
BEHRENS-2
0002:11:00.00
BEHRENS-1
0002:58:40.00
BEHRENS-1
0003:47:10.00
BEHRENS-1
0001:57:40.00
HAMMERLE-2
0005:24:00.00
BEHRENS-2
0004:38:20.00
HAMMERLE-3
0005:57:30.00
HAMMERLE-2
0006:57:38.00
HAMMERLE-2
0005:40:48.00
HAMMERLE-1
0007:05:50.00
BEHRENS-1
0004:56:20.00
HAMMERLE-3
0006:49:28.00
HAMMERLE-1
0007:38:00.00
HAMMERLE-1
0005:38:30.00
HAMMERLE-1
0005:41:45.00
```

APPENDIX 4.2.3.

APPENDIX 5.1

ROUTING OF JOBS IN THE JON DIRECTORY



S/BL/		10
	H3	11
	H1 (NC)	12
	H2	13
	H9	14
	H7	15
	OP 1.5mm	1
	H6	16
	H5 (NC)	17
	H4	18
	OP 2mm	2
	H3/H1/H3	3
	H2/H1	4
	H1/H3	5
	H9/H1	6
S/B1.5/		
	H1 (NC)	20
	H2	21
	H9	22
	H7	23
	OP 1.5	24
	NO FOLD	25
	MITRE/H1	26
MITRE		
	/H1	30
	/H2	31
	OP 1.5mm	40
	OP 2.0mm	41

S/B2mm/

H6	50
H5 (NC)	51
H4	53
OP2M	54

```
;JON\JON.MDL                       ;WORK OPERATED ON CONSECUTIVELY
                                   ;AS PER MOR COMMITTEE REQUEST
                                   ;DATE 14/11/89

L=(10)                             ;WORK IN PROGRESS
C=(0)                              ;CLOCK ACCURACY

O=(JON\JON.OLY)                    ;NAME OF OVERLAY

@COUNT=(0)                         ;SET COUNTERS TO ZERO
@CUTTER=(0)                        ;(VARIABLE VALUES)
@THISFAM=(0)
@THISBAT=(0)
@THISSTR=(0)
@BESTBAT=(0)
@BESTPTY=(0)
@R1PROD=(0)
@LASTCUT=(0)
@LASTPUN1=(0)
@LASTPUN2=(0)
@LASTPUN3=(0)
@LASTMITR=(0)
@LASTBND1=(0)
@LASTBND2=(0)
@LASTBND3=(0)
@LASTBND4=(0)
@LASTBND5=(0)
@LASTBND6=(0)
@LASTBND7=(0)
@LASTBND8=(0)
@LASTPRS1=(0)
@LASTPRS2=(0)
@RTE=(0)
@RTF=(0)
@PTY=(0)
@ORD=(0)
@START=(0)

@CU=(0)                            ;COUNTERS FOR JOBS
@BL=(0)                            ;JOBS PASSING THROUGH MACHINES
@B1=(0)
@B2=(0)
@M1=(0)
@M2=(0)
@L1=(0)
@L2=(0)
@L3=(0)
@L4=(0)
@L6=(0)
@H1=(0)
@H2=(0)
@H3=(0)
@H4=(0)
@H5=(0)
@H6=(0)
@H7=(0)
@H8=(0)
@O1=(0)
@O2=(0)

#EOF_FLAG=(9999)                   ;SET END COUNTERS
#HIGH=(9999)                       ;(CONSTANT VALUES)
```

```
%MOVE=(0:00:10.00)                  ;SET CONSTANT TIMES
                                    ;(CLOCK VALUES)
%CUTTER=(0:00:00.00)
%PUNCH1=(0:00:00.00)
%PUNCH2=(0:00:00.00)
%PUNCH3=(0:00:00.00)
%MITRE=(0:00:00.00)
%BEND1=(0:00:00.00)
%BEND2=(0:00:00.00)
%BEND3=(0:00:00.00)
%BEND4=(0:00:00.00)
%BEND5=(0:00:00.00)
%BEND6=(0:00:00.00)
%BEND7=(0:00:00.00)
%BEND8=(0:00:00.00)
%PRESS1=(0:00:00.00)
%PRESS2=(0:00:00.00)

                                    ;SETTING TIMES
    %CUT_SAME=(0:00:00.00)          ;SHEAR
    %CUT_DIFF=(0:00:00.00)

    %SET_DIFB=(0:21:00.00)          ;BEHRENS
    %SET_SAMB=(0:12:00.00)

    %SET_DIFM=(0:18:00.00)          ;MITRE
    %SET_SAMM=(0:12:00.00)

    %SET_DIFH=(0:28:00.00)          ;HAMMERLE
    %SET_SAMH=(0:12:00.00)

    %SET_DIFP=(0:18:00.00)          ;OP PRESS
    %SET_SAMP=(0:12:00.00)

                                    ;COUNTERS FOR REPORT ARRAY
 %WORK0=(00:00:00.00)               ;SAL
 %WORK1=(00:00:00.00)               ;BL
 %WORK2=(00:00:00.00)               ;B15
 %WORK3=(00:00:00.00)               ;MITRE
 %WORK4=(00:00:00.00)               ;LINE
 %WORK5=(00:00:00.00)               ;B2
 %WORK6=(00:00:00.00)               ;LINE
 %WORK7=(00:00:00.00)               ;H3
 %WORK8=(00:00:00.00)               ;H1
 %WORK9=(00:00:00.00)               ;H2
 %WORK10=(00:00:00.00)              ;H9
 %WORK11=(00:00:00.00)              ;H7
 %WORK12=(00:00:00.00)              ;OP15
 %WORK13=(00:00:00.00)              ;LINE
 %WORK14=(00:00:00.00)              ;H6
 %WORK15=(00:00:00.00)              ;H5
 %WORK16=(00:00:00.00)              ;H4
 %WORK17=(00:00:00.00)              ;OP2
 %WORK18=(00:00:00.00)              ;LINE
 %WORK19=(00:00:00.00)              ;STORE

##SCHED=(7,2000)                    ;SET UP ARRAY SIZE
#SFLG=(0)                           ;SET UP ARRAY CONTENTS
#PART=(1)
#SFAM=(2)
#SQTY=(3)
#SRTE=(4)
#SRTF=(5)
#SPTY=(6)
#SASC=(7)
```

```
XXSCHED=(3,2000)
#SCUT=(1)
#SPUN=(2)
#SBND=(3)

 XXPROD_TIMES=(5,2000)              ;REPORT ARRAY
 #PSAL=(0)                          ;JOB COMPLETION TIMES
 #P1ST=(1)
 #P2ND=(2)
 #P3RD=(3)
 #P4TH=(4)
 #PPROD=(5)

                                    ;M/C LOCATION
     *CUTTER0=(XY(1,4))             ;SET START POSITIONS
     *CUTTER=(XY(0,0))
     *START2=(XY(0,11))             ;START
                                    ;BEHRENS LOCATION
     *B11=(XY(15,1))
     *B12=(XY(18,1))
     *B13=(XY(20,1))                ;LASER
     *B14=(XY(22,1))

     *B21=(XY(15,4))
     *B22=(XY(18,4))
     *B23=(XY(20,4))                ;1_5MM
     *B24=(XY(22,4))

     *B31=(XY(15,13))
     *B32=(XY(18,13))
     *B33=(XY(20,13))               ;2_0MM
     *B34=(XY(22,13))
                                    ;MITRE LOCATION
     *M11=(XY(15,8))
     *M12=(XY(25,8))
     *M13=(XY(27,8))
     *M14=(XY(29,8))
                                    ;PRESS LOCATIONS
     *H11=(XY(39,1))
     *H12=(XY(42,1))
     *H13=(XY(44,1))                ;H3
     *H14=(XY(46,1))

     *H21=(XY(39,4))
     *H22=(XY(42,4))
     *H23=(XY(44,4))                ;H1
     *H24=(XY(46,4))

     *H31=(XY(39,6))
     *H32=(XY(42,6))
     *H33=(XY(44,6))                ;H2
     *H34=(XY(46,6))

     *H41=(XY(39,8))
     *H42=(XY(42,8))
     *H43=(XY(44,8))                ;H9
     *H44=(XY(46,8))

     *H51=(XY(39,10))
     *H52=(XY(42,10))
     *H53=(XY(44,10))               ;H7
     *H54=(XY(46,10))
```


```
     *H61=(XY(39,14))
```

```
*H62=(XY(42,14))
*H63=(XY(44,14))                        ;H6
*H64=(XY(46,14))

*H71=(XY(39,16))
*H72=(XY(42,16))
*H73=(XY(44,16))                        ;H5
*H74=(XY(46,16))

*H81=(XY(39,18))
*H82=(XY(42,18))
*H83=(XY(44,18))                        ;H4
*H84=(XY(46,18))
                                        ;OP LOCATIONS
*P11=(XY(39,12))
*P12=(XY(42,12))
*P13=(XY(44,12))                        ;1_5MM
*P14=(XY(46,12))

*P21=(XY(39,20))
*P22=(XY(42,20))
*P23=(XY(44,20))                        ;2MM
*P24=(XY(46,20))
                                        ;STORE LOCATIONS
*STORE=(XY(65,11))
*END=(XY(79,11))
*BATCH=(XY(15,18))                      ;PRINTING BATCH NO. ON SCREEN
*TIME=(XY(15,20))                       ;PRINTING BATCH TIME ON SCREEN
                                        ;UTILISATION
U=(1,BBUF,*B12)                         ;BEHRENS
U=(2,LASER,*B13)
U=(3,BBUF,*B22)                         ;A NUMBER OF UTILISATIONS
U=(4,B1_5,*B23)                         ;NOT USED AS PRINT OUT TOO
U=(5,BBUF,*B32)                         ;LARGE FOR PAPER!
U=(6,B2_0,*B33)
U=(7,MBUF,*M12)                         ;MITRE
U=(8,MITRE,*M13)
; U=(9,H3BUF,*H12)                      ;HAMMERLE
; U=(10,H3,*H13)
U=(11,H1BUF,*H22)
U=(12,H1NC,*H23)
; U=(13,H2BUF,*H32)
; U=(14,H2,*H33)
; U=(15,H9BUF,*H42)
; U=(16,H9,*H43)
; U=(17,H7BUF,*H52)
; U=(18,H7,*H53)
; U=(19,H6BUF,*H62)
; U=(20,H6,*H63)
U=(21,H5BUF,*H72)
U=(22,H5NC,*H73)
; U=(23,H4BUF,*H82)
; U=(24,H4,*H83)
; U=(25,OPBUF,*P12)                     ;OP PRESS
; U=(26,OP1_5,*P13)
; U=(27,OPBUF,*P22)
; U=(28,OP20,*P23)

                                        ;SET JOB QUANTITIES
J=(1,I,1,0,0,0,1)
J=(2,A,2,0,0,1,2000)
                                        ;BEGIN INITIALISING
BR(1,XY(1,1),0)                         ;BEGIN ROUTE
SV(@THISFAM,1)                          ;SET START TEST POINTERS
SV(@THISSTR,1)
```

```
    SV(@RTE,60)
    SV(@RTF,6)
    OF(JON\TEST.DAT)                    ;READ ARRAY (OPEN FILE)
:R1/10                                  ;LOAD DATA FROM FILE
    GD(@R1PROD)                         ;GET FIRST ENTRY IN LINE
    IF(@R1PROD,EQ,#EOF_FLAG,:R1/50)     ;QUIT IF END OF FLAG
    SV(@@SCHED(#SFLG,@R1PROD),1)        ;SET FLAG TO=1 LINE USED
    GD(@@SCHED(#PART,@R1PROD))          ;LOAD REST OF LINE..
    GD(@@SCHED(#SFAM,@R1PROD))
    GD(@@SCHED(#SQTY,@R1PROD))
    GD(@@SCHED(#SRTE,@R1PROD))
    GD(@@SCHED(#SRTF,@R1PROD))
    GD(@@SCHED(#SPTY,@R1PROD))
    GD(%%SCHED(#SCUT,@R1PROD))
    GD(%%SCHED(#SPUN,@R1PROD))
    GD(%%SCHED(#SBND,@R1PROD))
    GD(@@SCHED(#SASC,@R1PROD))
    JP(:R1/10)                          ;NEXT LINE (EOF_FLAG STOPS LOOP)
:R1/50
    OF(=.RPT)                           ;OPEN FILE REPORT
    PM(F,%%PROD_TIMES)                  ;PRINT MESSAGE PRODUCTION TIMES
    ER                                  ;END ROUTE

                                        ;BEGIN JOB
    BR(2,XY(0,2),0)                     ;BEGIN ROUTE
                                        ;RELEASE LOGIC
    SV(@START,@RTF)                     ;SET VALUE
:LAB5                                   ;JUMP LABEL
    IV(@RTF)                            ;INCREMENT COUNT
    IF(@RTF,LE,6,:LAB10)                ;IF
    SV(@RTF,1)                          ;SET VALUE
:LAB10
    SV(@ORD,0)                          ;SET VALUE
:LAB20
    IV(@ORD)                                    ;INCREMENT COUNT
    IF(@@SCHED(#SFLG,@ORD),EQ,0,:LAB50)         ;IF..THEN
    IF(@@SCHED(#SFLG,@ORD),NE,1,:LAB20)         ;IF..THEN
    IF(@@SCHED(#SRTF,@ORD),NE,@RTF,:LAB20)      ;JUMP
    JP(:LAB100)
:LAB50
    IF(@RTF,NE,@START,:LAB5)            ;CLOCK_SYSTEM CLOCK
    IF(CLOCK,GT,0,WAIT)                 ;WAIT TIL NEXT EVENT
:LAB100                                 ;OR CLOCK UPDATE
    SV(@@SCHED(#SFLG,@ORD),2)           ;SET FLAG
    SV(OBJ@1,@ORD)                      ;SET ORDER
    SV(OBJ@2,@@SCHED(#PART,@ORD))       ;GET PART
    SV(OBJ@3,@@SCHED(#SFAM,@ORD))       ;GET FAMILY
    SV(OBJ@4,@@SCHED(#SQTY,@ORD))       ;GET QUANTITY
    SV(OBJ@5,@@SCHED(#SRTE,@ORD))       ;GET FIRST ROUTE
    SV(OBJ@6,@@SCHED(#SRTF,@ORD))       ;GET SECOND ROUTE
    SV(OBJ@ID,@@SCHED(#SASC,@ORD))      ;GET ASCII VALUE OF ID
                                        ;MOVE LOGIC
    MA(*START2,0)                       ;MOVE ABSOLUTE
    SV(OBJ%1,CLOCK)                     ;SET TO SYSTEM CLOCK
    SV(%%PROD_TIMES(#PSAL,OBJ@1),CLOCK);CLOCK SET AS NO CUTTING
    IV(@CU)                             ;INCREMENT @CU
    PV(XY(0,10),@CU)                    ;PRINT VALUE


;--------NEW ROUTE--------------------

    IF(OBJ@6,EQ,1,:ROUTE11)             ;DESIGNATE JOB ROUTES
    IF(OBJ@6,EQ,2,:ROUTE20)
    IF(OBJ@6,EQ,3,:ROUTE30)
    IF(OBJ@6,EQ,4,:ROUTE40)
```

```
    IF(OBJ@6,EQ,5,:ROUTE51)
    IF(OBJ@6,EQ,6,:ROUTE60)

 :ROUTE10                               ;BLASER NO H
    MR(11,%MOVE)                        ;MOVE RIGHT
    MU(1,%MOVE)                         ;MOVE UP
    MR(19,%MOVE)
    IV(@L2)                             ;INCREMENT COUNTER
    PV(XY(68,0),@L2)
    MD(11,%MOVE)                        ;MOVE DOWN
    SV(%WORK6,CLOCK)
    AO(%WORK6,-,OBJ%1)                  ;ARITHMETIC OPERATION
    SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK6)
    PM(F,NO-MACHINING)
    PV(F,%WORK6)                        ;PRINT VALUE TO FILE
    JP(:END)                            ;JUMP TO END

 :ROUTE11                               ;BLASER H3
    MR(5,%MOVE)
    MU(10,%MOVE)
    MR(10,%MOVE)
    TP(2,*B12,*B13)
    MA(*B12,0)
    TP(1,*B13)
    IF(OBJ@3,EQ,@LASTPUN1,:SET1)        ;SET VALUE SETTING TIME
    WT(%SET_DIFB)
    JP(:CONT11)

:SET1
    WT(%SET_SAMB)

:CONT11
    SV(@LASTPUN1,OBJ@3)
    TP(*B13)
    MA(*B13,0)
    SV(%PUNCH1,%%SCHED(#SPUN,OBJ@1))
    AO(%PUNCH1,*,@@SCHED(#SQTY,OBJ@1))  ;ARITHMETIC OPERATION
    WT(%PUNCH1)
    JP(:CONT12)

:CONT12
    TP(1,*B14)
    MA(*B14,0)
    SV(%WORK2,CLOCK)
    AO(%WORK2,-,OBJ%1)
    SV(%%PROD_TIMES(#P1ST,OBJ@1),%WORK2)
    PM(F,BEHRENS-LASER)                 ;PRINT MESSAGE TO FILE
    PV(F,%WORK2)                        ;PRINT VALUE TO FILE
    IV(@BL)                             ;INCREMENT COUNTER
    PV(XY(10,2),@BL)                    ;PRINT COUNTER VALUE ON SCREEN
    SC(4,8)                             ;CHANGE JOB IDENTITY COLOUR
    IF(OBJ@5,EQ,10,:ROUTE10)            ;DESIGNATE ROUTES AFTER
    IF(OBJ@5,EQ,12,:ROUTE12)            ;THE BEHRENS OPERATION
    IF(OBJ@5,EQ,13,:ROUTE13)
    IF(OBJ@5,EQ,14,:ROUTE14)
    IF(OBJ@5,EQ,15,:ROUTE15)
    IF(OBJ@5,EQ,16,:ROUTE16)
    IF(OBJ@5,EQ,17,:ROUTE17)
    IF(OBJ@5,EQ,18,:ROUTE18)
    IF(OBJ@5,EQ,1,:ROUTEP1)
    IF(OBJ@5,EQ,2,:ROUTEP2)
    IF(OBJ@5,EQ,4,:ROUTE13)
    IF(OBJ@5,EQ,5,:ROUTE5)
    IF(OBJ@5,EQ,6,:ROUTE14)
```



```
    TP(*H11,:CONT13)
```

```
:CONT13
   MR(17,%MOVE)
   TP(2,*H12,*H13)
   MA(*H12,0)
   TP(1,*H13)
   IF(OBJ@3,EQ,@LASTBND1,:SET11)
   WT(%SET_DIFH)
   JP(:CONT14)

:SET11
   WT(%SET_SAMH)

:CONT14
   SV(@LASTBND1,OBJ@3)
   MA(*H13,0)
   SV(%BEND1,%%SCHED(#SBND,OBJ@2))
   AO(%BEND1,+,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND1)
   MA(*H14,%MOVE)
   SV(%WORK7,CLOCK)
   AO(%WORK7,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK7)
   PM(F,HAMMERLE-3)
   PV(F,%WORK7)
   IV(@H1)
   PV(XY(68,1),@H1)
   IF(OBJ@5,EQ,3,:ROUTE3)
   MR(6,%MOVE)
   MD(10,%MOVE)
   JP(:END)

:ROUTE12                              ;BLASER H1
   JC(*H21,:CONT23)
:CONT23
   MR(11,%MOVE)
   MD(3,%MOVE)
   MR(6,%MOVE)
   TP(2,*H22,*H23)
   MA(*H22,0)
   TP(1,*H23)
   IF(OBJ@3,EQ,@LASTBND2,:SET21)
   WT(%SET_DIFH)
   JP(:CONT24)

:SET21
   WT(%SET_SAMH)

:CONT24
   SV(@LASTBND2,OBJ@3)
   MA(*H23,0)
   SV(%BEND2,%%SCHED(#SBND,OBJ@2))
   AO(%BEND2,+,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND2)
   MA(*H24,%MOVE)
   SV(%WORK8,CLOCK)
   AO(%WORK8,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK8)
   PM(F,HAMMERLE-1)
   PV(F,%WORK8)
   IV(@H2)
   PV(XY(68,4),@H2)
   IF(OBJ@5,EQ,5,:ROUTE5)
   MR(6,%MOVE)
   MD(7,%MOVE)
   JP(:END)
```

```
:ROUTE13                                ;BLASER H2
   JC(*H31,:CONT33)
:CONT33
   MR(11,%MOVE)
   MD(5,%MOVE)
   MR(6,%MOVE)
   TP(2,*H32,*H33)
   MA(*H32,0)
   TP(1,*H33)
   IF(OBJ@3,EQ,@LASTBND3,:SET31)
   WT(%SET_DIFH)
   JP(:CONT34)

:SET31
   WT(%SET_SAMH)

:CONT34
   SV(@LASTBND3,OBJ@3)
   MA(*H33,0)
   SV(%BEND3,%%SCHED(#SBND,OBJ@2))
   AO(%BEND3,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND3)
   MA(*H34,%MOVE)
   SV(%WORK9,CLOCK)
   AO(%WORK9,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK9)
   PM(F,HAMMERLE-2)
   PV(F,%WORK9)
   IV(@H3)
   PV(XY(68,6),@H3)
   IF(OBJ@5,EQ,4,:ROUTE4)
   MR(6,%MOVE)
   MD(5,%MOVE)
   JP(:END)

:ROUTE14                                ;BLASER H9
   JC(*H41,:CONT43)
:CONT43
   MR(11,%MOVE)
   MD(7,%MOVE)
   MR(6,%MOVE)
   TP(2,*H42,*H43)
   MA(*H42,%MOVE)
   TP(1,*H43)
   IF(OBJ@3,EQ,@LASTBND4,:SET41)
   WT(%SET_DIFH)
   JP(:CONT44)

:SET41
   WT(%SET_SAMH)

:CONT44
   SV(@LASTBND4,OBJ@3)
   MA(*H43,0)
   SV(%BEND4,%%SCHED(#SBND,OBJ@2))
   AO(%BEND4,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND4)
   MA(*H44,%MOVE)
   SV(%WORK10,CLOCK)
   AO(%WORK10,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK10)
   PM(F,HAMMERLE-9)
   PV(F,%WORK10)
   IV(@H4)
```

```
   PV(XY(68,8),@H4)
   IF(OBJ@5,EQ,6,:ROUTE6)
   MR(6,%MOVE)
   MD(3,%MOVE)
   JP(:END)

:ROUTE15                              ;BLASER H7
   JC(*H51,:CONT53)
:CONT53
   MR(11,%MOVE)
   MD(9,%MOVE)
   MR(6,%MOVE)
   TP(2,*H52,*H53)
   MA(*H52,0)
   TP(1,*H53)
   IF(OBJ@3,NE,@LASTBND5,:SET51)
   WT(%SET_SAMH)
   JP(:CONT54)

:SET51
   WT(%SET_DIFH)

:CONT54
   SV(@LASTBND5,OBJ@3)
   MA(*H53,0)
   SV(%BEND5,%%SCHED(#SBND,OBJ@2))
   AO(%BEND5,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND5)
   TP(1,*H54)
   MA(*H54,%MOVE)
   SV(%WORK11,CLOCK)
   AO(%WORK11,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK11)
   PM(F,HAMMERLE-7)
   PV(F,%WORK11)
   IV(@H5)
   PV(XY(68,10),@H5)
   MR(6,%MOVE)
   MD(1,%MOVE)
   JP(:END)

:ROUTE16                              ;BLASER H6
   JC(*H61,:CONT63)
:CONT63
   MR(11,%MOVE)
   MD(13,%MOVE)
   MR(6,%MOVE)
   TP(2,*H62,*H63)
   MA(*H62,0)
   TP(1,*H63)
   IF(OBJ@3,EQ,@LASTBND6,:SET61)
   WT(%SET_DIFH)
   JP(:CONT64)

:SET61
   WT(%SET_SAMH)

:CONT64
   SV(@LASTBND6,OBJ@3)
   MA(*H63,0)
   SV(%BEND6,%%SCHED(#SBND,OBJ@2))
   AO(%BEND6,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND6)
   MA(*H64,%MOVE)
   SV(%WORK14,CLOCK)
```

```
   AO(%WORK14,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK14)
   PM(F,HAMMERLE-6)
   PV(F,%WORK14)
   IV(@H6)
   PV(XY(68,14),@H6)
   MR(6,%MOVE)
   MU(3,%MOVE)
   JP(:END)

:ROUTE17                              ;BLASER H5
   JC(*H71,:CONT73)
:CONT73
   MR(11,%MOVE)
   MD(15,%MOVE)
   MR(6,%MOVE)
   TP(2,*H72,*H73)
   MA(*H72,0)
   TP(1,*H73)
   IF(OBJ@3,EQ,@LASTBND7,:SET71)
   WT(%SET_DIFH)
   JP(:CONT74)

:SET71
   WT(%SET_SAMH)

:CONT74
   SV(@LASTBND7,OBJ@3)
   MA(*H73,0)
   SV(%BEND7,%%SCHED(#SBND,OBJ@2))
   AO(%BEND7,+,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND7)
   MA(*H74,%MOVE)
   SV(%WORK15,CLOCK)
   AO(%WORK15,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK15)
   PM(F,HAMMERLE-5)
   PV(F,%WORK15)
   IV(@H7)
   PV(XY(68,16),@H7)
   MR(6,%MOVE)
   MU(5,%MOVE)
   JP(:END)

:ROUTE18                              ;BLASER H4
   JC(*H81,:CONT83)
:CONT83
   MR(11,%MOVE)
   MD(17,%MOVE)
   MR(6,%MOVE)
   TP(2,*H82,*H83)
   MA(*H82,0)
   TP(1,*H83)
   IF(OBJ@3,EQ,@LASTBND8,:SET81)
   WT(%SET_DIFH)
   JP(:CONT84)

:SET81
   WT(%SET_SAMH)

:CONT84
   SV(@LASTBND8,OBJ@3)
   MA(*H83,0)
   SV(%BEND8,%%SCHED(#SBND,OBJ@2))
```

```
   AO(%BEND8,+,@@SCHED(#SQTY,OBJ@1))
```

```
   WT(%BEND8)
   MA(*H84,%MOVE)
   SV(%WORK16,CLOCK)
   AO(%WORK16,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK16)
   PM(F,HAMMERLE-4)
   PV(F,%WORK16)
   IV(@H8)
   PV(XY(68,18),@H8)
   MR(6,%MOVE)
   MU(7,%MOVE)
   JP(:END)

:ROUTEP1                                ;BLASER OP1_5MM
   JC(*P11,:CONTP3)
:CONTP3
   MR(11,%MOVE)
   MD(11,%MOVE)
   MR(6,%MOVE)
   TP(2,*P12,*P13)
   MA(*P12,0)
   TP(1,*P13)
   IF(OBJ@3,EQ,@LASTPRS1,:SETP1)
   WT(%SET_DIFP)
   JP(:CONTP4)

:SETP1
   WT(%SET_SAMP)

:CONTP4
   SV(@LASTPRS1,OBJ@3)
   MA(*P13,0)
   SV(%PRESS1,%%SCHED(#SBND,OBJ@2))
   AO(%PRESS1,*,@@SCHED(#SQTY,OBJ@1))
   WT(%PRESS1)
   MA(*P14,%MOVE)
   SV(%WORK12,CLOCK)
   AO(%WORK12,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK12)
   PM(F,OP15)
   PV(F,%WORK12)
   IV(@O1)
   PV(XY(68,12),@O1)
   MR(6,%MOVE)
   MU(1,%MOVE)
   JP(:END)

:ROUTEP2                                ;BLASER OP2MM
   JC(*P21,:CONTP23)
:CONTP23
   MR(11,%MOVE)
   MD(19,%MOVE)
   MR(6,%MOVE)
   TP(2,*P22,*P23)
   MA(*P22,0)
   TP(1,*P23)
   IF(OBJ@3,EQ,@LASTPRS2,:SETP21)
   WT(%SET_DIFP)
   JP(:CONTP24)

:SETP21
   WT(%SET_SAMP)
```



```
:CONTP24
   SV(@LASTPRS2,OBJ@3)
   MA(*P23,0)
```

```
   SV(%PRESS2,%%SCHED(#SBND,OBJ@2))
   AO(%PRESS2,*,@@SCHED(#SQTY,OBJ@1))
   WT(%PRESS2)
   MA(*P24,%MOVE)
   SV(%WORK17,CLOCK)
   AO(%WORK17,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK17)
   PM(F,OP20)
   PV(F,%WORK17)
   IV(@02)
   PV(XY(68,20),@02)
   MR(6,%MOVE)
   MU(9,%MOVE)
   JP(:END)


:ROUTE3                              ;BLASER H3 H1 H3
   JC(*H21,:CONT1001)

:CONT1001
   MR(3,%MOVE)
   MD(1,%MOVE)
   ML(9,%MOVE)
   TP(2,*H22,*H23)
   MA(*H22,0)
   TP(1,*H23)
   IF(OBJ@3,EQ,@LASTBND2,:SET1002)
   WT(%SET_DIFH)
   JP(:CONT1003)

:SET1002
   WT(%SET_SAMH)

:CONT1003
   SV(@LASTBND2,OBJ@3)
   MA(*H23,0)
   SV(%BEND2,%%SCHED(#SBND,OBJ@2))
   AO(%BEND2,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND2)
   MA(*H24,%MOVE)
   SV(%WORK8,CLOCK)
   AO(%WORK8,-,OBJ%1)
   SV(%%PROD_TIMES(#P3RD,OBJ@1),%WORK8)
   PM(F,HAMMERLE-1)
   PV(F,%WORK8)
   IV(@H2)
   PV(XY(68,4),@H2)
   JC(*H11,:CONT1004)

:CONT1004
   MR(3,%MOVE)
   MU(1,%MOVE)
   ML(10,%MOVE)
   TP(2,*H12,*H13)
   MA(*H12,0)
   TP(1,*H13)
   IF(OBJ@3,EQ,@LASTBND1,:SET1005)
   WT(%SET_DIFH)
   JP(:CONT1006)

:SET1005
   WT(%SET_SAMH)

:CONT1006
   SV(@LASTBND1,OBJ@3)
   MA(*H13,0)
```

```
   SV(%BEND1,%%SCHED(#SBND,OBJ@2))
   AO(%BEND1,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND1)
   MA(*H14,%MOVE)
   SV(%WORK7,CLOCK)
   AO(%WORK7,-,OBJ%1)
   SV(%%PROD_TIMES(#P4TH,OBJ@1),%WORK7)
   PM(F,HAMMERLE-3)
   PV(F,%WORK7)
   IV(@H1)
   PV(XY(68,1),@H1)
   MR(6,%MOVE)
   MD(10,%MOVE)
   JP(:END)

:ROUTE4                              ;BLASER H2 H1
   JC(*H21,:CONT1010)
:CONT1010
   MR(3,%MOVE)
   MU(1,%MOVE)
   ML(9,%MOVE)
   TP(2,*H22,*H23)
   MA(*H22,0)
   TP(1,*H23)
   IF(OBJ@3,EQ,@LASTBND2,:SET1012)
   WT(%SET_DIFH)
   JP(:CONT1013)

:SET1012
   WT(%SET_SAMH)

:CONT1013
   SV(@LASTBND2,OBJ@3)
   MA(*H23,0)
   SV(%BEND2,%%SCHED(#SBND,OBJ@2))
   AO(%BEND2,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND2)
   MA(*H24,%MOVE)
   SV(%WORK8,CLOCK)
   AO(%WORK8,-,OBJ%1)
   SV(%%PROD_TIMES(#P3RD,OBJ@1),%WORK8)
   PM(F,HAMMERLE-1)
   PV(F,%WORK8)
   IV(@H2)
   PV(XY(68,4),@H2)
   MR(6,%MOVE)
   MD(7,%MOVE)
   JP(:END)

:ROUTE5                              ;BLASER H1 H3
   JC(*H21,:CONT1030)
:CONT1030
   MR(11,%MOVE)
   MD(3,%MOVE)
   MR(6,%MOVE)
   TP(2,*H22,*H23)
   MA(*H22,0)
   TP(1,*H23)
   IF(OBJ@3,EQ,@LASTBND2,:SET1031)
   WT(%SET_DIFH)
   JP(:CONT1032)

:SET1031
   WT(%SET_SAMH)
```

```
:CONT1032
   SV(@LASTBND2,OBJ@3)
   MA(*H23,0)
   SV(%BEND2,%%SCHED(#SBND,OBJ@2))
   AO(%BEND2,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND2)
   MA(*H24,%MOVE)
   SV(%WORK8,CLOCK)
   AO(%WORK8,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK8)
   PM(F,HAMMERLE-1)
   PV(F,%WORK8)
   IV(@H2)
   PV(XY(68,4),@H2)
   JC(*H11,:CONT1033)

:CONT1033
   MR(3,%MOVE)
   MU(1,%MOVE)
   ML(10,%MOVE)
   TP(2,*H12,*H13)
   MA(*H12,0)
   TP(1,*H13)
   IF(OBJ@3,EQ,@LASTBND1,:SET1034)
   WT(%SET_DIFH)
   JP(:CONT1035)

:SET1034
   WT(%SET_SAMH)

:CONT1035
   SV(@LASTBND1,OBJ@3)
   MA(*H13,0)
   SV(%BEND1,%%SCHED(#SBND,OBJ@2))
   AO(%BEND1,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND1)
   MA(*H14,%MOVE)
   SV(%WORK7,CLOCK)
   AO(%WORK7,-,OBJ%1)
   SV(%%PROD_TIMES(#P3RD,OBJ@1),%WORK7)
   PM(F,HAMMERLE-3)
   PV(F,%WORK7)
   IV(@H1)
   PV(XY(68,1),@H1)
   MR(6,%MOVE)
   MD(10,%MOVE)
   JP(:END)


:ROUTE6                              ;BLASER H9 H1
   JC(*H21,:CONT1021)
:CONT1021
   MR(3,%MOVE)
   MU(3,%MOVE)
   ML(9,%MOVE)
   TP(2,*H22,*H23)
   MA(*H22,0)
   TP(1,*H23)
   IF(OBJ@3,EQ,@LASTBND2,:SET1022)
   WT(%SET_DIFH)
   JP(:CONT1023)

:SET1022
   WT(%SET_SAMH)
```



```
:CONT1023
```

```
   SV(@LASTBND2,OBJ@3)
   MA(*H23,0)
   SV(%BEND2,%%SCHED(#SBND,OBJ@2))
   AO(%BEND2,+,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND2)
   MA(*H24,%MOVE)
   SV(%WORK8,CLOCK)
   AO(%WORK8,-,OBJ%1)
   SV(%%PROD_TIMES(#P3RD,OBJ@1),%WORK8)
   PM(F,HAMMERLE-1)
   PV(F,%WORK8)
   IV(@H2)
   PV(XY(68,4),@H2)
   MR(6,%MOVE)
   MD(7,%MOVE)
   JP(:END)

;--------NEW ROUTE--------------------

:ROUTE20                              ;B1_5MM H1
   MR(5,%MOVE)
   MU(7,%MOVE)
   MR(10,%MOVE)
   TP(2,*B22,*B23)
   MA(*B22,%MOVE)
   TP(1,*B23)
   IF(OBJ@3,EQ,@LASTPUN2,:SET101)
   WT(%SET_DIFB)
   JP(:CONT111)

:SET101
   WT(%SET_SAMB)

:CONT111
   SV(@LASTPUN2,OBJ@3)
   TP(1,*B23)
   MA(*B23,0)
   SV(%PUNCH2,%%SCHED(#SPUN,OBJ@1))
   AO(%PUNCH2,+,@@SCHED(#SQTY,OBJ@1))
   WT(%PUNCH2)
   JP(:CONT112)

:CONT112
   TP(1,*B24)
   MA(*B24,0)
   SV(%WORK2,CLOCK)
   AO(%WORK2,-,OBJ%1)
   SV(%%PROD_TIMES(#P1ST,OBJ@1),%WORK2)
   PM(F,BEHRENS-1.5MM)
   PV(F,%WORK2)
   IV(@B1)
   PV(XY(10,5),@B1)
   SC(5,8)
   IF(OBJ@5,EQ,21,:ROUTE21)
   IF(OBJ@5,EQ,22,:ROUTE22)
   IF(OBJ@5,EQ,23,:ROUTE23)
   IF(OBJ@5,EQ,24,:ROUTE24)
   IF(OBJ@5,EQ,25,:ROUTE25)
   IF(OBJ@5,EQ,26,:ROUTE26)

   JC(*H21,:CONT113)
:CONT113
   MR(17,%MOVE)
   TP(2,*H22,*H23)
   MA(*H22,0)
```

```
   TP(1,*H23)
   IF(OBJ@3,EQ,@LASTBND2,:SET111)
   WT(%SET_DIFH)
   JP(:CONT114)

:SET111
   WT(%SET_SAMH)

:CONT114
   SV(@LASTBND2,OBJ@3)
   MA(*H23,0)
   SV(%BEND2,%%SCHED(#SBND,OBJ@2))
   AO(%BEND2,+,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND2)
   MA(*H24,%MOVE)
   SV(%WORK8,CLOCK)
   AO(%WORK8,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK8)
   PM(F,HAMMERLE-1)
   PV(F,%WORK8)
   IV(@H2)
   PV(XY(68,4),@H2)
   MR(6,%MOVE)
   MD(7,%MOVE)
   JP(:END)

:ROUTE21                              ;B1_5MM H2
   JC(*H31,:CONT123)
:CONT123
   MR(11,%MOVE)
   MD(2,%MOVE)
   MR(6,%MOVE)
   TP(2,*H32,*H33)
   MA(*H32,0)
   TP(1,*H33)
   IF(OBJ@3,EQ,@LASTBND3,:SET121)
   WT(%SET_DIFH)
   JP(:CONT124)

:SET121
   WT(%SET_SAMH)

:CONT124
   SV(@LASTBND3,OBJ@3)
   MA(*H33,0)
   SV(%BEND3,%%SCHED(#SBND,OBJ@2))
   AO(%BEND3,+,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND3)
   MA(*H34,%MOVE)
   SV(%WORK9,CLOCK)
   AO(%WORK9,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK9)
   PM(F,HAMMERLE-2)
   PV(F,%WORK9)
   IV(@H3)
   PV(XY(68,6),@H3)
   MR(6,%MOVE)
   MD(5,%MOVE)
   JP(:END)

:ROUTE22                              ;B1_5MM H9
   JC(*H41,:CONT133)
:CONT133
   MR(11,%MOVE)
   MD(4,%MOVE)
```

```
   MR(6,%MOVE)
   TP(2,*H42,*H43)
   MA(*H42,0)
   TP(1,*H43)
   IF(OBJ@3,EQ,@LASTBND4,:SET131)
   WT(%SET_DIFH)
   JP(:CONT134)

:SET131
   WT(%SET_SAMH)

:CONT134
   SV(@LASTBND4,OBJ@3)
   MA(*H43,0)
   SV(%BEND4,%%SCHED(#SBND,OBJ@2))
   AO(%BEND4,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND4)
   MA(*H44,%MOVE)
   SV(%WORK10,CLOCK)
   AO(%WORK10,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK10)
   PM(F,HAMMERLE-9)
   PV(F,%WORK10)
   IV(@H4)
   PV(XY(68,8),@H4)
   MR(6,%MOVE)
   MD(3,%MOVE)
   JP(:END)

:ROUTE23                          ;B1_5MM H7
   JC(*H51,:CONT143)
:CONT143
   MR(11,%MOVE)
   MD(6,%MOVE)
   MR(6,%MOVE)
   TP(2,*H52,*H53)
   MA(*H52,0)
   TP(1,*H53)
   IF(OBJ@3,EQ,@LASTBND5,:SET141)
   WT(%SET_DIFH)
   JP(:CONT144)

:SET141
   WT(%SET_SAMH)

:CONT144
   SV(@LASTBND5,OBJ@3)
   MA(*H53,0)
   SV(%BEND5,%%SCHED(#SBND,OBJ@2))
   AO(%BEND5,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND5)
   TP(1,*H54)
   MA(*H54,%MOVE)
   SV(%WORK11,CLOCK)
   AO(%WORK11,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK11)
   PM(F,HAMMERLE-7)
   PV(F,%WORK11)
   IV(@H5)
   PV(XY(68,10),@H5)
   MR(6,%MOVE)
   MD(1,%MOVE)
   JP(:END)

:ROUTE24                          ;B1_5MM OP1
```

```
   JC(*P11,:CONT153)
:CONT153
   MR(11,%MOVE)
   MD(8,%MOVE)
   MR(6,%MOVE)
   TP(2,*P12,*P13)
   MA(*P12,0)
   TP(1,*P13)
   IF(OBJ@3,EQ,@LASTPRS1,:SET151)
   WT(%SET_DIFP)
   JP(:CONT154)

:SET151
   WT(%SET_SAMP)

:CONT154
   SV(@LASTPRS1,OBJ@3)
   MA(*P13,0)
   SV(%PRESS1,%%SCHED(#SBND,OBJ@2))
   AO(%PRESS1,+,@@SCHED(#SQTY,OBJ@1))
   WT(%PRESS1)
   MA(*P14,%MOVE)
   SV(%WORK12,CLOCK)
   AO(%WORK12,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK12)
   PM(F,OP15)
   PV(F,%WORK12)
   IV(@01)
   PV(XY(68,12),@01)
   MR(6,%MOVE)
   MU(1,%MOVE)
   JP(:END)

:ROUTE25                              ;B1_5MM ONLY
   MR(11,%MOVE)
   MD(9,%MOVE)
   MR(19,%MOVE)
   IV(@L4)
   PV(XY(68,13),@L4)
   MU(2,%MOVE)
   SV(%WORK13,CLOCK)
   AO(%WORK13,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK13)
   PM(F,NO-MACHINING)
   PV(F,%WORK13)
   JP(:END)


:ROUTE26                              ;B1_5MM M H1
   JC(*M11,:CONT171)
:CONT171
   MR(2,%MOVE)
   MD(3,%MOVE)
   TP(2,*M12,*M13)
   MA(*M12,0)
   TP(1,*M13)
   IF(OBJ@3,EQ,@LASTMITR,:SET172)
   WT(%SET_DIFM)
   JP(:CONT173)

:SET172
   WT(%SET_SAMM)

:CONT173
   SV(@LASTMITR,OBJ@3)
   TP(1,*M13)
```

```
   MA(*M13,0)
   SV(%MITRE,%%SCHED(#SPUN,OBJ@1))
   AO(%MITRE,*,@@SCHED(#SQTY,OBJ@1))
   WT(%MITRE)
   MA(*M14,%MOVE)
   SV(%WORK3,CLOCK)
   AO(%WORK3,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK3)
   PM(F,MITRE)
   PV(F,%WORK3)
   IV(@M2)
   PV(XY(25,6),@M2)
   JC(*H21,:CONT175)

:CONT175
   MR(2,%MOVE)
   MU(4,%MOVE)
   MR(8,%MOVE)
   TP(2,*H22,*H23)
   MA(*H22,0)
   TP(1,*H23)
   IF(OBJ@3,EQ,@LASTBND2,:SET176)
   WT(%SET_DIFH)
   JP(:CONT177)

:SET176
   WT(%SET_SAMH)

:CONT177
   SV(@LASTBND2,OBJ@3)
   MA(*H23,0)
   SV(%BEND2,%%SCHED(#SBND,OBJ@2))
   AO(%BEND2,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND2)
   MA(*H24,%MOVE)
   SV(%WORK8,CLOCK)
   AO(%WORK8,-,OBJ%1)
   SV(%%PROD_TIMES(#P3RD,OBJ@1),%WORK8)
   PM(F,HAMMERLE-1)
   PV(F,%WORK8)
   IV(@H2)
   PV(XY(68,4),@H2)
   MR(6,%MOVE)
   MD(7,%MOVE)
   JP(:END)

;--------NEW ROUTE---------------------

:ROUTE30                          ;MITRE H1
   MR(5,%MOVE)
   MU(3,%MOVE)
   MR(10,%MOVE)
   IV(@M1)
   PV(XY(10,9),@M1)
   TP(2,*M12,*M13)
   MA(*M12,0)
   TP(1,*M13)
   IF(OBJ@3,EQ,@LASTMITR,:SET501)
   WT(%SET_DIFM)
   JP(:CONT511)

:SET501
   WT(%SET_SAMM)

:CONT511
   SV(@LASTMITR,OBJ@3)
```

```
   TP(1,*M13)
   MA(*M13,0)
   SV(%MITRE,%%SCHED(#SPUN,OBJ@1))
   AO(%MITRE,*,@@SCHED(#SQTY,OBJ@1))
   WT(%MITRE)
   MA(*M14,%MOVE)
   SV(%WORK3,CLOCK)
   AO(%WORK3,-,OBJ%1)
   SV(%%PROD_TIMES(#P1ST,OBJ@1),%WORK3)
   PM(F,MITRE)
   PV(F,%WORK3)
   IV(@M2)
   PV(XY(25,6),@M2)
   IF(OBJ@5,EQ,31,:ROUTE31)

   JC(*H21,:CONT513)
:CONT513
   MR(2,%MOVE)
   MU(4,%MOVE)
   MR(8,%MOVE)
   TP(2,*H22,*H23)
   MA(*H22,0)
   TP(1,*H23)
   IF(OBJ@3,EQ,@LASTBND2,:SET511)
   WT(%SET_DIFH)
   JP(:CONT514)

:SET511
   WT(%SET_SAMH)

:CONT514
   SV(@LASTBND2,OBJ@3)
   MA(*H23,0)
   SV(%BEND2,%%SCHED(#SBND,OBJ@2))
   AO(%BEND2,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND2)
   MA(*H24,%MOVE)
   SV(%WORK8,CLOCK)
   AO(%WORK8,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK8)
   PM(F,HAMMERLE-1)
   PV(F,%WORK8)
   IV(@H2)
   PV(XY(68,4),@H2)
   MR(6,%MOVE)
   MD(7,%MOVE)
   JP(:END)

:ROUTE31                          ;MITRE H2
   JC(*H31,:CONT523)

:CONT523
   MR(2,%MOVE)
   MU(4,%MOVE)
   MR(2,%MOVE)
   MD(2,%MOVE)
   MR(6,%MOVE)
   TP(2,*H32,*H33)
   MA(*H32,0)
   TP(1,*H33)
   IF(OBJ@3,EQ,@LASTBND3,:SET521)
   WT(%SET_DIFH)
   JP(:CONT524)

:SET521
```

```
   WT(XSET_SAMH)

 :CONT524
   SV(@LASTBND3,OBJ@3)
   MA(*H33,0)
   SV(XBEND3,XXSCHED(#SBND,OBJ@2))
   AO(XBEND3,*,@@SCHED(#SQTY,OBJ@1))
   WT(XBEND3)
   MA(*H34,XMOVE)
   SV(XWORK9,CLOCK)
   AO(XWORK9,-,OBJX1)
   SV(XXPROD_TIMES(#P2ND,OBJ@1),XWORK9)
   PM(F,HAMMERLE-2)
   PV(F,XWORK9)
   IV(@H3)
   PV(XY(68,6),@H3)
   MR(6,XMOVE)
   MD(5,XMOVE)
   JP(:END)

;--------NEW ROUTE--------------------

 :ROUTE40                              ;NO B OP1
   MR(5,XMOVE)
   MD(1,XMOVE)
   MR(10,XMOVE)
   SV(XWORK4,CLOCK)
   AO(XWORK4,-,OBJX1)
   SV(XXPROD_TIMES(#P1ST,OBJ@1),XWORK4)
   PM(F,NO-MACHINING)
   PV(F,XWORK4)
   IV(@L1)
   PV(XY(10,11),@L1)
   IF(OBJ@5,EQ,41,:ROUTE41)

   JC(*P11,:CONT413)
 :CONT413
   MR(18,XMOVE)
   MR(6,XMOVE)
   TP(2,*P12,*P13)
   MA(*P12,0)
   TP(1,*P13)
   IF(OBJ@3,EQ,@LASTPRS1,:SET411)
   WT(XSET_DIFP)
   JP(:CONT414)

 :SET411
   WT(XSET_SAMP)

 :CONT414
   SV(@LASTPRS1,OBJ@3)
   MA(*P13,0)
   SV(XPRESS1,XXSCHED(#SBND,OBJ@2))
   AO(XPRESS1,*,@@SCHED(#SQTY,OBJ@1))
   WT(XPRESS1)
   MA(*P14,XMOVE)
   SV(XWORK12,CLOCK)
   AO(XWORK12,-,OBJX1)
   SV(XXPROD_TIMES(#P2ND,OBJ@1),XWORK12)
   PM(F,OP15)
   PV(F,XWORK12)
   IV(@O1)
   PV(XY(68,12),@O1)
   MR(6,XMOVE)
   MU(1,XMOVE)
```

```
   JP(:END)

:ROUTE41                          ;NO B OP2
   JC(*P21,:CONT423)
:CONT423
   MR(18,%MOVE)
   MD(8,%MOVE)
   MR(6,%MOVE)
   TP(2,*P22,*P23)
   MA(*P22,0)
   TP(1,*P23)
   IF(OBJ@3,EQ,@LASTPRS2,:SET421)
   WT(%SET_DIFP)
   JP(:CONT424)

:SET421
   WT(%SET_SAMP)

:CONT424
   SV(@LASTPRS2,OBJ@3)
   MA(*P23,0)
   SV(%PRESS2,%%SCHED(#SBND,OBJ@2))
   AO(%PRESS2,*,@@SCHED(#SQTY,OBJ@1))
   WT(%PRESS2)
   MA(*P24,%MOVE)
   SV(%WORK17,CLOCK)
   AO(%WORK17,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK17)
   PM(F,OP20)
   PV(F,%WORK17)
   IV(@02)
   PV(XY(68,20),@02)
   MR(6,%MOVE)
   MU(9,%MOVE)
   JP(:END)

;--------NEW ROUTE---------------------

:ROUTE50                          ;B2_OMM NO H
   MR(30,%MOVE)
   IV(@L4)
   PV(XY(68,13),@L4)
   MU(2,%MOVE)
   SV(%WORK13,CLOCK)
   AO(%WORK13,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK13)
   PM(F,NO-MACHINING)
   PV(F,%WORK13)
   JP(:END)

:ROUTE51                          ;B2_OMM H6
   MR(5,%MOVE)
   MD(2,%MOVE)
   MR(10,%MOVE)
   TP(2,*B32,*B33)
   MA(*B32,0)
   TP(1,*B33)
   IF(OBJ@3,EQ,@LASTPUN3,:SET301)
   WT(%SET_DIFB)
   JP(:CONT311)
```



```
:SET301
   WT(%SET_SAMB)

:CONT311
```

```
   SV(@LASTPUN3,OBJ@3)
   TP(1,*B33)
   MA(*B33,0)
   SV(%PUNCH3,%%SCHED(#SPUN,OBJ@1))
   AO(%PUNCH3,*,@@SCHED(#SQTY,OBJ@1))
   WT(%PUNCH3)
   JP(:CONT312)

:CONT312
   TP(1,*B34)
   MA(*B34,0)

   SV(%WORK5,CLOCK)
   AO(%WORK5,-,OBJ%1)
   SV(%%PROD_TIMES(#P1ST,OBJ@1),%WORK5)
   PM(F,BEHRENS-2MM)
   PV(F,%WORK5)
   IV(@B2)
   PV(XY(10,14),@B2)
   SA(7,9)
   IF(OBJ@5,EQ,50,:ROUTE50)
   IF(OBJ@5,EQ,52,:ROUTE52)
   IF(OBJ@5,EQ,53,:ROUTE53)
   IF(OBJ@5,EQ,54,:ROUTE54)

   JC(*H61,:CONT313)
:CONT313
   MR(11,%MOVE)
   MD(1,%MOVE)
   MR(6,%MOVE)
   TP(2,*H62,*H63)
   MA(*H62,0)
   TP(1,*H63)
   IF(OBJ@3,EQ,@LASTBND6,:SET311)
   WT(%SET_DIFH)
   JP(:CONT314)

:SET311
   WT(%SET_SAMH)

:CONT314
   SV(@LASTBND6,OBJ@3)
   MA(*H63,0)
   SV(%BEND6,%%SCHED(#SBND,OBJ@2))
   AO(%BEND6,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND6)
   MA(*H64,%MOVE)
   SV(%WORK14,CLOCK)
   AO(%WORK14,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK14)
   PM(F,HAMMERLE-6)
   PV(F,%WORK14)
   IV(@H6)
   PV(XY(68,14),@H6)
   MR(6,%MOVE)
   MU(3,%MOVE)
   JP(:END)

:ROUTE52                          ;B2_0MM H5
   JC(*H71,:CONT323)
:CONT323
   MR(11,%MOVE)
   MD(3,%MOVE)
   MR(6,%MOVE)
   TP(2,*H72,*H73)
```

```
   MA(*H72,0)
   TP(1,*H73)
   IF(OBJ@3,EQ,@LASTBND7,:SET321)
   WT(%SET_DIFH)
   JP(:CONT324)

:SET321
   WT(%SET_SAMH)

:CONT324
   SV(@LASTBND7,OBJ@3)
   MA(*H73,0)
   SV(%BEND7,%%SCHED(#SBND,OBJ@2))
   AO(%BEND7,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND7)
   MA(*H74,%MOVE)
   SV(%WORK15,CLOCK)
   AO(%WORK15,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK15)
   PM(F,HAMMERLE-5)
   PV(F,%WORK15)
   IV(@H7)
   PV(XY(68,16),@H7)
   MR(6,%MOVE)
   MU(5,%MOVE)
   JP(:END)

:ROUTE53                          ;B2_OMM H4
   JC(*H81,:CONT333)
:CONT333
   MR(11,%MOVE)
   MD(5,%MOVE)
   MR(6,%MOVE)
   TP(2,*H82,*H83)
   MA(*H82,0)
   TP(1,*H83)
   IF(OBJ@3,EQ,@LASTBND8,:SET331)
   WT(%SET_DIFH)
   JP(:CONT334)

:SET331
   WT(%SET_SAMH)

:CONT334
   SV(@LASTBND8,OBJ@3)
   MA(*H83,0)
   SV(%BEND8,%%SCHED(#SBND,OBJ@2))
   AO(%BEND8,*,@@SCHED(#SQTY,OBJ@1))
   WT(%BEND8)
   MA(*H84,%MOVE)
   SV(%WORK16,CLOCK)
   AO(%WORK16,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK16)
   PM(F,HAMMERLE-4)
   PV(F,%WORK16)
   IV(@H8)
   PV(XY(68,18),@H8)
   MR(6,%MOVE)
   MU(7,%MOVE)
   JP(:END)

:ROUTE54                          ;B2_OMM OP2
   JC(*P21,:CONT343)
:CONT343
   MR(11,%MOVE)
```

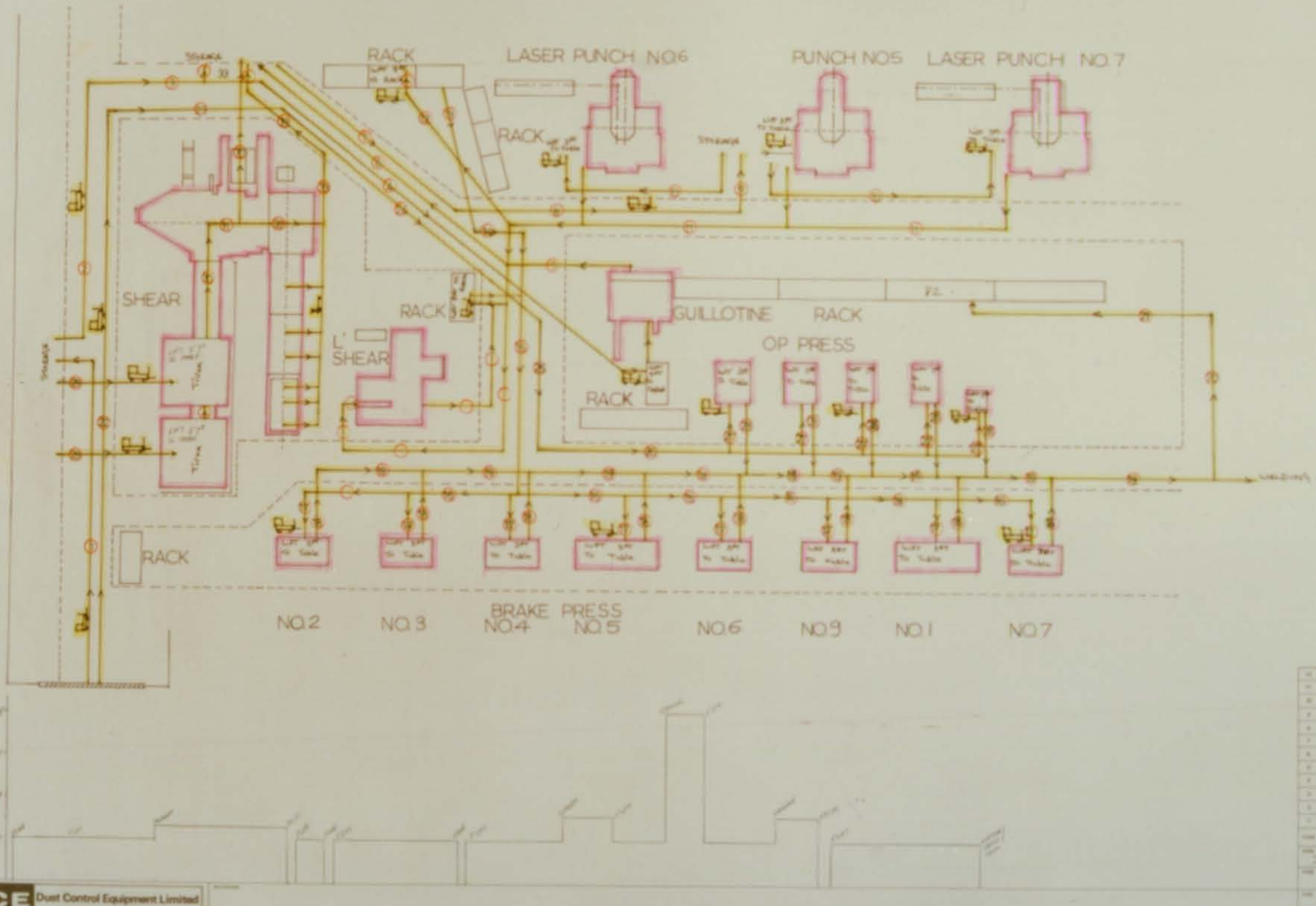
```
   MD(7,%MOVE)
   MR(6,%MOVE)
   TP(2,*P22,*P23)
   MA(*P22,0)
   TP(1,*P23)
   IF(OBJ@3,EQ,@LASTPRS2,:SET341)
   WT(%SET_DIFP)
   JP(:CONT344)

:SET341
   WT(%SET_SAMP)

:CONT344
   SV(@LASTPRS2,OBJ@3)
   MA(*P23,0)
   SV(%PRESS2,%%SCHED(#SBND,OBJ@2))
   AO(%PRESS2,*,@@SCHED(#SQTY,OBJ@1))
   WT(%PRESS2)
   MA(*P24,%MOVE)
   SV(%WORK17,CLOCK)
   AO(%WORK17,-,OBJ%1)
   SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK17)
   PM(F,OP20)
   PV(F,%WORK17)
   IV(@02)
   PV(XY(68,20),@02)
   MR(6,%MOVE)
   MU(9,%MOVE)
   JP(:END)

;--------NEW ROUTE---------------------

:ROUTE60                               ;NO OPERATIONS
   MR(5,%MOVE)
   MD(11,%MOVE)
   IV(@L6)
   PV(XY(10,21),@L6)
   MR(47,%MOVE)
   MU(11,%MOVE)
   SV(%WORK18,CLOCK)
   AO(%WORK18,-,OBJ%1)
   SV(%%PROD_TIMES(#P1ST,OBJ@1),%WORK18)
   PM(F,NO_OPERATIONS)
   PV(F,%WORK18)
   JP(:END)

                                       ;END LOGIC
:END                                   ;END LABEL
   MR(13,%MOVE)                        ;MOVE RIGHT 13
   WT(30)                              ;WAIT 30 SECONDS
   MR(14,%MOVE)                        ;MOVE RIGHT 14
   SV(%WORK19,CLOCK)
   AO(%WORK19,-,OBJ%1)
   SV(%%PROD_TIMES(#PPROD,OBJ@1),%WORK19)
   PM(F,STORE)
   PV(F,%WORK19)
   SV(@@SCHED(#SFLG,OBJ@1),3)          ;SET FLAG TO BATCH ENDED
   IV(@COUNT)                          ;INCREMENT COUNT
   PV(*BATCH,@COUNT)                   ;PRINT VALUE @COUNT ON SCREEN
   PV(*TIME,CLOCK)                     ;PRINT VALUE CLOCK ON SCREEN
   ER                                  ;END ROUTE
```

```
                                                ;APPENDIX 5.3.1
 :ROUTE12                          ;BLASER H1
    JC(*H21,:CONT23)
 :CONT23
    MA(XY(34,4),0)                 ;NEW, MOVE ABSOLUTE

;   MR(11,%MOVE)                   ;DELETED, FOR MOVE ABSOLUTE
;   MD(3,%MOVE)                    ;DELETED, FOR MOVE ABSOLUTE

    MR(5,%MOVE)                    ;DISTANCE CHANGED FROM 6
    TP(2,*H22,*H23)
    MA(*H22,0)
    TP(1,*H23)
    IF(OBJ@3,EQ,@LASTBND2,:SET21)
    WT(%SET_DIFH)
    JP(:CONT24)

 :SET21
    WT(%SET_SAMH)

 :CONT24
    SV(@LASTBND2,OBJ@3)
    MA(*H23,0)
    SV(%BEND2,%%SCHED(#SBND,OBJ@2))
KBCIN  FN=JON\SOC.MDL                    SZ=35564  L=0445  C=001  Help=F1
```

```
                                                ;APPENDIX 5.4.1
 :ROUTE12                          ;BLASER H1
    TP(2,*H22,*H23)
    MA(*H22,0)
    TP(1,*H23)
    IF(OBJ@3,EQ,@LASTBND2,:SET21)
    WT(%SET_DIFH)
    JP(:CONT24)

 :SET21
    WT(%SET_SAMH)

 :CONT24
    SV(@LASTBND2,OBJ@3)
    MA(*H23,0)
    SV(%BEND2,%%SCHED(#SBND,OBJ@2))
    AO(%BEND2,*,@@SCHED(#SQTY,OBJ@1))
    WT(%BEND2)
    MA(*H24,%MOVE)
    SV(%WORK8,CLOCK)
    AO(%WORK8,-,OBJ%1)
    SV(%%PROD_TIMES(#P2ND,OBJ@1),%WORK8)
    PM(F,HAMMERLE-1)
    PV(F,%WORK8)
KBCIN  FN=JON\KANBAN.MDL                 SZ=33821  L=0447  C=074  Help=F1
```

DO NOT SCALE, IF IN DOUBT ASK !

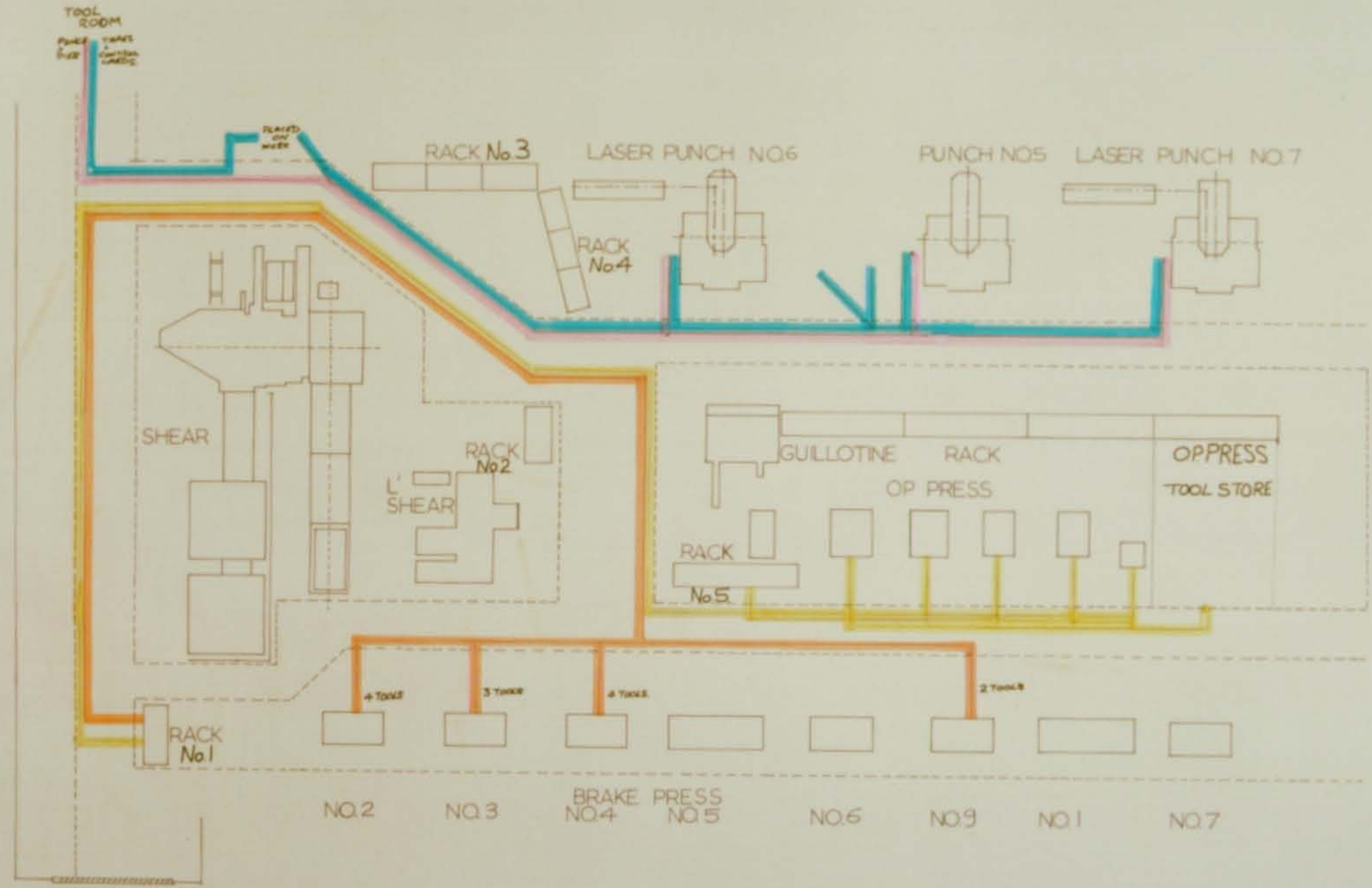
APPENDIX 6.2.1.



FLOW PROCESS CHART
FRONT END OF FACTORY

DCE Dust Control Equipment Limited

DO NOT SCALE, IF IN DOUBT ASK !

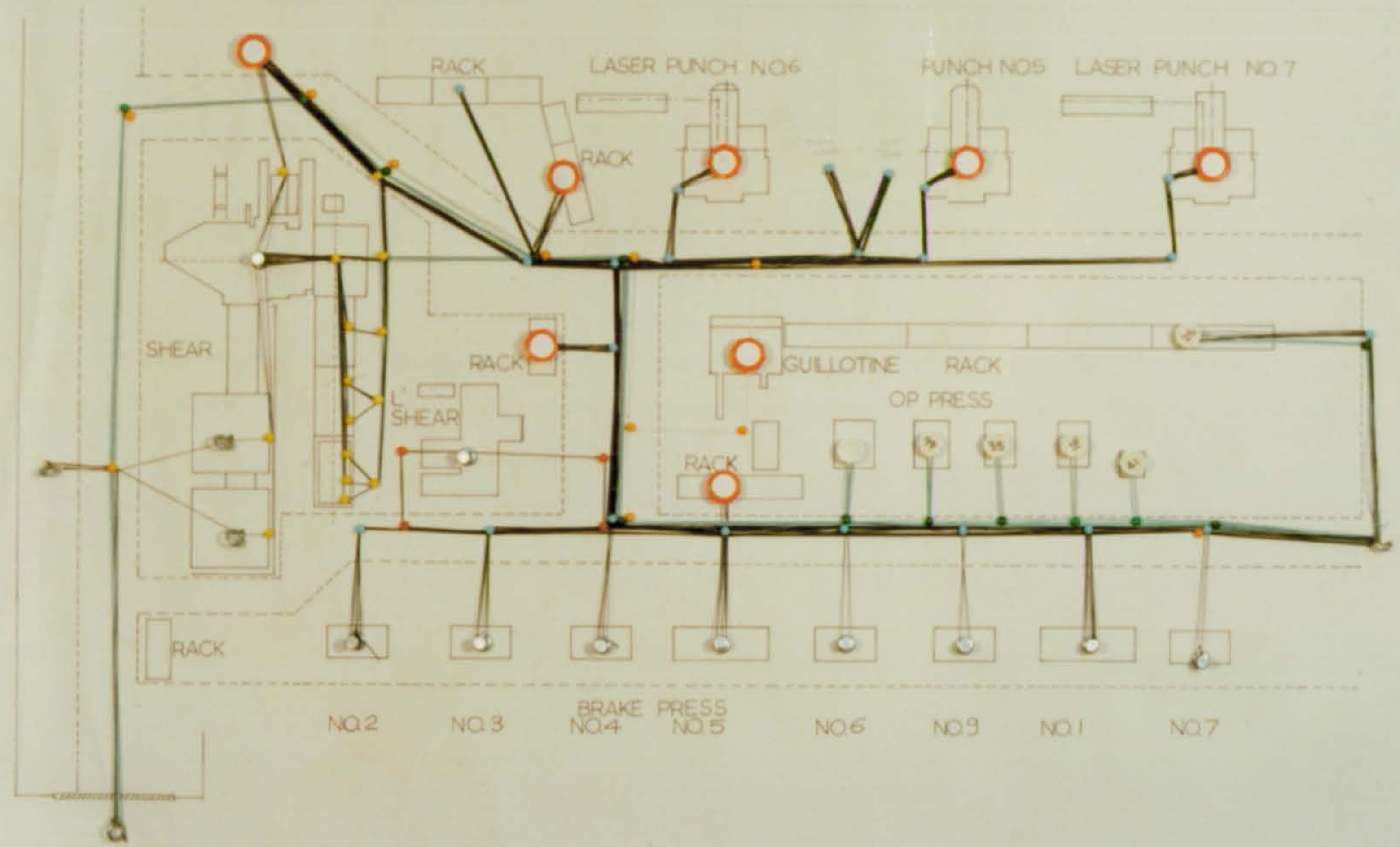


APPENDIX 6.2.2.

DCE Dust Control Equipment Limited

FLOW PROCESS CHART
FRONT END OF FACTORY

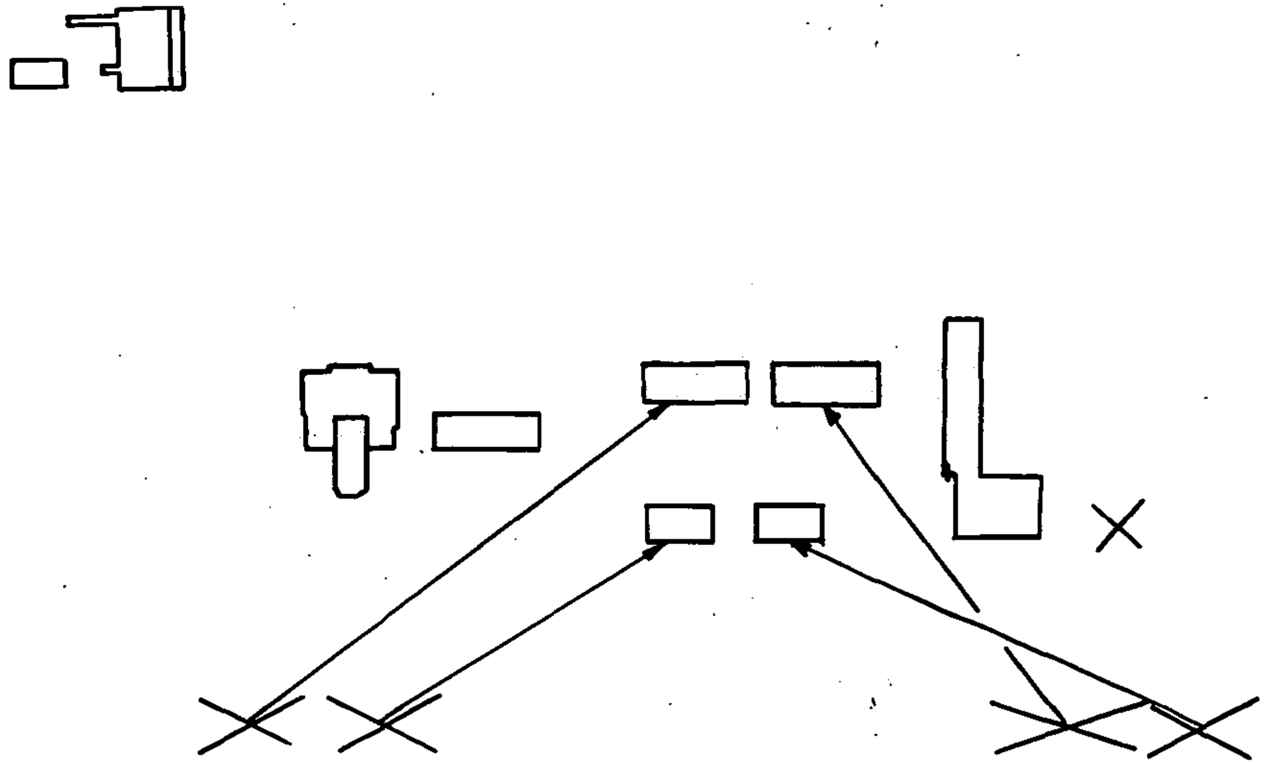
DO NOT SCALE, IF IN DOUBT ASK !



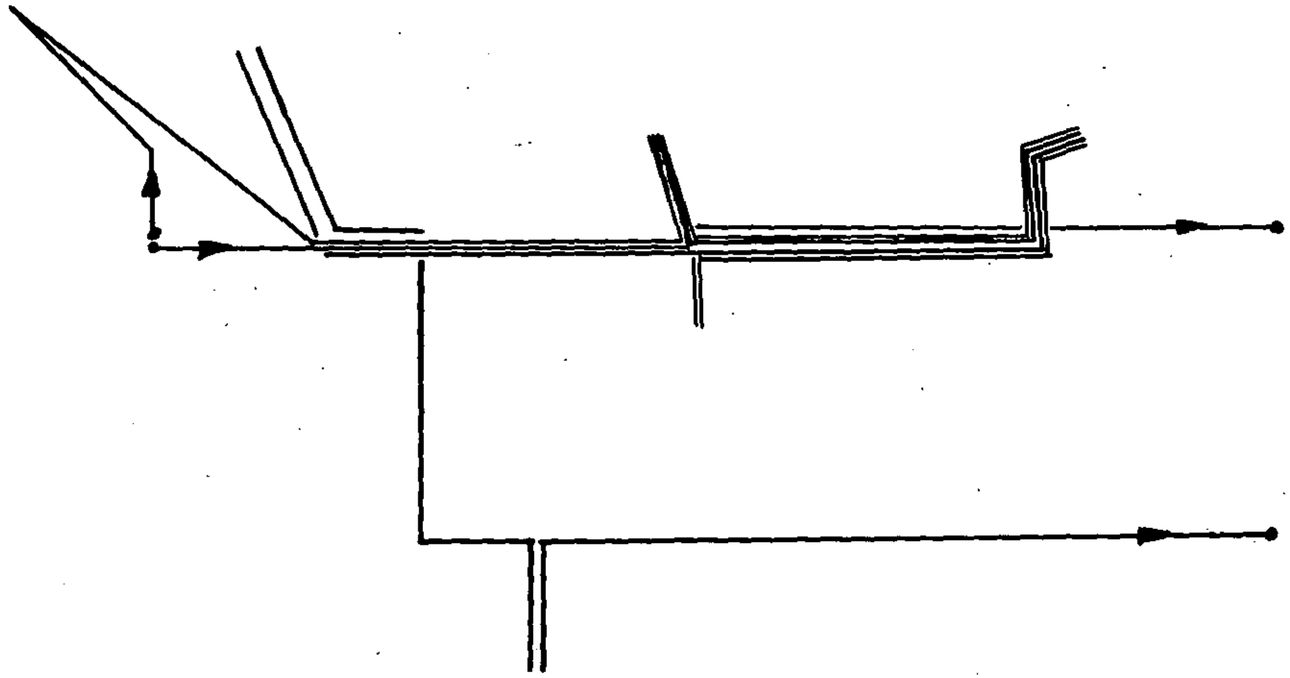
APPENDIX 6.2.3

FLOW PROCESS CHART
FRONT END OF FACTORY

DCE Dust Control Equipment Limited



APPENDIX 6.2.4.



APPENDIX 6.2.5.

APPENDIX 6.3.2.

O.P. PRESS

PURPOSE

What is actually done?

1 - Steel picked up from store by forklift.
or Steel cut by Salvagnini/Stacked/Picked up by forklift.

2 - Steel moved to O.P. Press.

3 - Steel unloaded.

4 - Steel punched etc.

5 - Steel onto pallett.

6 - Steel picked up by forklift.

7 - Steel moved to Paint/Assembly/Welding.

8 - Steel Unloaded

O.P. PRESS

PURPOSE

Why is it necessary at all?

1 - To move the Steel
To produce the right size steel.

2 - To get to the machine.

3 - To free the foklift truck.

4 - This is the operation.

5 - To enable the steel to be moved.

6 - To move the steel.

7 - Movement to the next process.

8 - To release the forlkift for its next job.

O.P. PRESS

PLACE

Where is it being done?
Why is it done at that particular place?

1 - Front end of the shop floor.
That is where the steel is kept on the Salvagnini. This is the prime cutting machine.

2 - Along the shop floor.
Convenience.

3 - Next to the O.P. Press.
Next best place to on the machine.

4 - On the O.P. Press.
Best machine for the job.

5 - Next to the machine.
Direct unloading of machine by operator.

6 - Where the steel is next to the machine.
Thats where the steel is.

7 - On the shop floor.
Convenience.

8 - At Paint Line/Welding/Assembly.
Next Operation

O.P. PRESS

SEQUENCE

When is it done?
Why is it done at that particular time?

1 - When required.
Steel required by machine.

2 - After loading onto forklift.
To keep the forklift occupied.

3 - After it has been moved by forklift.
To free forklift for other work.

4 - When the machine is free.
To obtain maximum utilisation of machine.

5 - After the operation.
To clear room for more work.

6 - When there is a forlift available.
To move work to next station.

7 - When the work is on a forklift.
To free the forlift for more work.

8 - When arrived at next work place.
To free the forklift for more work.

O.P. PRESS

PERSON

Who is doing it?
Could it be done better by someone else?

1 - Forklift Driver - No.
Machine Operator/Forklift Driver - No

2 - Forklift Driver
No.

3 - Forklift Driver/Operator
No

4 - Operator
No

5 - Operator
No

6 - Forklift Driver
No

7 - Forklift Driver
No

8 - Forklift Driver
No

O.P. PRESS

MEANS

How is it being done?
Is there any possibility of doing it more economically in some other way.

1 - Forklift
Trolley/Pallet/Conveyor/AGV/Hand.

2 - Forlkift
Trolley/Pallet/Conveyor/AGV/Hand.

3 - Forklift
Trolley/Pallet/Conveyor/AGV/Hand.

4 - O.P. Press
Bough out

5 - Forklift
Trolley/Pallet/Conveyor/AGV/Hand.

6 - Forklift
Trolley/Pallet/Conveyor/AGV/Hand.

7 - Forklift
Trolley/Pallet/Conveyor/AGV/Hand.

8 - Forklift
Trolley/Pallet/Conveyor/AGV/Hand.

APPENDIX 6.3.4.

BEHRENS PUNCH / HEMMERLE BRAKE PRESS, PUNCH DATA

LOW GAUGE 1.5mm

BEHRENS NO. 6

- Punch/Laser
- 18 Tool Turret - Installed in February 1983

Routing for:

fins (flat/rolled), Hoppers (H2/H1), Seal Frames (H3/H1/H3)/Shaped Parts.

HAMMERLE NO. 3

- Brake Press
- 200 Ton Capacity - Installed in 1981
- Manual Setting

Tools:

2 off. Seal Frame for 1.6mm gauge.
1 off. Flaring Tool for 1.5mm gauge.

Links required with: - Hammerle No. 1
No. 2
No. 7
No. 9

- O.P. Press

- Flat Work

Work:

- DLMV Weather Cowls, requires wide jaws. (1 Hw 3, 1 HW 4)
- Seal Frames

These jobs only.

LOW GAUGE 1.6MM

Behrens No. 5

- Punch/Non-Laser
- 24 tool Turret - Installed in July 1977

Links required with MITRE/NIBBLER - HAMMERLE NO. 1
(15 350V 2, UMA Side outlet, Base Front Panel Doors).

No. 2

(5 ADT 1B/1C, ADT Motor Base Sides).

Hammerle No. 1

- Brake Press
- 100 Ton Capacity - Installed in 1983
- N.C. Control (Magnetic Tape)
- 3 Point Bend

Links required with: Hammerle No. 2 - UMA Hoppers
No. 3 - Seal Frames
No. 9 - UMA Side Panel
(270 M 4BX)

Hammerle No. 2

- Brake Press
- 100 Ton Capacity
- Manual Setting

Tools: 4 off. Old type Dalamatic 600 insert for 1.5mm gauge. These can also be used on Hammerle No. 4, prefered on that machine.

Work: Hoppers, the tool only fits this machine.
Sealer Gear, flattens the handles.

Hammerle No. 7 (possible to dispose of this M/C)

- Brake Press - Installed in 1980
- 50 Ton Capacity
- Manual Setting

Work: Small parts.

Lin required with ROLLING, Inlet Plate G3 60Hz. (13 C6)

Hammerle No. 9

- Brake Press - Installed in 1977
- 175 Ton Capacity
- Cushioned Beds
- Back Stops Fitted
- Manual Setting

Tools: 1 off. Joggling Tools, for 1.5mm gauge.
1 off. "A" Stiffener Tool, for 1.5mm gauge.

Work: Case Side Stiffeners

Links required with: O.P. Press
Flat Parts

HIGH GAUGE 1.6mm

Behrens No. 7

- Punch/Laser
- 18 Tool Turrett - Installed in June 1983

Hammerle No. 4 (possible to dispose of this M/C)

- Brake Press - Installed in 1962
- 175 Ton Capacity
- No Back Stops
- Manual Setting

Tools: 4 off. Old Type Dalamatic 600 Insert Tools for 1.5mm gauge.

These can also be sued on Hammerle No. 2

Hammerle No. 5

- Brake Press - Installation 1983
- 100 Ton Capacity
- N.C. Control (Magnetic Tape)
- 3 Point Bend

Hammerle No. 6

- Brake Press - Installed in 1977
- 100 Ton Capacity
- Plug Board

Work: Fold of 8mm Top Door Frames (77 C10 1)
Small Edge Fold Pedistals (4 C10 1)
1 XH 15 Exit Header Fastener LOW GAUGE

Links required with: O.P. Press

N.B. HAMMERLES require 3 x width of gangway for W.I.P.
HAMMERLES N.C. average 30 jobs per shift - 60 jobs total.
BEHRENS average 10 jobs per shift - 30 jobs total.
Cannot put HAMMERLE NO. 6 on the Laser Line.

APPENDIX 6.4.0.

ADDITION TIME CARD

PART NO./DESCRIPTION: 2 450R RUNNER SUPPORT (52 OFF)

DATE/TIME MOVED	MOVED TO: PLACE	INITIALS
04.04.90 0915 HRS	SALVAGNINI	
09.04.90 0700 HRS	BEHRENS	
01.05.90	FOLD	
	PAINTING	
08.05.90	RETRIEVED FROM STORES	
	PLEASE RETURN FROM WELDING TO J. BEVAN (D. TAYLOR'S DEPT.)	

APPENDIX 6.5.2.

MACHINERY MOVEMENT COSTS

BEHRENS	£ 8K EACH	X 3
HAMMERLE	£ 1.5K EACH	X 8
GUILLOTINE	£ 1K EACH	X 1
O.P. PRESS	£ 1.5K EACH	X 5
SERVICES	£ 5K	
TOTAL	£49.5K	

Ref: Ivor Reynolds - Production Engineering
Steff Sulyma - Production Engineering

Date: 04/02/90

APPENDIX 6.5.3

RECOMMENDATIONS FOR SHOP FLOOR IMPROVEMENTS

As a result of following a weeks worth of jobs through the first 4 sections of the production facility, the following points became apparent:

- **PAYMENT**

  Different jobs have different payments allocated to them and as such the better paid easy jobs always get done first. With difficult low paid jobs only being done as and when they are being progressed (urgently required). It is proposed, that an order be given to the jobs, so that all the jobs get done in a regular manner. This will eliminate a lot of confusion and help get the jobs done on time.

- **SHIFT PATTERNS**

  In the 4 sections examined there are 3 different shift times leading to a large fluctuation of W.I.P. between the sections. Giving an erratic job flow through the production facility. It is proposed to match the laborious sections outputs to minimise the amount of W.I.P. and smooth job flow, by evaluating the necessary shift times required. This will improve the work flow and minimise W.I.P.

- **DEVELOPMENT WORK**

  New jobs often require amendments and held up the job flow through the production line. It is proposed to run new jobs\trials on the weekends with all the necessary staff present being paid the necessary remuneration. Enabling the production line to run for its full scheduled time per week.

- ON LINE SHORTAGES

Assembly often requires parts urgently due to shortages, when these are placed as priority jobs onto the production line the scheduled jobs are delayed. It is proposed, that a specific line be set up for shortages, working on normal jobs when there are not shortages. Paying the operators their top rate to ensure a fast turn around of jobs. This will help to minimize the setting times on the machines not involved, maximising output.

- MATCHED QUANTITIES

Sub-assemblies often do not have the required numbers of parts to complete the required numbers of units. It is proposed that through the use of specialised pallets and the loading of matching quantities onto the production line to eliminate this. Saving time spent obtaining the correct parts.

- PROGRESS CHASING

Section Leaders are being used to progress jobs through the production process. Whilst doing this they cannot be doing their work of leading the section. It is proposed, that through the ordering of the job flow through the manufacturing process, the Section Leaders work will revert back to that of leading the section. A specific progress chaser not being required, leading to a more efficient use of the resources available.

- JOB PACK DRAWINGS

Currently drawings are kept, marked-up, on the shop floor. This leads to time being spent finding the relevant drawing and ensuring that it is current. It is proposed to place the relevant drawings into the job packs in the Production Control Deptartment and for them to be maintained by this department. This will free the Section Leader to lead his section and the number of faults being produced will be reduced, due to up to date drawings being used.

- SPECIALISED PALLETS

At present almost all jobs are processed on standard pallets no allowance being made sub-assemblies or usually shaped components. It is proposed, to review the number of components that this technology can be applied to, then design and implement the necessary pallets. This will enable groups of components to be more readily identified and obtain a higher utilisation of the shop floor available.

- COMPLETION OF ORDER

An order is "live" only as far as welding, it is assumed that it will find its own way from welding, through painting and into the stores. It is proposed, to extend the live order to stores, so that it is known when a job has been completed and ready to be used. This will prevent the re-ordering of work already on the production facility and keep control of its process.

- ROUTE CARDS

These move through the manufacturing process with the job from its inception to its arrival at the stores before assembly. Where they are destroyed. It is proposed that by returning these route cards to the issuing authority a closed loop system would be instigated rather than the open loop system of present. If necessary utilising a bar code system, this would enable more accurate records to be maintained and save a great deal of wasted production

- JOB PACKS

These are currently split, one folder going to the section leaders desk, the other staying with the job. The section leader has to locate both parts of the paperwork before work can commence on the job. It is proposed to keep the job pack complete with the job. Installing and maintaining an order in the storage space available. With the objective of saving the Section Leaders time.

- SALVAGNINI PROGRAMME

The Salvagnini Shear job loading is programmed in a weekly quantity, but that quantity refers to 4 weeks jobs. It is proposed that where possible the programmed week is kept to that weeks jobs. Placing jobs produced not required for that designated week into a specified rack for retrieval in the relevant week. Whilst moving the urgent work from the Shear to the start of the next process at the earliest opportunity. This will enable the processes down stream of the Shear the chance of producing a weeks jobs in a week and help improve the movement of jobs through the manufacturing process.

- WORK AWAITING MACHINING

Whilst jobs are waiting between operations, they are stored wherever there is room for them. It is proposed, that the racking\floor space available is allocated for specific jobs. Thus minimising the time spent looking for jobs and bring order to a random process.

- WORK IN WELDING

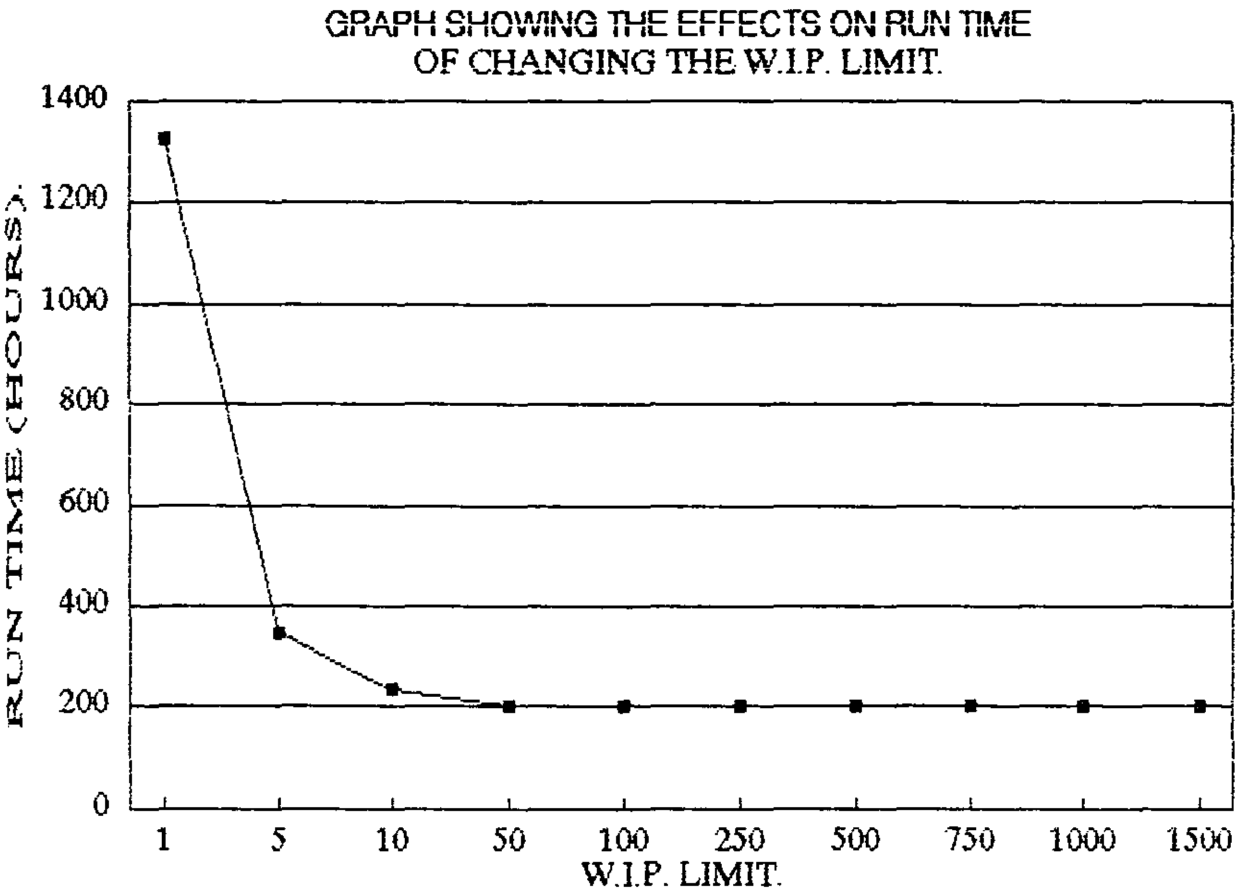
Non-urgent jobs are left until last, resulting in some jobs taking excessive times to be processed. It is proposed to process jobs using "first in, first out" method and separate line for on line shortages etc. Thus minimising the time spent looking for jobs and floor space used to store jobs.

- WELDING AND FETTLING

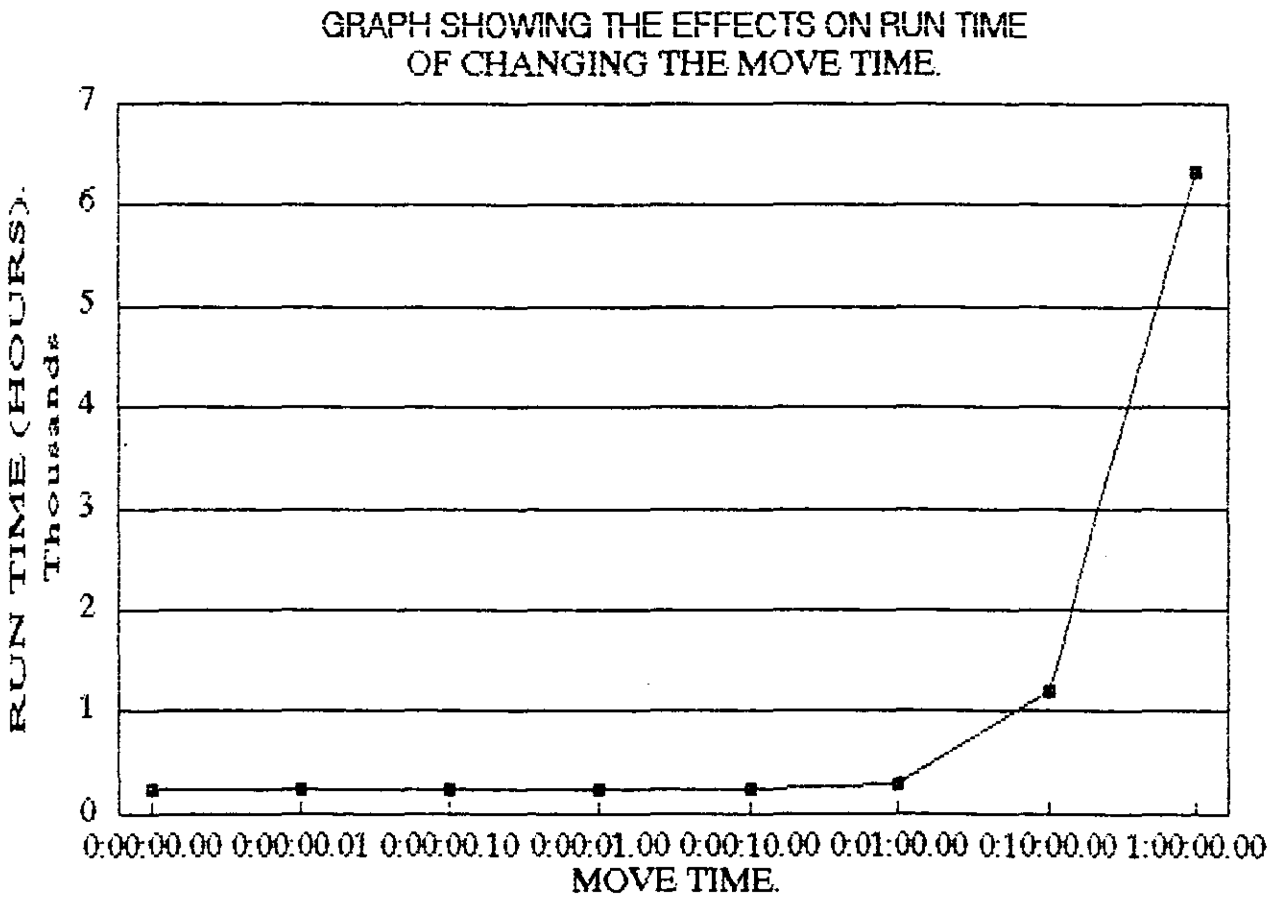
Welding is carried out on the whole of a job and then at some later date that job is fettled. This leads to a high W.I.P. in the welding section and under utilisation of floor space. With fettling done largely on the floor and large amount of W.I.P. in the area it tends to become very dirty. This leads to complaints of physical discomfort and pain, due to the constant bending over and movement of W.I.P. It is proposed to move individual fettling booths to be adjacent to individual welding booths and make the process more ergonomically acceptable. Changing the working practice so that as a batch of work is welded it is fettled. Thus minimizing the W.I.P., optimising production rate and floor space.

The above points will still be present as and when the move to MRPII is made. Implying that dramatic long term savings can be made easily and with little expense.

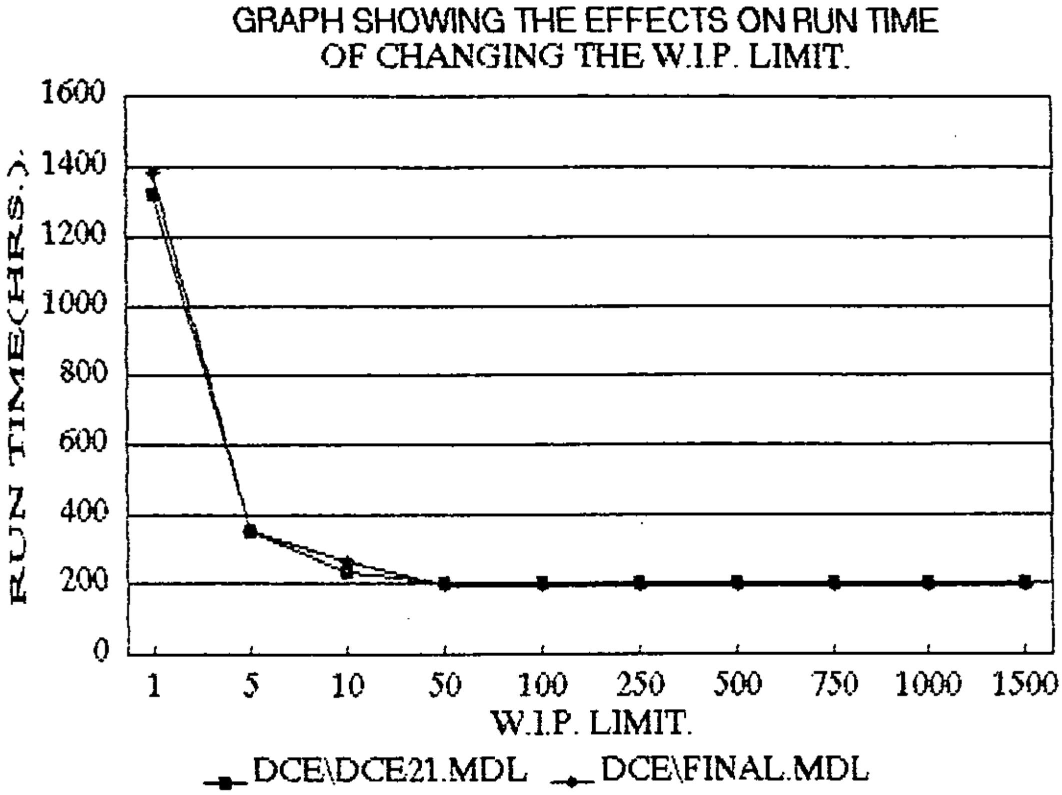
GRAPH 3.3.1.



GRAPH 3.3.2.

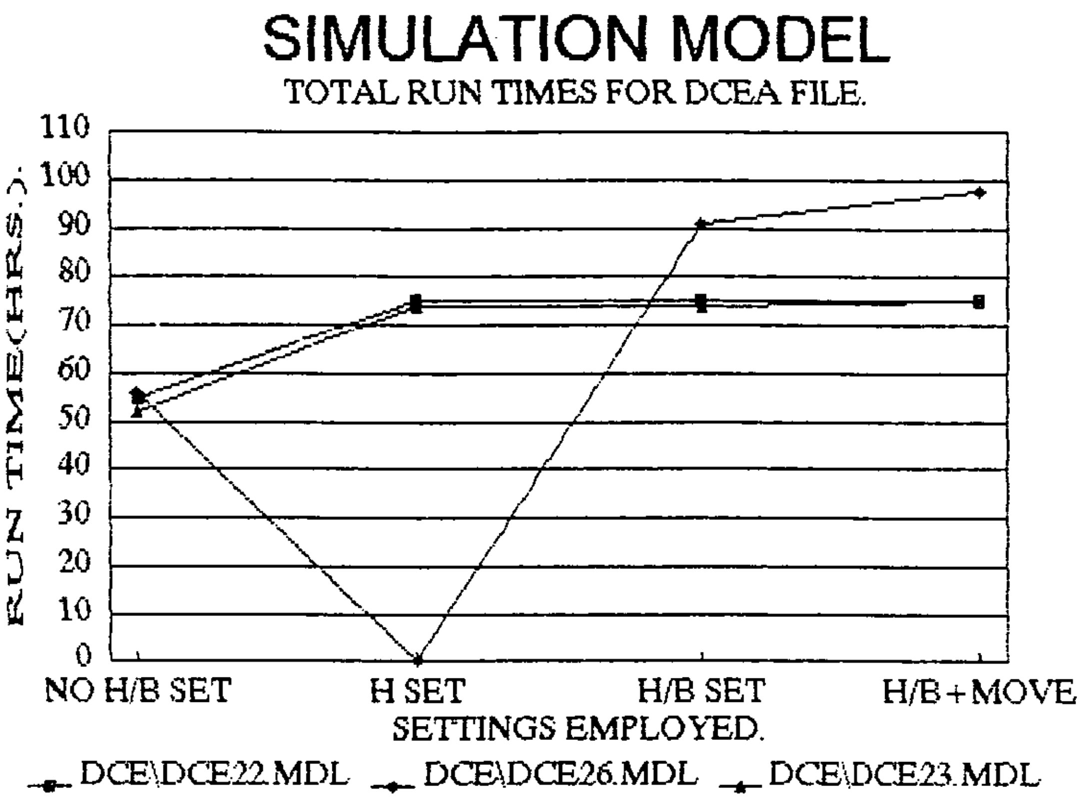


GRAPH 3.3.3.

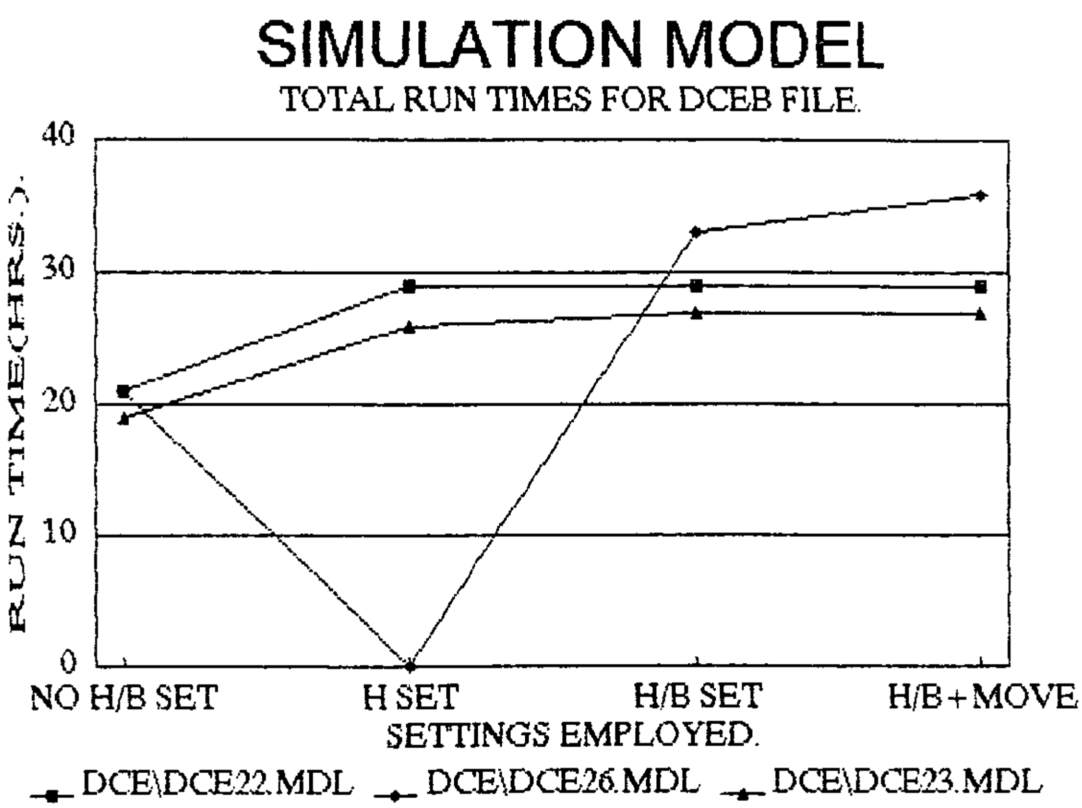


GRAPH SHOWING THE EFFECTS ON RUN TIME
OF CHANGING THE W.I.P. LIMIT.
RUN TIME(HRS.)
1600
1400
1200
1000
800
600
400
200
0
1
5
10
50
100
250
500
750
1000
1500
W.I.P. LIMIT.
DCE\DCE21.MDL
DCE\FINAL.MDL

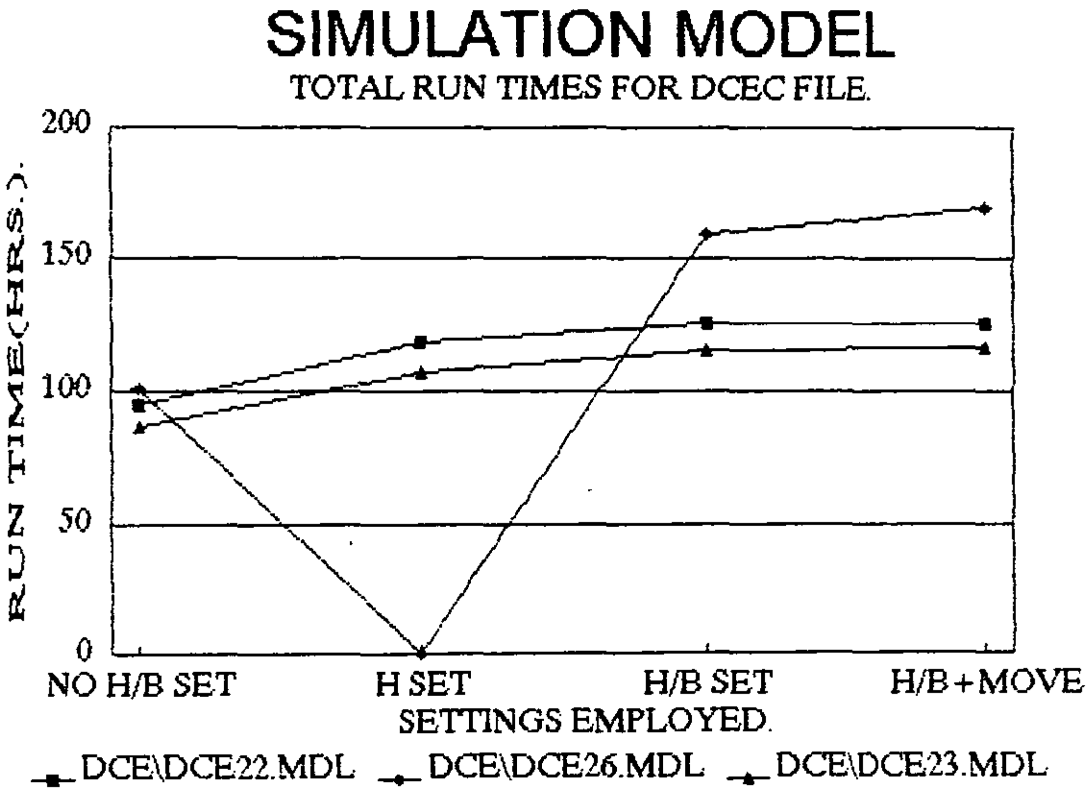
GRAPH 3.3.4.



GRAPH 3.3.5.

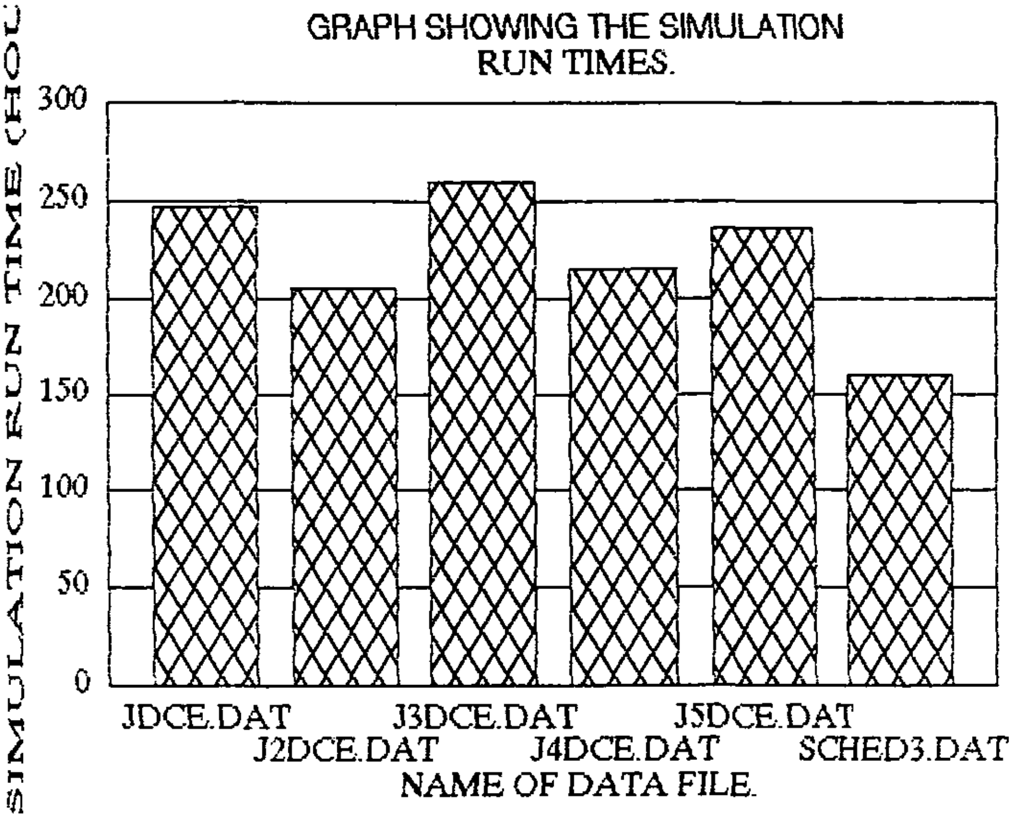


GRAPH 3.3.6.

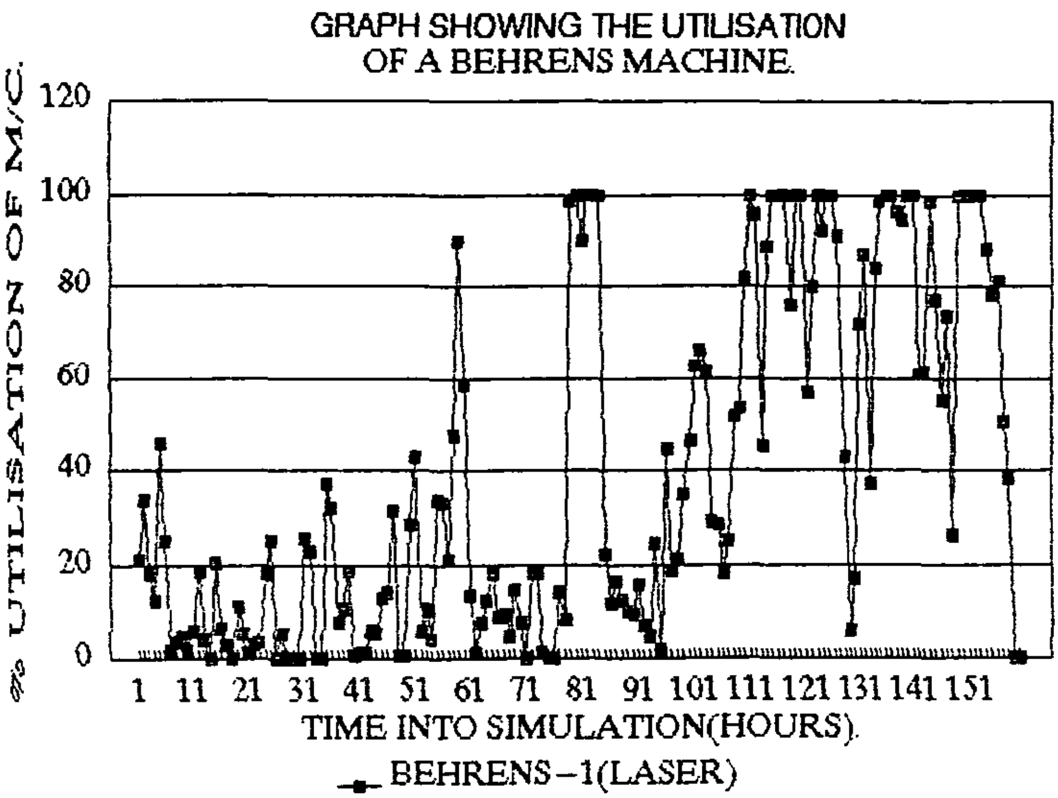


SIMULATION MODEL
TOTAL RUN TIMES FOR DCEC FILE.
200
150
100
50
0
RUN TIME(HRS.)
NO H/B SET
H SET
H/B SET
H/B+MOVE
SETTINGS EMPLOYED.
DCE\DCE22.MDL
DCE\DCE26.MDL
DCE\DCE23.MDL

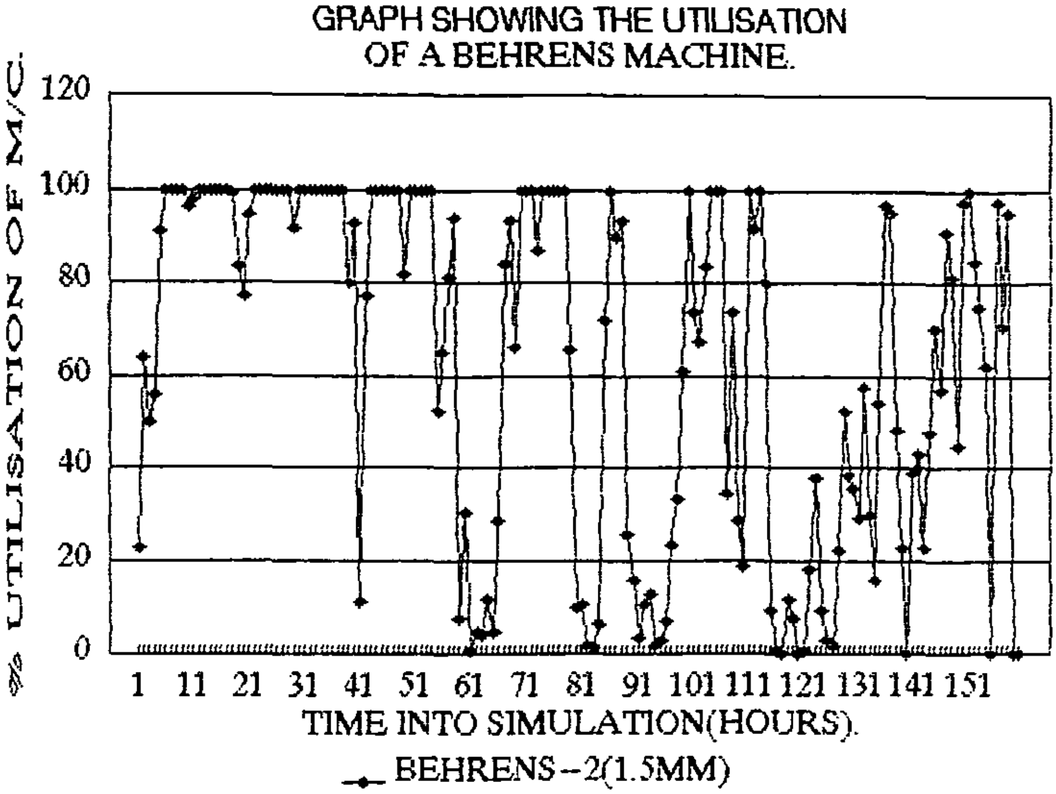
GRAPH 4.3.1.


GRAPH SHOWING THE SIMULATION
RUN TIMES.
SIMULATION RUN TIME (HOU
300
250
200
150
100
50
0
JDCE.DAT
J2DCE.DAT
J3DCE.DAT
J4DCE.DAT
J5DCE.DAT
SCHED3.DAT
NAME OF DATA FILE.

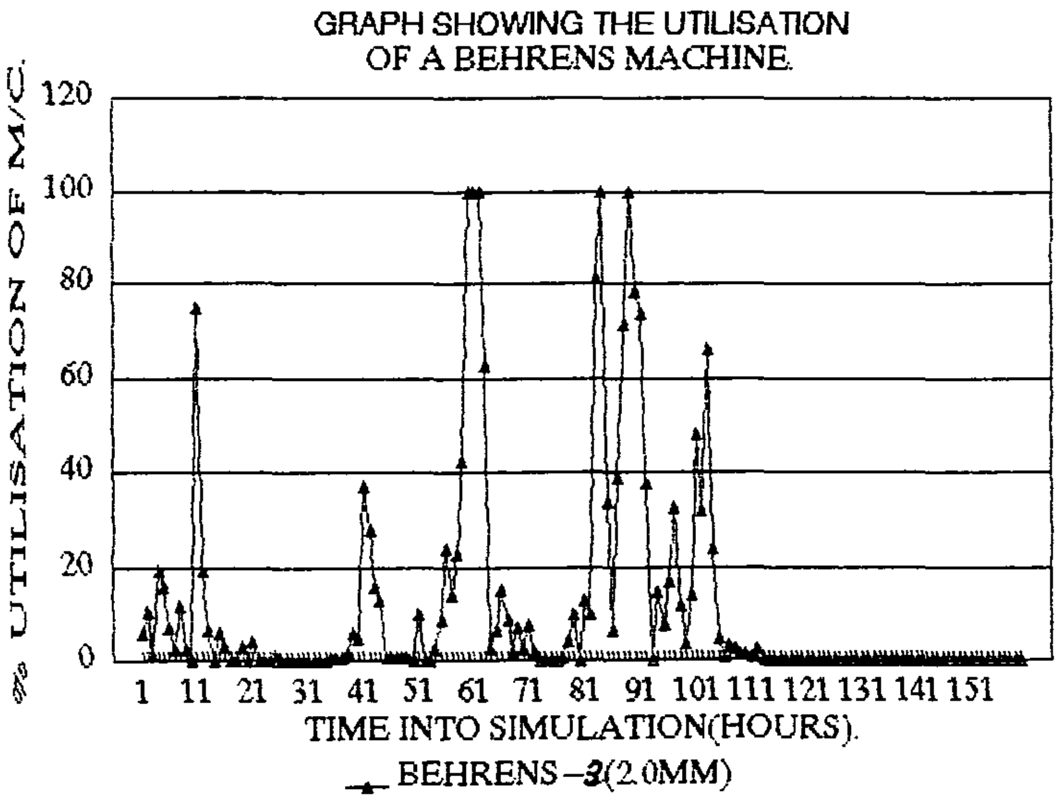
GRAPH 4.3.2.


GRAPH SHOWING THE UTILISATION
OF A BEHRENS MACHINE.
% UTILISATION OF M/C.
120
100
80
60
40
20
0
1 11 21 31 41 51 61 71 81 91 101 111 121 131 141 151
TIME INTO SIMULATION(HOURS).
BEHRENS-1(LASER)

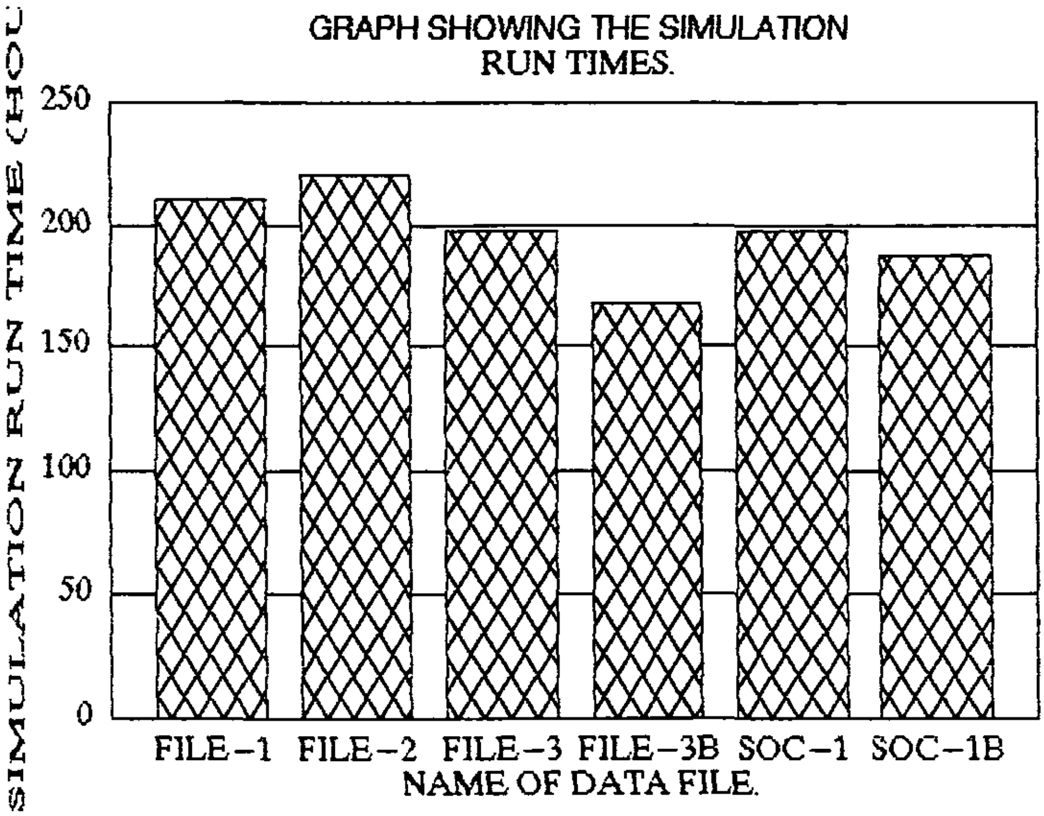
GRAPH 4.3.3.



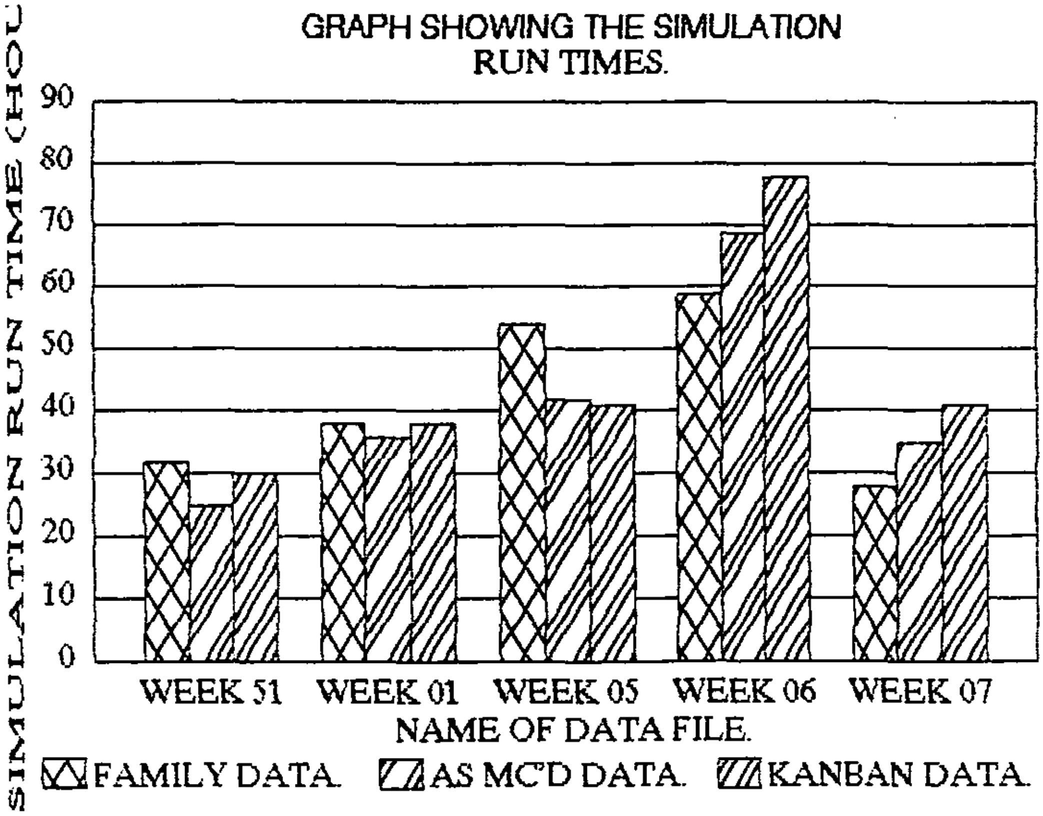
GRAPH 4.3.4.



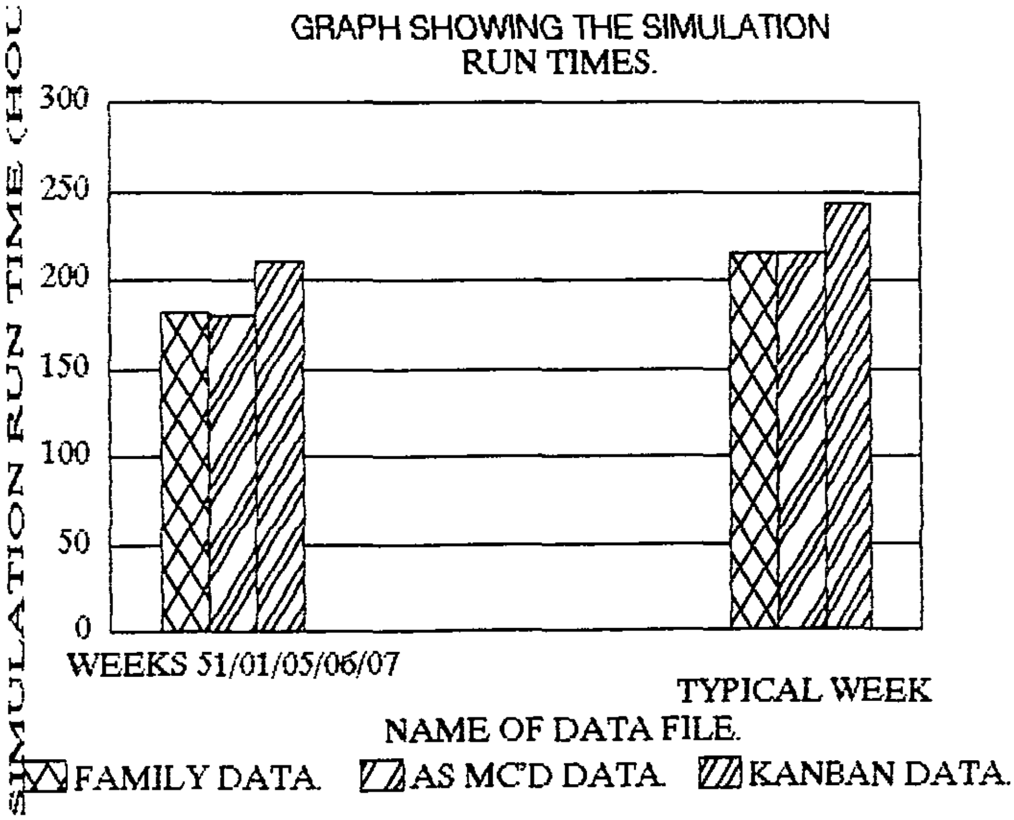
GRAPH 5.3.1.


GRAPH SHOWING THE SIMULATION
RUN TIMES.
SIMULATION RUN TIME (HOU
250
200
150
100
50
0
FILE-1
FILE-2
FILE-3
FILE-3B
SOC-1
SOC-1B
NAME OF DATA FILE.

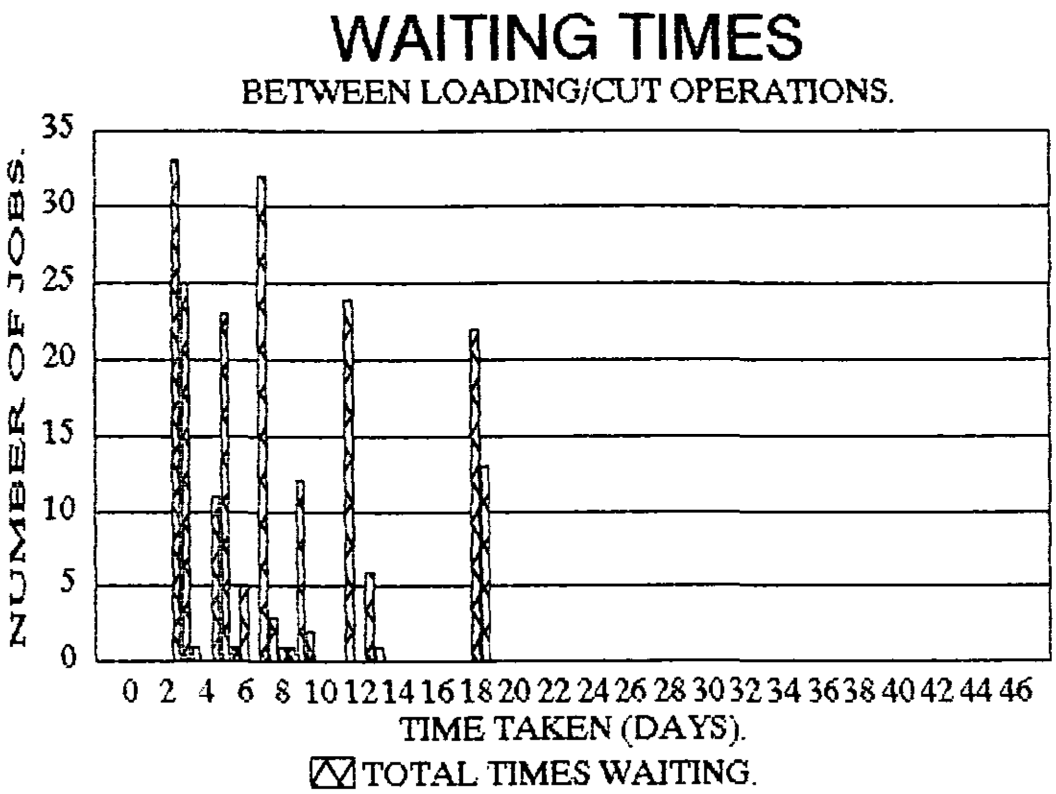
GRAPH 5.3.2.



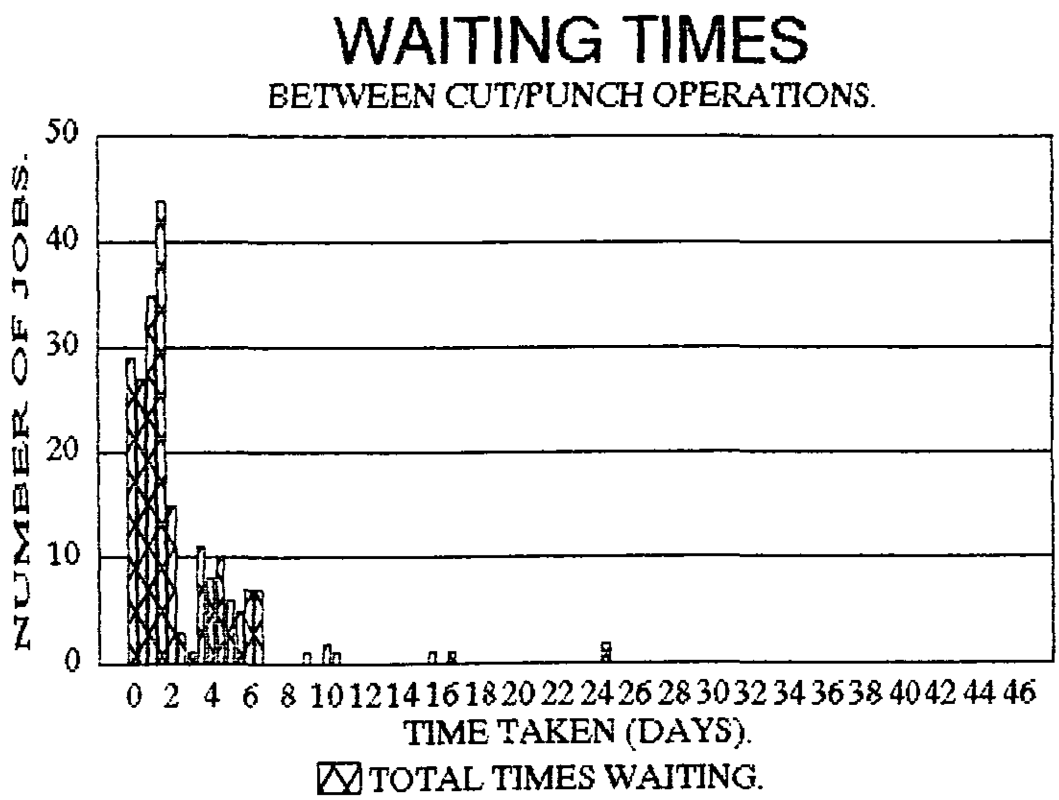
GRAPH 5.3.3.



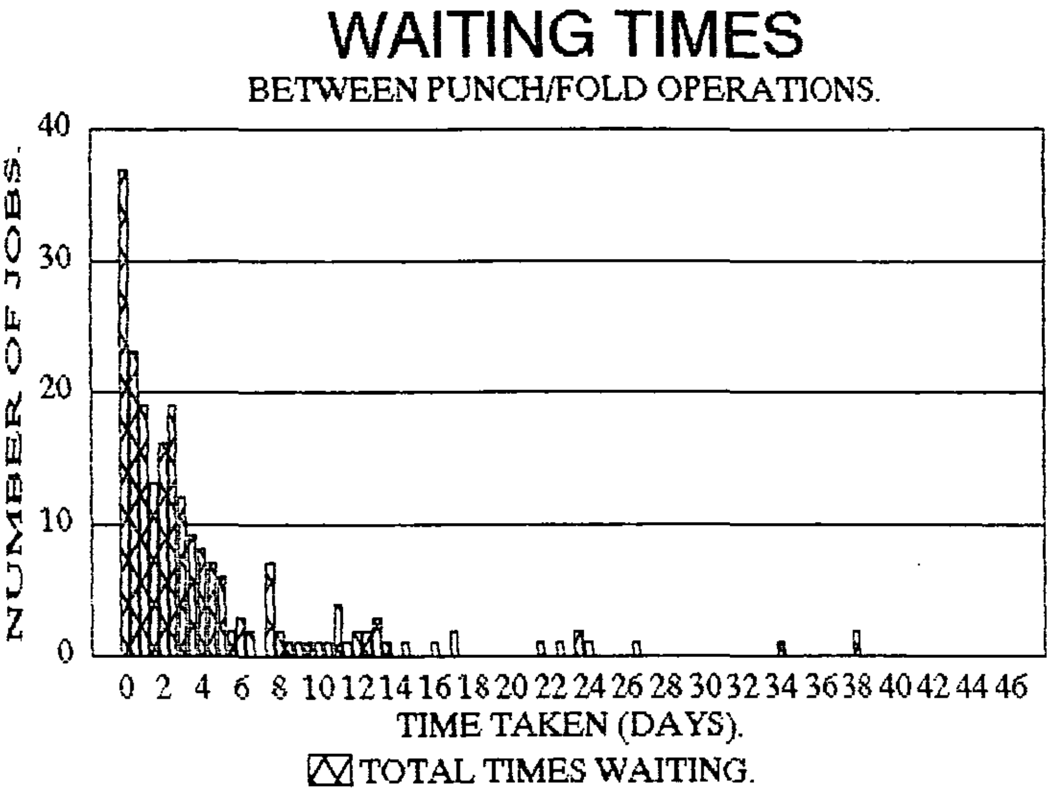
GRAPH 6.4.1.



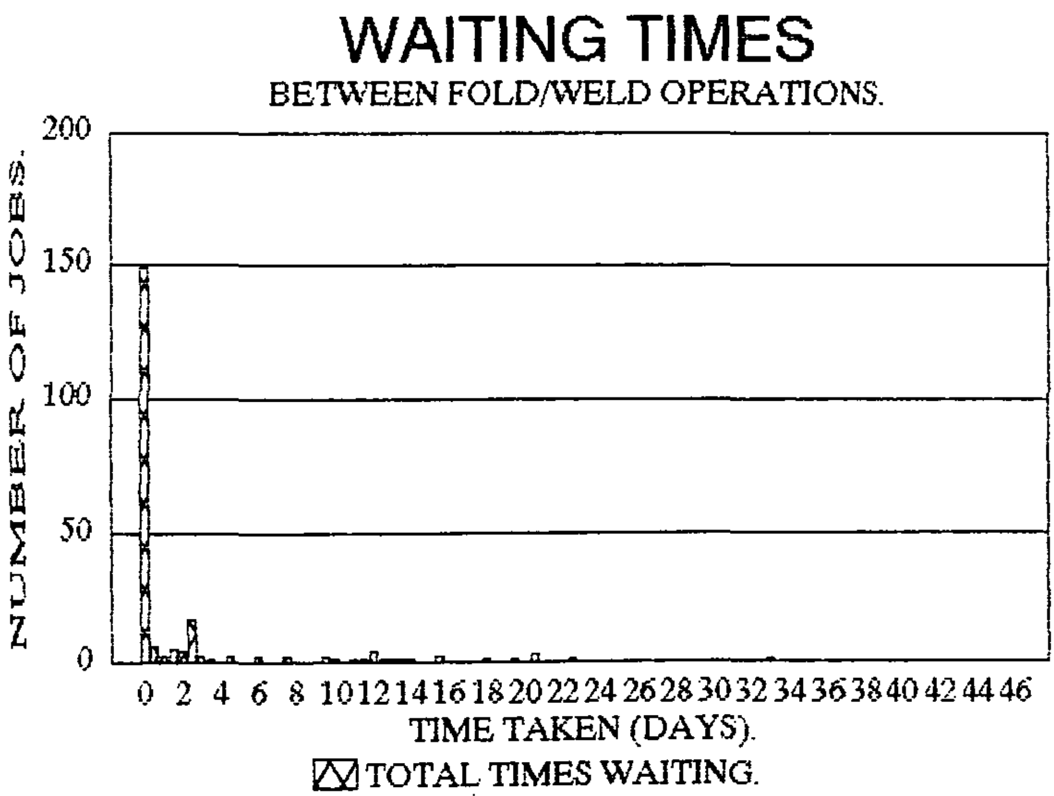
GRAPH 6.4.2.



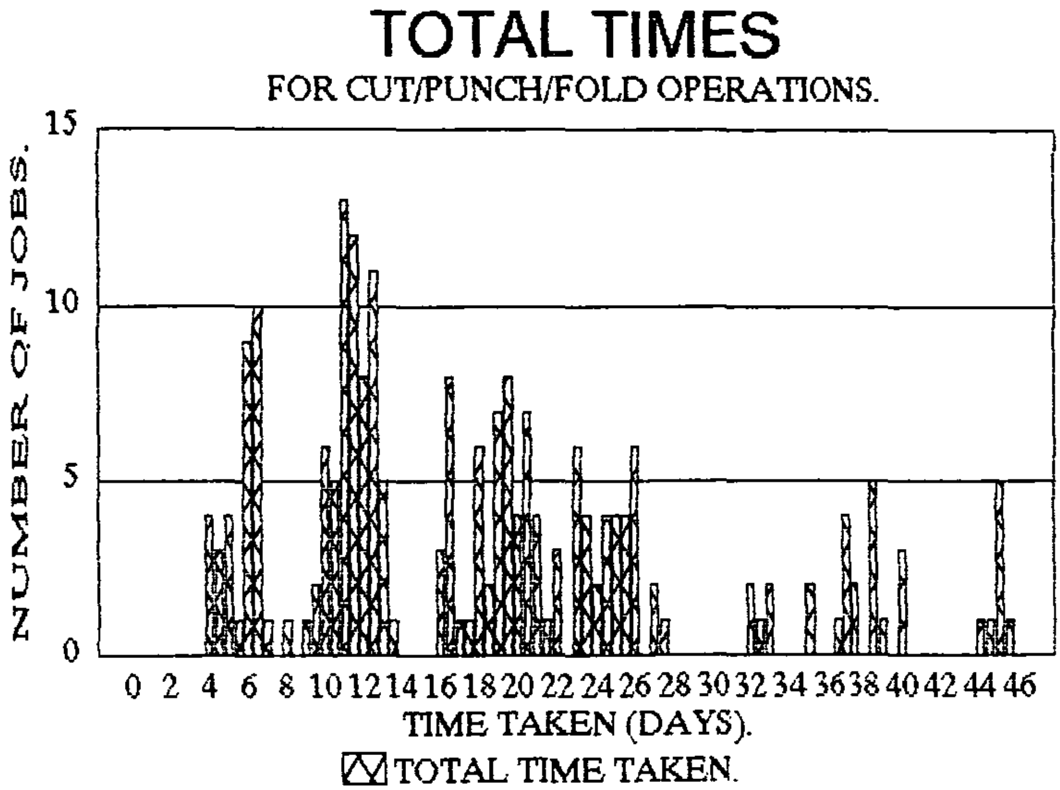
GRAPH 6.4.3.



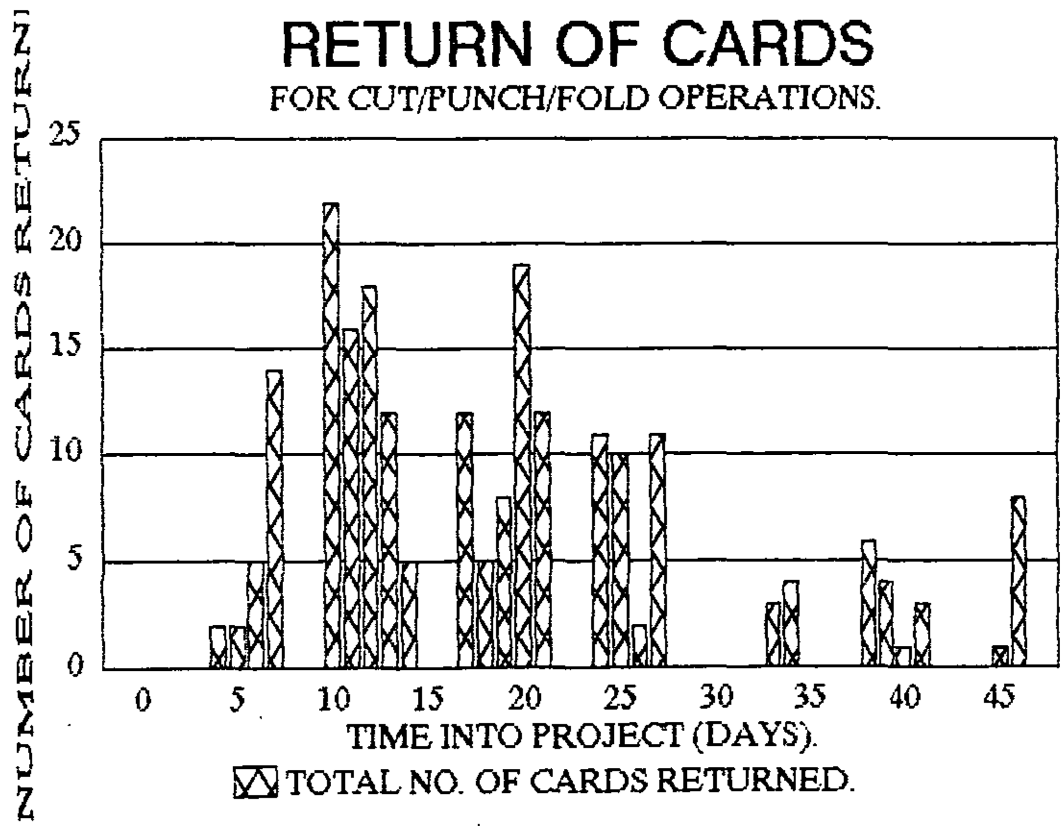
GRAPH 6.4.4.



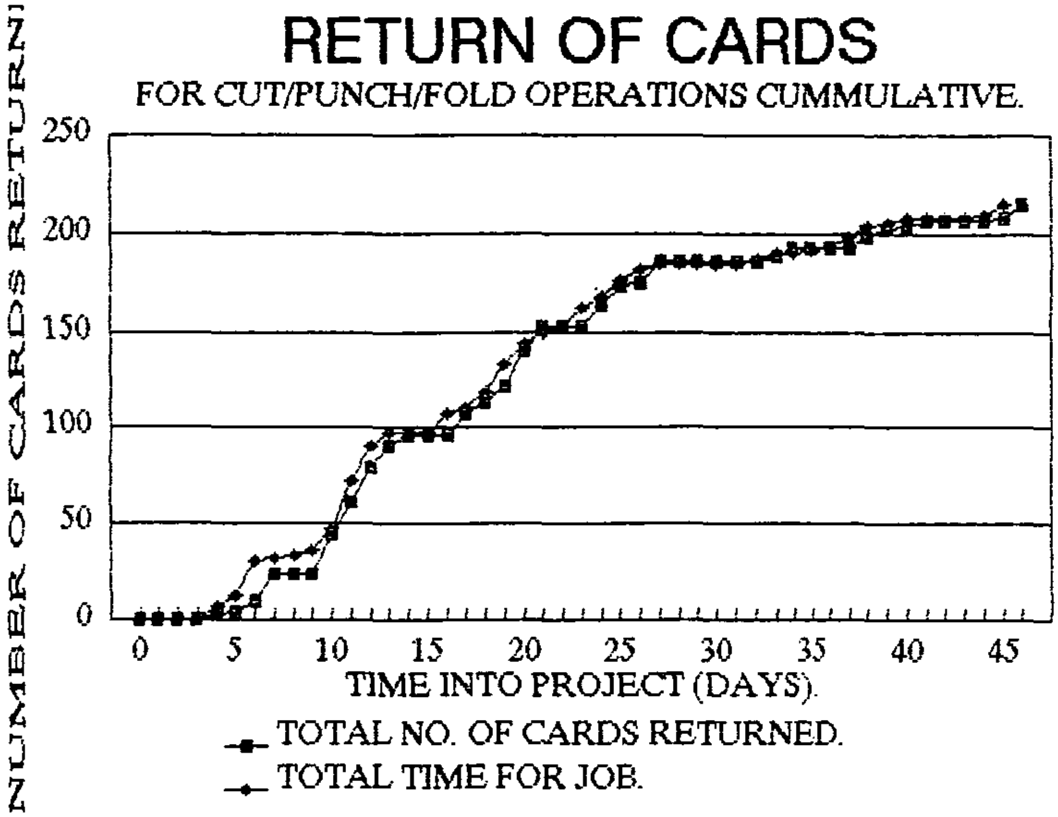
GRAPH 6.4.5.



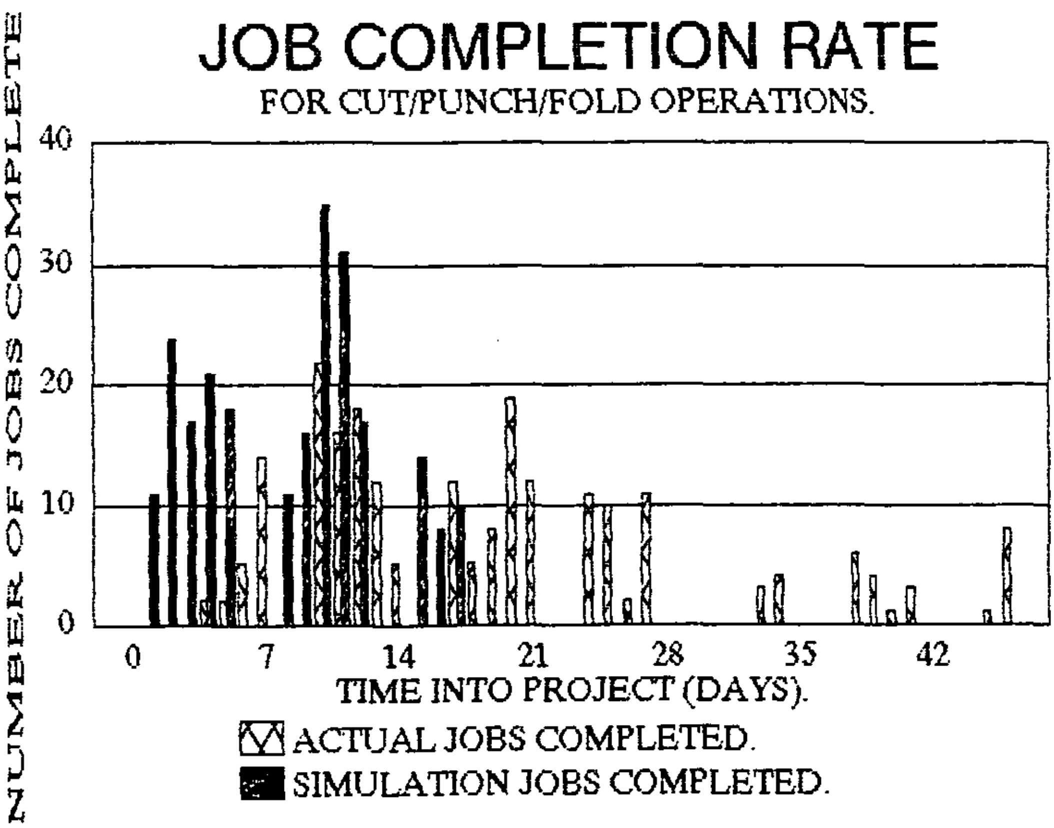
GRAPH 6.4.6.



GRAPH 6.4.7.



GRAPH 6.4.8.



C H A P T E R 11

R E F E R E N C E S \ B I B L I O G R A P H Y

11.1 REFERENCES

1 - Simulation - Taking the Risk out of Investment

By D.A. Hearn

Proc. 4th Int. Conf. Simulation in Manufacturing
275 - 283 November 1988

2 - Design of Just-in-Time Manufacturing Systems

By A. Garside

Proc. 4th Int. Conf. Simulation in Manufacturing
239 - 249 November 1988

3 - Principles of Engineering Production - 2nd Edition

By A.J. Lissaman \ S.J. Martin

Published by Hodder and Stoughton

4 - Materials and Processes in Manufacturing

By E. Paul Degarmo

Published by Collier Macmillan

5 - Manufacturing Simulators Save Time, Provide Good Data For Layout Evaluation

By Angel A. Almadouar

I.E. June 1988

6 - 1.2.3. The Complete Reference

By Mary Campbell

Published by Osbourne McGraw-Hill

7 - Lotus 1.2.3 Release 2.0.1

Reference Manuals

By Lotus Development Corporation

8 - Lotus 2.1.3 Release 3.0

Reference Manuals

By Lotus Development Corporation

9 - P.C. Model Handbook

By Simulation Software Systems

10 - Simulation Models Factory Floors

By Partha Protim Bose

Published in : American Machinist, August 1988

11 - Introduction to Simulation and Slam II

By A.A.B. Pritsker

Published by Halsted Press

12 - Simulation Software Products for Analysing Manufacturing Systems

By S. Wali Haider and Jerry Banks

Published in: I.E., July 1986

13 - Pitfalls in the Simulation of Manufacturing Systems

By Averill M. Law and Michael G. McComas

Published in: Proceedings of the 1986 Winter Simulation Conference.

14 - Introduction to Work Study.
By the International Labour Office.
Published by Atar Geneva (Switzerland)

11.2 BIBLIOGRAPHY

1 - Simulation Scores on the Shop Floor
Real - Time Scheduling\Control

By Brian Icellock

Published in: Engineering Computers Magazine
May 1989

2 - Simulating Just-in-Time Systems

By D. Fallon and J. Browne

Published in: International Journal of Operations
and Production Management, No. 6,
1988, pp 30 - 45, 6 figs.

3 - Group Technology in Production Management - A Tool
to Simplify some Scheduling Problems

By J.M. Progh

Published by: Inria-Lorraine, Chateau De Montet,
54550 Vandeouvre Les Nancy

4 - On Dynamic Routing in FMS

By: Yong F. Choong and Oded X. MaiMon

Published by: IEEE 1986
Ref: CH2282-2/86/0000/1476$01.00

5 - Decision - Aid in Job Shop Scheduling
A Knowledge Based Approach

By J. Erschler and P. Esquirol

Published by: IEEE 1986
Ref: CH2282-2/86/0000/1651$01.00