AN APPLICATION OF SILVER MEAL HEURISTIC TO MRP LOT SIZING DECISIONS AT TÜRK TRAKTOR FABRIKASI

MBA THESIS

BERKHAN N. ESMER

JUNE 1995

AN APPLICATION OF SILVER MEAL HEURISTIC TO MRP LOT SIZING DECISIONS AT TÜRK TRAKTÖR FABRIKASI

A THESIS
SUBMITTED TO THE DEPARTMENT OF MANAGEMENT
AND
GRADUATE SCHOOL OF BUSINESS ADMINISTRATION
OF
BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF BUSINESS ADMINISTRATION

By BERKHAN N. ESMER

June,1995

I certify that I have read this thesis and in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Business Administration

Assoc.Prof. Erdal Erel

I certify that I have read this thesis and in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Business Administration

Assist.Prof. Murat MERCAN

I certify that I have read this thesis and in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Business Administration

Assist.Prof.Serpil SAYIN

Approved for the Graduate School of Business Administration

Prof. Şübidey Togan

ABSTRACT

AN APPLICATION OF SILVER-MEAL HEURISTIC TO MRP LOT SIZING DECISIONS AT TÜRK TRAKTÖR FABRIKASI

By

Berkhan N. Esmer

Supervisor: Assist. Prof. Erdal Erel

MRP Lot-sizing, capacity planning, Silver-Meal, Türk Traktör Fabrikasi

MRP does not consider setup and inventory holding costs for lot sizing but very quick algorithms like Silver-Meal can be used for lot sizing decisions. A module to apply Silver-Meal Heuristic to manufacturing work order releases at Türk Traktör Fabrikasý, Ankara, Turkey is designed and tested for 3 months. The module processes demand for finished goods, determines lot sizes, checks capacities by loading machines according to the schedule and reports the setup, inventory holding and overtime costs. These cost figures are compared with the MRP lot sizes. It is observed that, the planning time decreased by 90% by designing module in PCs connected to the main databases, and the total costs decreased by 50%.

ÖZET

SILVER-MEAL METODUNUN TÜRK TRAKTÖR FABRİKASI MALZEME İHTİYAÇ PLANLAMASI KAFİLE MİKTARI BELİRLEMESİNDE UYGULANMASI

Hazırlayan

Berkhan N. Esmer

Tez yöneticisi: Assocc. Prof. Erdal Erel

Malzeme İhtiyaç Planlaması, kafile miktarları, kapasite planlaması, Türk Traktör Fabrikası

Malzeme İhtiyaç Planlaması, kafile miktarlarını hesaplarken tezgah hazırlama ve stokta tutma maliyetlerini göz önünde bulundurmazken, Silver-Meal Metodu bu amaç için kullanılabilir. Silver-Meal Metodunu kullanan bir modül geliştirilerek, Ankara'da bulunan Türk Traktör Fabrikası imalat atölyelerine iş emri verilmesinde uygulanmıştır. Bu modül bitmiş ürünlere gelen talebi alt parçalarına dağıtır, kafile miktarlarını belirler, makinalar üzerine gelen kapasite ihtiyaçlarını hesaplar, ve tezgah hazırlama, stokta tutma ve fazla mesai maliyetlerini raporlar. 3 aylık bir denemeden sonra, modülün ana sisteme bağlı bir PC üstünde geliştirilmesinden dolayı planlama zamanında %90, ve toplam maliyetlerde de %50 iyileştirme yaptığı saptanmıştır.

ACKNOWLEDGEMENTS

I gratefully acknowledge patient supervision and helpful comments of Assist.Prof. Erdal EREL, throughout the preparation of this study. I would also like to thank to other thesis committee members Assoc.Prof. Murat MERCAN and Assoc.Prof.Serpil SAYIN for their valuable comments and suggestions.

I also thank to my family and teachers for the opportunities, they give me, and special thanks to Gülay BITIRICI for her kind support and patience.

TABLE OF CONTENTS

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

Materials Requirement Planning (MRP) in production planning and control was first used in 1950's as the development in computer technology made it possible to process information at reasonable costs. After the product tree was configured, a master production schedule and the records of inventory on hand were sufficient to form orders both for the make parts and buy parts.

When the stable demands of the 1950's left the ground to variable demands and customized products with variations, traditional MRP applications did not answer to production planning and control activities. The engineering change notices, and changes in master production schedule were proved to be difficult to be handled in MRP systems. Another drawback of MRP was the absence of the knowledge about the setup costs and inventory holding costs to determine optimal lot sizes. To overcome this drawback, quick procedures like Economic Order Quantity or Least Unit Cost, used to aggregate the orders in different time buckets and by Silver-Meal heuristic, this lot-sizing decisions become very easy to be solved.

With the improvements in technology, flexible manufacturing systems help the production people to cope with variable demand and customized products with variations. Just-in-time (JIT) philosophy, introduced by Japan manufacturers began to be analyzed by European and American manufacturers, and some other hybrid applications like Optimal

Production Technique (OPT) and Single kanban systems are created. These systems required low setup costs to produce many different products without making the lot sizes to increase over demand for not to increase inventory holding cost and satisfy the need of the customer immediately.

The requirements of JIT philosophy caused JIT not to be applied very widely in US and Europe. Zero lead time, zero inventory, zero set up times, zero defectives were not achieved by many manufacturers. For the products that they can accurately forecast, they shift to MRP-like planning systems, and left the ones with changing demands to be handled with JIT and some other production planning systems

For Turkey, situation is somewhat harder. The technological inefficiencies and macroeconomics unstability caused manufacturers to build systems on personal experiences, and production planning and control is made by persons rather than systems. For example, Türk Otomobil Fabrikası (TOFAŞ) still has not constructed its product tree, Aselsan bought Manufacturing Management (MANMAN) MRPII system in 1990, but used only Order Release Module and could not install Capacity Planning Module. Turkish Aerospace Industry (TAI) got the MRP system from General Dynamics, but they have the same problems of not loading lead times and not using capacity planning modules.

In Turkey, although demand level is unsatiable, there are not so many customized products, and for this reason the product trees, once configured, can be valid for years. This property of the market increased MRP usage in Turkey.

One of the firms who use MRP and created a sophisticated product tree is Türk Traktör Fabrikası in Ankara. In this study, lot sizing decisions of the dependent demand parts in this firm will be made by an heuristic, which will consider inventory holding costs and setup costs and results of the heuristic will be compared with MRP.

In this sequel, literature survey on lot sizing decisions are presented in Chapter II. Chapter III will give a general knowledge about the company. The data gathering for the study will be presented in Chapter IV. Data processing and assumptions will be stated in Chapter V. The module generated to handle the lot sizing decisions and capacity planning will be described in Chapter VI. The application of the module to the company's planning process for the first three months of 1995 will be presented in Chapter VII. The findings from the study and further research items will be stated in Chapter VIII.

CHAPTER II

LITERATURE SURVEY

Lot sizing decisions on MRP systems have been studied by many researchers considering setup and inventory holding costs. These researches have also identified capacity costs and tried to minimize total cost by linear programming models.

In their study, Olhager and Rapp(1985), used a linear programming model for simultaneous lot sizing and capacity planning. The model used total cost to count for number of setups required, and the average inventory level holding cost. The capacity problem is taken as a constraint and the interval of production is tried to be found by the interval which enables the capacity of resources enough for production. The model requires the product tree to be grouped as parents and off-springs. By the increases in number of parts, the model requires a very accurate determination of the groups. The computational difficulty increases resulting longer CPU time.

Tempelmeier and Helber (1994) also presented an LP formulation for the multi-item, multi-level dynamic lot sizing problem under capacity constraints. Model tries to optimize the set-up costs and inventory holding costs. The objective function is bounded by the capacity constraint. The capacity constraints are taken into consideration as proportion of demanded resource time to available capacity, and this ratio is bounded to be smaller than 1. As this ratio increases the inventory holding cost increases and setup costs decrease. The calculation requires the formulation of each product group to be spread over the families, and the number of constraints increase tremendously by complex product trees.

The computational problems due to formulation is tried to be resolved by the heuristic that is presented by the researchers. The heuristic takes the cost and demand as it reads the product tree. Then, the heuristic takes the first level in the product tree, and for each item in that level determines the set of resources and allocate some manufacturing time in these resources by the amount of demanded size. Then every resource is taken one by one and the allocated time for each item is summed. The lot sizing decisions are given by applying Dixon-Silver heuristic. Dixon-Silver heuristic tries to find the minimum cost per unit capacity by looking forward the planning horizon. This heuristic is quite simple, but tries to minimize capacity costs in general. The comparisons are done according to the capacity constraints and the inventory holding costs and setup times can deviate by 8-9% from the optimal solution.

Dillenberger (1994), proposed rather complex formulation to satisfy the requirements of the real manufacturers on part types, machine groups, storable and non-storable resources, setup families. Although these definitions were very helpful for the manufacturers, the computational difficulties were hard to overcome. The computation time required for 44 types of parts and 14 types of machines was 221.42 minutes and there were 3408 nodes to be checked on IBM RS/6000-540 and AIX XL C/6000 V1.1, known as the most sophisticated hardware and software. So, thinking that the computation time will increase by the increased number of parts and machines, LP formulation seems to be very inflexible and unsuitable for real manufacturers who have over 500 parts and over 250 machines, even with this very sophisticated hard and software.

Computational difficulties also raise the interest to find out one time formulations like EOQ to serve to balance inventory holding, and setup costs. Olhager and Rapp, on their trial to improve the EOQ formulation which they blamed of not considering setup times as capacity constraints, introduce capacity cost , queuing factor, processing times and furthermore they pronounce the opportunity cost of capacity utilization. The total cost takes the costs of capacity, holding and setup cost. Capacity cost includes both used and excess capacity. Inventory holding cost occur both by lot sizing decisions and the increased queuing time by the high capacity utilization of the machines. Here the formulation finds out first the extra capacity required and then the quantity that will be produced.

In literature survey, it was also interesting to find out that, the capacity and lot sizing decisions were transferred into spreadsheet designs, to increase the understanding of the people and the usage by managers. Eppen, Martin and Schrage (1994), reported that Ford Motor Co., have designed scenarios and spreadsheets on Lotus 1-2-3. By the effective usage of these spreadsheets by more people and managers, Ford Motor Co. post higher net income than General Motors, which had followed an aggressive capacity planning strategy and faced with too much capacity.

As a result of the literature survey, it can be noted that LP formulations have computational and design problems when tried to be used in real life manufacturing firms which have many parts to produce and assembly. And the improvements in Windows based information systems are replacing mainframes and networks are becoming the future of the computer science. The flexibility introduced by Windows and Oracle systems make it possible the use of spreadsheets in production planning activities by easily accessing the databases and using it as one wishes for his/her own need.

CHAPTER

III. TÜRK TRAKTÖR FABRİKASI

III.1. Company

Türk Traktör Fabrikası was founded in 1954 as Minneapolis-Moline Türk Traktör ve Ziraat Makineleri A.Ş. The problems faced in Turkish market caused Minneapolis to leave the factory to Italian Fiat, the State and Koç Group in 1959. In 1991, the State sold its shares to Koç Group and Fiat Agriculture is merged with New Holland Group. Recently, Koç Group has 75% and New Holland has 25% of share.

Türk Traktör Fabrikası is the oldest automotive plant in Turkey. In the first years, the factory was more like an assembly line. Most of the parts were imported from the joint venture Italian Fiat. By the industrialization of Turkey, TTF invested in different types of machines and began to manufacture most of the parts internally. The developing suppliers began to supply more parts for TTF. TTF still is a huge manufacturer with its 256 machines, 50 CNC's, and Heat Treatment Shop. There are about 2500 parts assembled, of which 500 is manufactured internally including engine blocks, transmission, gearboxes, axles. There are about 4000 operations registered in the routine sheets that are actively manufacturing.

III.2. Organization

TTF has a functional organization structure. The three business functions, production, financing and procurement, are under the responsibility of three general manager deputies. Production General Manager Deputy has engineering, production, quality control and maintenance managers reporting to him. Procurement General Manager Deputy has the raw material procurement, part procurement, price analysis and sales departments supervising. Financial General Manager Deputy is responsible from finance, accounting, and cost accounting managers. The Personal Department and Management Information Systems are directly reporting to the General Manager. (App .A1.)

III.3. Products And Production

Products of TTF are the tractors and crocieres(tractor bodies which are exported). Tractors are varied according to their horse powers, gear boxes and tracks. Horse power is varied from 55 HP to 80 HP, gear boxes from 8 front, 4 back to 12 front, 12 back and double or single track are available. The products and their features can be seen in Table 1.

Table 1 Products of TTF

TRACTOR	HORSE POWER	GEAR BOX	TRACK
54 JUNIOR	55	8	SINGLE
54 SPECIAL	55	8/12	SINGLE
55-56	55	8/12	SINGLE
60-56	60	8/12	SINGLE
70-56	70	8/12	SINGLE/DOUBLE
80-66	80	12	SINGLE/DOUBLE
CROCIERES	55-70	8/12	SINGLE/DOUBLE

TTF has a typical manufacturing environment of metal cutting processes. There are 256 different machines which include drilling machines, lathes, broaching, grinders, presses, welding machines, and heat treatment machines of nitrurizing, sementation, inductions. The average age of the machines is 10-15 years. There are both very old machines and very new ones like CNC's which are flexible enough to be a part of CAM.

There are about 520 parts which are manufactured in the factory, from all of 2500 parts assembled to the final product. The high labor cost, because of a strong union, force the firm to increase the proportion of buy parts. In turn, the factory managers want to manufacture high value added parts like engine block and head, where the company invested $13 million, and crank shafts where $6.8 million will be invested. And small parts which requires less skillful workers will be outsourced. By the increased number of high value added parts, the inventory holding costs, set up costs and capacity costs are becoming more important to the company.

III.4. Production Planning Activities

The MRP logic is used for the production planning. There is a database which holds the part usage quantities and the types that they are used in. There are 10 major types of tractor. There are 12 other tractor types which are derived from these major categories, which change in tire size, optional security frames, etc. which do not affect the manufacturing department, because of these optional parts are all purchased from the side industries.

The lead times for the manufacturing parts are taken as one month for every part. Two separate master production plans are prepared for manufacturing and assembly. The main reason for two separate master plan is the capacity problem in the machining shop. The demand structure which is low between November and May, increases during May to October, reaching its peak in October. So, manufacturing parts are inventoried during low demand months and used in peak demand months. Capacity is tried to be stabilized in this manner. Also, production of spare parts are also scheduled in the first three months of the year. Capacity planning is made roughly considering only the one tractor's capacity requirement in the machine shop multiplied by total production. The capacities of the resources used in the machining department is not considered while preparing the master manufacturing plan.

Every month new production orders are given to the machine shop, just the amount that are needed for that month's manufacturing plan. No consideration of setup costs or inventory holding cost is made. No special properties of the parts are taken into consideration. The capacity requirements on the resources is also left to the machining shop, and the problems seen are tried to be handled by overtime and more importantly by high stock levels. The inventory turnover for the manufacturing parts is 5.8, which means there exists about two months' need of parts as WIP or ready-to-assembly.

The database used in TTF is quite rich and helping for the study. The accounting system is not the standard but actual accounting system. This enables the cost of inventory holding cost, and capacity cost known at each stage of production. The routes of the parts, the standard machines for the operations, part usage, labor cost for each operation, standard times, prices of the raw materials, overheads, book values of machines are all part of the database and will be used in lot sizing decisions.

There are many schedule busters going on the manufacturing stages. Most common one is the supply delays in raw materials. The delay seen in supply of raw materials makes the planning activities not to work efficiently. Because of the high proportion of old machines, breakdowns are seen frequently. Capacity problems are also tried to be solved by altering lot sizes, and lots are divided into smaller lots by machine shop.

Production Planning is made by a central planning department which is responsible from all orders and inventories for any part that is used in the finished goods or indirectly used by the workers or in the production processes.

The procedure used in planning is simply MRP logic. A parts list has the parts and quantities that are used in the products. The requirements for every part is calculated, then the available stock is subtracted, and the remaining quantity is ordered to the suppliers for buy parts or material requirement forms are released to the Raw Material Warehouse, for the machining shop to manufacture the make parts.

The lead times are accepted to be one month for every make part, and these materials releases are submitted to shop floor at the end of every month for the production of the following month. The scheduling is handled by the Machine Shop Department. The operations stocks are kept in the department by entering the Work Tickets which are filled by the direct workers who are reporting the time that they work and the quantity of the produced parts. By the help of these operation stocks and the finished parts, shop floor schedules the production in the machines by the most critical part that the stock level decreased to low levels. The main aim of the shop floor is to make only one setup for every part for the whole load of one month's requirement. But, this principle sometimes broken, because of urgent parts that require setup changes. Sometimes the lots of one month's requirement is divided into smaller lots because of this setup changes.

III.5. Review Of The Current System

Current production planning activities result in high inventories, low throughput, and imbalance of material flow. The raw materials are transported to the machine shop on the last week of the month and resources necessary for this activity are kept very busy on this week and idle on the other weeks. Again the initial operations for casting and forging materials are heat treatment and some very common cutting and drilling operations. In the first week of the months overcapacity is faced on these machines. Because of the one month's material flow to the shop floor at the same time, the average inventory is about two weeks demand. If we add one month phase difference between assembly and machining shops, this adds up to one month and a half months of inventory. When WIP is high, it becomes very hard to schedule jobs for the machines. This mess sometimes caused manufacturing time to go beyond two months, just because of waiting in the queues for urgent parts to be processed.

III.6 Performance Measures

Since the goal is to minimize costs, the related costs with the lot sizing decisions must be defined. The costs that are results of lot sizing decisions are ***inventory holding cost, setup cost and overtime cost.*** As lot sizes are decreased, inventory holding cost is decreased since average inventory levels decrease. Also, the queuing time can decrease leading to higher throughputs. But, decreased lot sizes make it necessary to make more setups which can cause capacity problems and increase setup cost. So, there is a trade off between inventory holding cost and set up cost. At the same time, it is necessary to analyze overtime costs because resources have finite capacities. A schedule must not load machines imbalanced that one week it will be a bottleneck for the system and the other week it will be idle. The overloads can be captured by the overtime costs required by the resources to produce the work orders. The performance of the schedule, related to capacity, can be measured by total overtime cost.

For this study, inventory holding costs, setup costs and overtime costs will be considered as the measures which can evaluate the performance of a manufacturing schedule by the lot sizing decisions.

CHAPTER IV.

DATA GATHERING

IV.1. Software Selected

The software selected for this application is Microsoft Excel Ver. 5 and Microsoft Access Ver 2. The reason for making these applications independent from the mainframe is just to have the advantages of the Era III[1] in Computer Technology. The networks enable the users to access the company database by their personal computers and process the data to increase the effectiveness and flexibility of the users. To get benefit of the effectiveness and flexibility, MRP lot sizing problem is designed at PC by using PC software, but manipulating the data from IBM A6 mainframe of Türk Traktör Fabrikası (TTF).

IV.2. Database

The firm has a big database that holds the parts list, the operations, the processing times, setup times, machine overhead rates, the amount of material necessary, the stocks, the stock movements, etc. The cost accounting system of the firm is actual costing, so that the overhead values and direct labor costs are traced by the machines, and available data for the holding costs can be obtained. The necessary data for a lot sizing application were available in the factory. From these data, necessary tables to be used in Microsoft Access are created.

[1] CASH, McFARLAN, McKENNEY,APLLEGATE, Corporate Information Systems, Texts and Cases.
Era III is specified as the regulated free market in IT. The users are free to build their applications, but have to obey the regulations set by the IT- staff.

IV.3. Bill Of Material (BOM)

The first important file for the MRP is the BOM. For the sake of the computational simplicity, the tree structure is turned out into a matrix form, in which the dimensions are the products produced and the parts used. The entries are the quantities that are used from that part in the given product. For the sake of matrix multiplication, the table is created in MS Excel Ver.5. as a spreadsheet. A portion of the sheet can be seen in Fig. 1a and the related table in Fig. 1b.

This spreadsheet is used to hold the part numbers, the required material for manufacturing, and the quantities used in tractor models. This table is necessary to obtain the gross requirements of parts for the given planning horizon. The master production quantities for every week in the planning horizon of 3 months will be entered in Microsoft Excel. By the matrix multiplication of master production and PartsList, the requirement for every week will be calculated in Microsoft Excel. Then the Silver-Meal algorithm, will formulate the necessary lot sizes to be released to the shop floor every week, looking at the setup and inventory holding costs.

IV.4. Operations, Direct Labor and Overhead

The routings for the parts that are manufactured in TTF are stored in TTF-A6 system. These operations are used for the flow of material, the productivity measures, WIP counts, and cost accounting system.

The cost accounting system holds the labor costs and overheads for each operation and machine. When the table "operasyon" was created, the part number, operation number, machine id, process time, direct labor cost, overhead rates are transformed into MS Access Ver.2. The table definition and fields can be seen in Fig.2. The fields in "Operasyon", enables to measure the base for inventory holding cost. The total value added in Machine Shop can be calculated from the operation table by the help of processing time, direct labor cost and overhead rate. This table is mainly used for obtaining the base costs of parts to calculate the inventory holding costs, and to load capacities to the machines.

IV.5. Setup Times

Setup costs have never been calculated in TTF. Setup times are considered in the scheduling process and it is thought that, setup is a cost element and when you increase the number of setups, you will have more costs and low efficiency. Set up times were stored in Production Engineering files. The table Setup is created by transforming the partno, operation no, machine id, and setup time necessary. Table Setup can be seen in Fig.3.

CHAPTER V

DATA PROCESSING

Performance measures selected to evaluate lot sizing decisions and schedules will be generated by the processing of the data gathered from TTF A-6 database in MS Excel and MS Access.

V.1. Setup Costs

To find out setup costs of parts, the table "operasyon", which hold partno, operation no, machine id, direct labor cost, and overhead rate and the table setup is processed. The Query "setupcostsofop" is created. This query looks at the part routing from table "operasyon" and for each operation of the part finds out the setup time necessary for the operation. For every operation on the part, the required setup time is multiplied by the direct labor cost plus overhead rate. The query can be seen in Fig. 4., with its properties, SQL, and fields.

The overhead is included in setup costs, because the only overhead cost that are not actually used during setup is electricity and coolants, which are a small portion in the overheads. The other overhead items are forgone during setup, like engineering costs, administration costs, etc.

After the setup time is quantified as a cost measure on every operation for a part, the query "setupcostofparts" are created to sum up all the setup cost of operations of a part. The query can be seen in Fig.5.

Setup cost is considered to be a cost item in traditional cost accounting. But increased necessity of flexibility in production looks at setup as a manufacturing time and only the inefficiencies in setup is considered as a cost. Modern accounting systems are supposed to look at the time to manufacture the product from the order to the customer hands. And everything that will reduce this time will be considered as an asset. So in fact, frequent setups must be seen as an asset, since it reduces the queuing time, decreasing WIP and increasing flexibility. But the capacity can be a constraint for frequent setups, if setup times are not less than an hour. But for the machines that are not bottlenecks, or that have no capacity constraints, the setups are just a better usage of resources which will be idle because of low demand or which will be kept busy to produce to stock.

V.2. Inventory Holding Cost

Inventories are unfortunately seen as an asset in balance sheets. But in fact, they are huge amounts of money tied up in warehouses to bring 0% return. However, in Turkey, the government bonds which have default risks, offer 10% of interest for a month.

For lot sizing module, inventory holding cost is considered to be 10% of the inventories on hand for the month. The main reason for 10%, is the government bond rates. Although it is obvious that higher interest rates are possible for a month, to measure the risks of tractor manufacturing and compare with other alternatives that have the same risk level is beyond this study.

The holding cost for the parts consider only labor cost and overhead cost as the part base price. The raw material price is not considered as an inventory holding cost base. The only reason is that, the supplier contracts are designed to be shipments made between the 1st and 5th days of the months and shipments more than once in a month is not programmed in current A6 system. The shortest planning horizon is one month in all TTF-A6 database. Since, there is no chance of regulating raw material shipments, the cost of raw material is not included in the base price of inventory holding cost.

The query "holding cost" can be seen in Fig.6. For every operation the total process time is multiplied by the direct labor cost plus overhead rate. All the operation costs of the parts are then summed . The total cost can be seen as the total value added to the products in the machine shop. This total value add is the base for inventory holding cost. The WIPs and finished products are multiplied by the base price and total inventory value is found. This total inventory level is multiplied by 10% for the monthly inventories and 2.5% for the weekly inventories.

V.3. Overtime Costs

The capacity constraints are made roughly in most organizations. That is, the capacity requirement for each finished product is known and this amount is multiplied by the production quantities. This is very rough for many departments which have multiple resources to do the job. The problems of resources, temporary or permanent, can cause delays in previous jobs, and most part of the available capacity can be allocated for these late one or two parts. Or the most frequently faced problem, the late deliveries of raw materials from suppliers can cause a major imbalance in Machine Shop and certain resources can be very loaded for certain periods.

But none of these problems can be detected by the rough capacity planning. These imbalances in production schedules in terms of capacity must be quantified to show the planner the result of the schedules and lot sizes on machine capacities.

In TTF, the workers are hourly paid and TTF is responsible to offer 9 hours of work for five days of the week. So, whether the worker is idle during 9 hours of work or not is not considered in wage pays. If the worker is in the factory, s/he will be paid. As long as the production schedules can cause many setups, but no overtime, there is no additional cost of the lot sizing. But if the lot sizing cause overtime because of frequent setups, or very big lots, this may be an additional cost which will not be incurred if monthly releases were to be continued.

The idle time of resources caused by the new lot sizing procedure is not considered as a cost, because every cost is a sunk cost. The price of the machine is paid, the salary of the worker will be paid, whether s/he will work or not. Overhead may be incurred during idle time of the machine, but this will be double count of the overhead, because overhead cost is included in trade off between inventory holding costs and setup costs, by including overhead in the base price for holding cost.

Overtime cost will be found as the necessary overtime for the period, multiplied by the direct labor cost plus overhead rate. The query "overtimecost" can be seen in Fig 7.a and 7.b. with its properties, SQL and fields.

To calculate necessary overtime for the period, another query "machineload" is designed. This query gives out the total time required for machines to process the parts which have been scheduled to be manufactured for the period. The query can be seen in Fig.8. This query looks at the production schedule and lot size. For every lot to be processed, it goes through its routing and multiply the process time of operations by the lot size and stores the result as the capacity load for the machine.

The setup times necessary for the machines are added to the process times by the query "machineloadandsetup". This query looks at the production schedule over the planning horizon and adds setup times for parts that has been scheduled to be manufactured. This query can be seen in Fig. 9. These costs are aggregated to weekly costs by the query "machineloadbyweek" in Fig. 10.

By the aggregation of necessary processing and setup times for machines, the required overtime can be calculated by the difference between capacity loads and current capacity of the machines.

Although the capacities of machines for the planning horizon must be stored as a table to decide about overtime, there were no available data on this issue. Machine shop was working two shifts, and they were trying to use all the machines for two shifts. So, the capacity of every machine is taken as 5*16*60 = 4800 minutes per week. If the breakdown rate data can be obtained, these rates should also be reflected to the capacities available.

V.4. Scheduling Problem

TTF has no available data on lead times. The one month inventory of the manufactured parts is sufficient for every part and used as a lead time. But as the CNCs are increasing in machine shops, one week of lead time becomes a reality.

The lead time for any part will be one week. So, the lead time is taken as a week for every part, and forward scheduling is made to anticipate the loads on the machines. It is assumed that if the work order of a part is released to the shop floor, it will be manufactured within that week. This fact simplified the scheduling problem, and no consideration of backward or forward scheduling is made.

CHAPTER VI

THE MRP LOT - SIZING MODULE

VI.1. Purpose

The purpose of the application module is to present the users a flexible, helping and friendly alternative to lot sizing decisions of work orders by the use of Silver-Meal Heuristic. The application module must be easy to understand and utilize, and must be accurate and reliable. Also, the module must show the situation from different perspectives like setups, inventories and capacities.

VI.2. General Framework

The general framework of the MRP Lot Sizing Module can be seen in App A2. Application module starts with the master plan entrance. These amounts are multiplied by the usage quantities of the parts and gross requirements are obtained. These requirements are processed by Silver-Meal Heuristic and the lot sizes and schedules are processed by several queries to obtain the machine loads, setup costs, overtime costs. The decision maker can go through these queries and fine tune the schedules.

First option is the production plan entry. Master Production Plan can be entered to the system by this option. The other option under Data Entry is the transfer of data. The data on MS Excel Ver.5. about the production schedules is transferred to MS Access Ver.2. when the option is selected. The other option group is Details of Part Schedules and Machine Capacities. Capacity load of a machine, or the route of a part that will go through can be seen with these options. The option group of Cost By Parts and Machines, gives out the setup costs and inventory holding costs of selected parts and overtime costs of selected machine. Costs by Weeks option determine the total setup, inventory and overtime costs by periods in the planning horizon.

VI.3. Master Plan Entrance

Master Plan Schedules are prepared by the Production Planning Department. Production Planning Department determines the assembly schedules of the tractors and prepares a Manufacturing Master Plan accordingly. The amounts of tractors that must be manufactured by the machine shop is converted into work orders on parts and delivered to Machine Shop.

A sample production plan can be seen in Fig.11. The products to be manufactured, the production amounts and the production weeks are determined in the schedule. Master Plan Entrance is designed in MS Excel. The main reason is to utilize the MS Excel matrix multiplication functions and MS Excel Visual Basic Programming Language for the data to be processed by Silver-Meal Heuristic. By the entrance of Master Plan, MS Excel multiplies the production amounts, by the part usage quantities and drives out total requirements for each part. The result of these matrix multiplication is put in a spreadsheet to be processed by Silver Meal Heuristic.

A sample portion of these sheet can be seen in Fig.12. In this portion, setup costs and holding cost for the parts which were calculated in queries of MS Access is copied. Silver-Meal Algorithm will use this data after, when lot sizing decisions are to be made.

VI.4. Silver And Meal Algorithm

Lot-Sizing heuristics were tried to be made by Linear Programming models in order to have an optimum solution. But the necessity of a high computational power to process complex production facilities, some heuristics were created, like Economic Order Quantity.

Silver and Meal in 1973 developed the heuristic which try to minimize the total relevant costs per unit time for the duration of the planning horizon. The heuristic was intended to cover the drawbacks of Economic Order Quantity which requires very stable demand patterns and Wagner - Whitin approach which have complex nature of algorithm, and require a well-defined ending point for the demand pattern, and the necessary assumption that replenishments can be made only at discrete intervals.

If a replenishment arrives at the beginning of the first period and it covers requirements through the end of the Tth period, then the criterion function can be written as :

$$\frac{\text{(Setup Cost)} + \text{(Total Carrying Costs to the End of Period T)}}{T}$$

This criterian function will be used in deciding lot sizes.

VI.4. 1 The Essence Of Algorithm

The replenishments at the beginning of the periods force the planners to make replenishment quantities that last for an integer number of periods. Consequently, it can be thought that, decision variable for a particular replenishment will be T periods that the replenishment will last, with T constrained to integer values. The replenishment quantity Q, associated with a particular value of T is :

$$Q = \sum_{j=1}^{T} \text{Demand}(j)$$

According to the chosen criterion, decision variable is the number of periods T, which will minimize the total relevant costs per unit time of replenishment and carrying costs over the time period T.

If total relevant costs associated with a replenishment that lasts for T periods, denoted by TRC(T), then we can use TRCUT(T) to denote the cost per unit time, where :

$$\text{TRCUT} = \text{TRC(T)} / \text{T} = (\text{ Setup Cost} + \text{Inventory Holding Cost }) / \text{T}$$

The basic idea of the heuristic is to evaluate TRCUT(T) for increasing values of T until, for the first time ,

$$\text{TRCUT(T+1)} > \text{TRCUT(T)}$$

that is the total relevant costs per unit time start increasing. When this happens, the associated T is selected as the number of periods that the replenishment should cover. The replenishment quantity Q is the total need of the parts to satisfy the time period of T.

VI.4. 2 Coding and the Results

Silver Meal heuristic is coded on Microsoft Excel Ver.5 by Visual Basic. The spreadsheet holds the requirements for each week for each part and the setup and holding costs.

The procedure first finds out TRCUT(T)s for every period and writes on the spreadsheet. When the TRCUT(T+1) > (TRCUT(T), the total amount of requirements necessary for T periods is taken as a release for machine shop and again written to a separate region of the spreadsheet.

The code of the algorithm can be seen in App. B3. The algorithm starts with initializing the old values. Then it gets critical data of the planning horizon, number of items that will be planned, and the interest rate that will be valid for the planning horizon. Although it is possible to include forecasted interest rates that will be valid for each period, in Turkey it is very hard to see the coming periods valid interest rates. So, one interest rate is made relevant to all periods in the planning horizon.

Then the loop that will take the parts one by one is started. The loop runs for the number of parts that is defined in the constant parameter of t_parts(total number of parts). When one part is started to be executed by the algorithm, released quantity and last released period values are initialized to 0 and 1 respectively. That means that, there have been no released part lot to be manufactured, and the last period that the lot release was given is the period 1. The current period number is hold on the parameter *j*.

These two variables are the backbones of the algorithm. First parameter, ***released*** actually holds the lot size that will be released to the manufacturing departments. This is the decision variable Q which is the sum of demands of the periods which are to be satisfied with a single production lot. The other parameter ***lastrel*** holds the last period that the demand was satisfied. So, when the lot size decision made for the succeeding production, the period that the lot will be released is the lastrel period, because this periods demand is not satisfied yet.

After these initializations, algorithm starts and began to calculate period costs. The period cost TRC(T), is simply the setup cost plus inventory holding cost. Setup cost is taken from the second column of the requirements sheet which can be seen in Fig.12. The inventory holding cost is somehow harder. The inventories to be hold differ in number of periods that is stocked at shop floor or warehouses. When you give a lot that will be enough to satisfy demands of 100,150,200 parts per period, first period demand is not stocked. It is pulled in the first week. But 150 of them stocked for one week, and 200 of them stocked for two weeks. So, holding cost must take care of total inventory holding time also.

This problem is solved by the introduction of ***prevcost*** parameter which means, previous period cost, which will also be necessary for the comparison of TRCUT(T-1) and TRCUT(T). So, prevcost is TRCUT(T-1). To calculate the inventory holding cost of the current period, the previous period cost is taken and changed into TRC(T-1) by multiplying the TRCUT(T-1) by (T-1). To these total cost ,only the inventory holding cost of current period demand for the periods between last released period and current period is added and TRCUT(T) is find.

The calculated period costs (TRCUT(T)) are written on the spreadsheet and can be seen in Fig.13. When period cost is calculated, it is compared with previous cost. If the period cost is lower than the previous period cost, it means that the holding cost of the parts is less than the necessary setup cost to satisfy this periods demand separately, so this period demand must be satisfied by the production in last released period. By this decision the ***released*** parameter is updated to include this period demand.

If the previous cost is less than current period cost, it means that the additional inventory holding cost to satisfy this period demand with the production of the previous demands is more costly than an additional setup. So, another setup is advised in order not to incur holding costs of higher then the setup costs.

By the decision of T, the total demand which is hold in the parameter ***released*** is written to the spreadsheet range which holds the production schedule. The released amount is written to the range of spreadsheet under the column which shows the last period whose demand is not satisfied by the preceding production lots (***lastrel***) . This range can be seen in Fig. 14. The part number, the lot sizes and the production periods are tabulated as a production schedule.

By the determination of the lot size by finding a period where previous cost is smaller than the current period cost, the parameter released is initiated to 0, to show the algorithm that the current period demand is not satisfied, and lastrel is updated to include the previous period number, to show that the last satisfied demand was the previous period.

VI.5. Transfer Of Data

The transfers between MS Excel Ver.5. and MS Access Ver.2. are made by two macros. The first one is the macro "Masteral" which can be seen in Fig. 15. The macro runs MS Excel Ver.5. and open the file called "Master" which holds the BOM structure and Silver-Meal Algorithm. And the data from this file is transferred back to MS Access Ver.2. by the macro "Transfer". "Transfer" runs the delete queries to delete previous production schedules data from the database and imports the new relevant ones.

The production schedule is imported to the table under the name "IHT". This table can be seen in Fig 17. with its properties, and fields. The ending inventories are imported to table "INV" which can be seen in Fig. 18.

VI. 6. The Capacity Loading

As the production schedule is determined, the capacity loading must be checked. The resources necessary for the production of parts are loaded according to the production schedule, process time and set up times. The loadings are done by the queries mentioned in section V.3. The option group "Details" in the Main Menu help the user to go through part and machine details.

The part details button, activates the "Part Details Menu" and the user have the authority to see production schedules and the route that the part will go through, and the necessary capacity to be allocated to that part from the resource machine.

The Part Schedule Details option, activates the Query "PartSchedule". This query asks the user the part number, and the production schedule is selected from table "IHT". The user have the authority to change the schedule in this screen.

This option is very important for users, who have faced the very strict lot sizing modules of central databases. Many databases require authorization formalities and programming difficulties, when changing the production schedule. Some of them have some restrictions on changing the schedule. And many of them can not immediately show the result of changed schedule, you have to run all the module to see the effects of the changes.

The query "PartsSchedule" can be seen in Fig. 19. If for any reason the schedule is desired to be fixed, and not be changed, MS Access Ver.2. offers a password protection for the data on queries.

To look at the part manufacturing route of the part, the Part Loads on Machines Option is available. The query "PartCapDetails" is designed to show the user, the resources that the part will go through, the necessary time to be allocated for the part by the resource. The query take the values from the another query "MachineLoadAndSetup" which is mentioned in section V.3. The query "PartCapDetails" can be seen in fig.20.

The machine capacity details is the other option presented to the user. The user can see the capacity loads on the machines and parts that are loaded. The query designed for this purpose is "CapacityCostAnalysis". When the machine no is entered, the parts that uses that machine, the parts that use that machine, operation no, the process time and the setup time, the overtime necessary for that machine and the overtime cost is seen. The query can be seen in

Fig.21 with its properties, SQL, and the fields. The user can see the effects of the schedule to the capacities of the resources.

As the last option the user can see the period costs under the current production schedule. This option group gives the total inventory holding costs, setup costs, and overtime costs. These costs are just the aggregation of Detailed Costs by Parts and Machines group options. The aggregation queries OverallSetupCosts, OverallHoldingCost, and OverallOverTimeCost can be seen in Figures 22,23 and 24 respectively.

CHAPTER VII

THE APPLICATION OF THE MRP LOT - SIZING MODULE

The module generated is used to prepare the production schedule of TTF for the months of April, May and June 1995 and the theoretical results, with the actual WIP inventory value and capacity usage and overtime is compared with the old MRP system.

VII. 1. The Master Production Plan

The required amounts of finished goods can be seen in Fig.11. For the Lot Sizing Module these values are entered, but for the test of the old system, 4 weeks of demand is summed and entered as the monthly demand. The first application is made by the complete balance of demand to four weeks, that is the monthly tractor orders are divided into 4 to find out weekly demands. But, by the acceptation of the lot sizing module, the firm will change the orders to weekly, and weekly tractor assemblies will be fixed and shipments to dealers will be made according to these fixed schedules.

The parameters reflect the high demand for tractors, which is very high for the company. Unfortunately, the company was facing a very high demand that the production goes through 12,000 tractors per year. The firms average production was 9,000 tractors/year, and this increased production cause the overtime costs to increase tremendously. The firm prefers overtime to hire new workers, because the demand is not supposed to be permanent, and the belief is that next two and three years sales are made from today by the new subsidy policy of government which increase the farmers income by 400% relative to the last years income values.

VII. 2. The Part Requirements And Lot Sizes

After the master production plan entry, Silver-Meal Algorithm is executed. The period costs can be seen in Fig. 13. The costs are decreasing for the parts which have relatively higher setup costs then inventory holding costs. Or the costs are not decreasing for the parts which have very negligible setup times.

The part 121573 5118742 have 0 setup cost, because it does not require any tools or fixtures to be processed. It is operated in a special purpose machine and does not require any additional setup. So, the Heuristic gives production for every week in order not to incur holding cost.

But on the contrary, the part 119144 599155 have a TL1, 310,677 setup cost and only TL11,553 inventory holding cost base price. For not to incur a setup cost of TL1,310,677, the algorithm gives the all periods' demand at the first week, and when the part is fixed to the machines, the whole demand of 3 months will be processed.

To compare the two production schedules, Fig.25a and Fig.25b can be examined. The former method gives the monthly requirements in one lot as can be seen in Fig.25a. But in Fig.25b, the lot sizes differ and scheduled throughout the planning horizon.

The part 126224 596244 has a requirement of 42 and 50 units for the first and the second month. But the new lot sizing module aggregates all the demand for three months in one lot. Because the setup cost is TL. 1,027,000 whereas the holding cost is only TL. 85,590. But the part 126502 599933 has the three week's demand as lot sizes because the holding cost is TL. 274,238 per part but setup cost is TL. 1,549,606 .

VII.3 Setup Costs

After scheduling is done and transferred to MS Access, the setup costs are calculated. Set-up costs of the parts are tabulated by a visual report on the screen (Fig.26a and 26b). In those tables, the production schedule is changed to include setup cost of part instead of lot sizes. If there is a lot scheduled for the period, set up cost must be incurred in order to manufacture the part. The total setup costs of parts in the planning horizon can be seen in the last column.

For the part 126224 596244, the total setup cost of three months is TL. 2,054,000 for the former method but TL. 1,027,000 for the lot sizing method. And for the part 126502 599933 it is TL 4,648,818 for the former method and TL. 6,198,424 for the lot sizing method.

Setup costs for the former system is tabulated in Table 2. Since, the former system is designed to make only one setup for each part, setup costs are expected to be stable over months and easy to anticipate. Setup costs for the new system is tabulated in Table 3, and a comparison of the setup costs for both system is made in Exh 1.

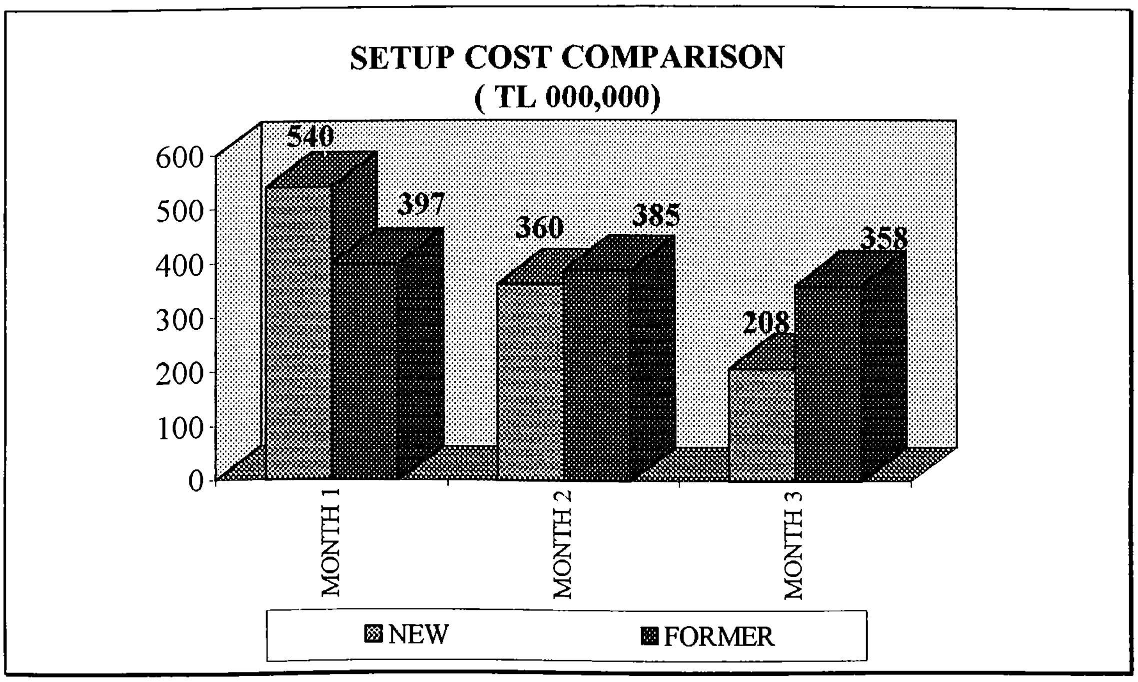
Table 2 Setup Costs for former system (000 TL)

1. Month	2.Month	3.Month
397,343	385,531	358,785

Table 3 Setup Costs for the new system (000 TL)

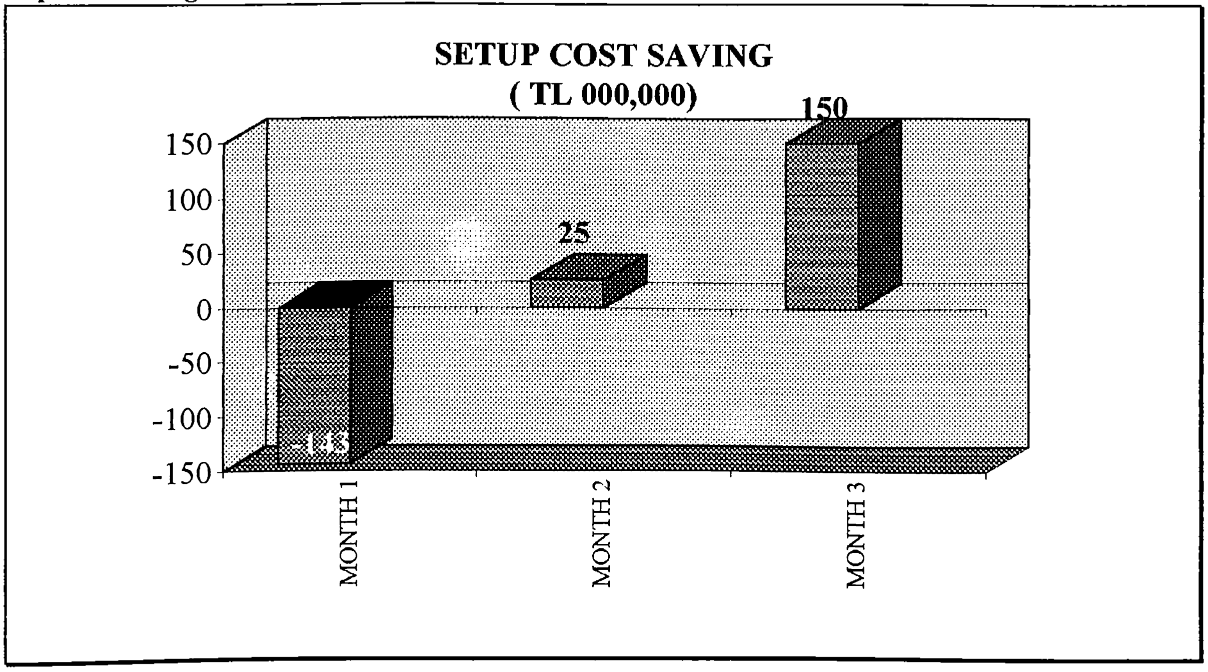
	MONTH 1				MONTH 2				MONTH 3			
WEEKS	1	2	3	4	5	6	7	8	9	10	11	12
COST	401,484	5,219	51,395	82,537	118,995	38,661	141,374	61,964	114	85,488	71,232	52,027
TOTAL	540,635				360,995				208,860			

Exhibit 1. Setup Cost Comparison



As seen from the figures, the savings from setup costs are increasing as the new system is used for the next coming periods. As it is stated, the new system has to give work orders for every part in the first week and for some parts, the orders are for demands of several periods. The savings from the system can be seen in Exh.2.

Exh.2. Setup Cost Saving



VII.4 Holding Costs

The holding costs are calculated by the help of another table, "Inv". Inventory table has the lot sizes minus the demands of that period. So, the fields give the ending inventories. These ending inventories are multiplied by the appropriate interest rate and holding costs are generated.

The holding costs for the former and new lot sizing module can be seen in Fig.27a and 27b. The part 126224 596244 has TL 98,429 holding cost in the former system and TL 740,355 in the new one. The part 126502 599933 has a total of TL 3,222,300 in the former and TL 6,547,439 in the new one.

Table 4. Trade Off Between Setup and Inventory Holding Costs

Part No: 126224 596244

	Holding Cost	*Setup Cost*	*Total*
	85,550	1,027,070	
New	740,355	1,027,000	1,767,355
Former	98,429	2,054,000	2,152,429

Part No: 126502 599933

	Holding Cost	*Setup Cost*	*Total*
	274,238	1,549,606	
New	3,222,300	6,198,424	9,420,724
Former	6,444,593	4,648,818	11,093,411

As can be seen in Table 4, the new lot sizing module trade off between set up and holding costs decrease the total cost of operations. Inventory Holding Costs for the former system and the new one can be seen in Table 5 and 6, respectively.

Table 5. Inventory Holding Costs for former system (000 TL)

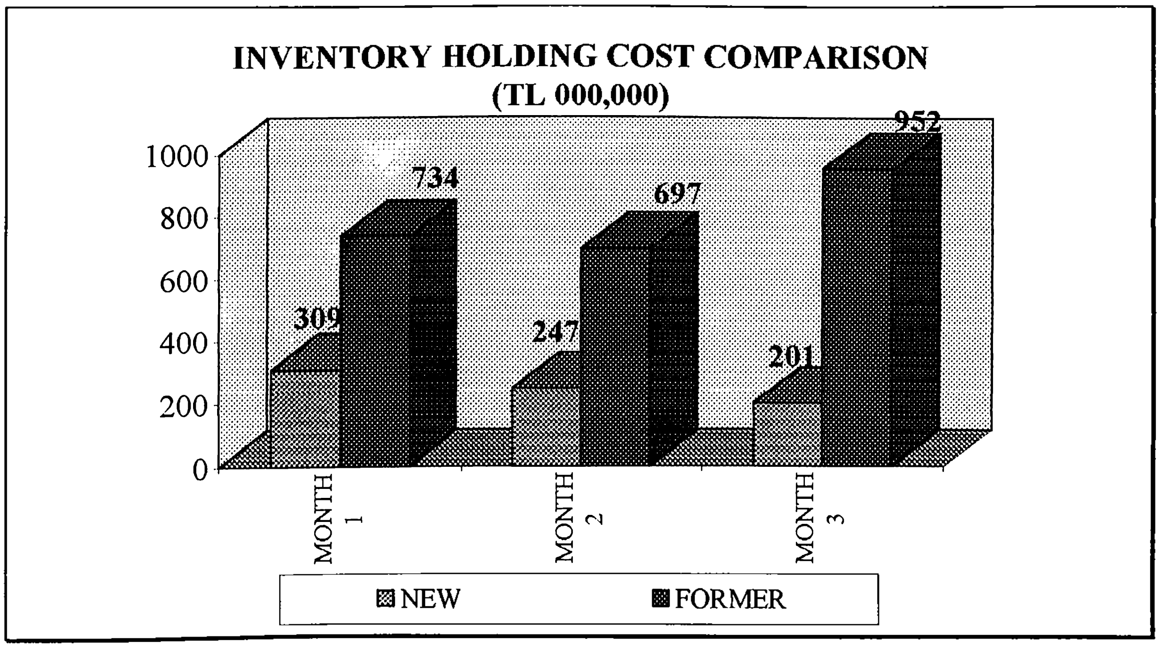
1. Month	2.Month	3.Month
734,026	697,751	652,878

Table 6. Inventory Holding Costs for the new system (000 TL)

	MONTH 1				MONTH 2				MONTH 3			
WEEKS	1	2	3	4	5	6	7	8	9	10	11	12
COST	#####	67,924	62,034	53,360	81,042	40,878	80,087	41,815	62,248	52,730	46,749	40,185
TOTAL	309,703				243,822				201,912			

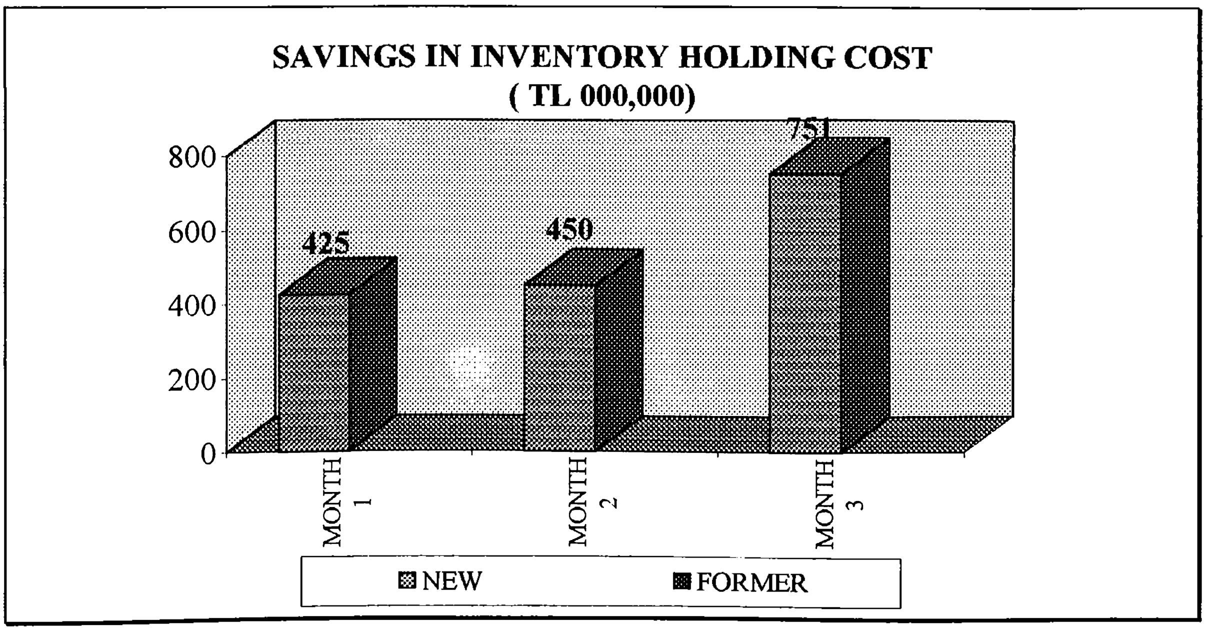
The new system decreases total inventory holding costs by more than 50%. In Exh. 3 the comparison of the holding costs can be seen. The parts which have manufactured to satisfy the demand of more than 1 month causes inventory holding costs to decrease steadily, and more setups for costly parts enable lot sizes and average inventory levels to decrease.

Exh 3. Inventory Holding Costs Comparison



By the new lot sizing method, a significant saving is made in inventory holding costs. Increase in savings for holding cost is mainly decreased work orders released after the first week and production quantities of finished goods. The savings from the new lot sizing method can be seen in Exh. 4.

Exh. 4. Inventory Holding Cost Saving



VII.5 Overtime Costs

The loadings on machines are done according to the schedules and the results are reported both in terms of loaded minutes and the overtime costs. These costs give feedback to the planner about the performance of the plan according to the capacity constraints.

Firstly, the loads on machines can be investigated by the planner. The reports in Fig. 28a and 28b show the total load on the machines. The report is helpful to find out the bottlenecks. The bottleneck resources can be further analyzed by the "Machine Capacity Details" option on the menu. The parts that go through the bottlenecks can be seen by this option. If necessary, the parts schedule that go through the bottlenecks may be viewed and changed by the " Part Schedules and Lot Sizes " option under the "Part Details " option group. The altered part can also be viewed to see what other machines will be affected by the change of the schedule by using the "Part Details " option group.

In order to decide decreasing lot sizes and adding more setups in front of the bottleneck machines, the planner can trade of between the overtime costs and the holding and setup costs. To help planners, the overtime costs are reported which can be seen in Fig. 29a and 29b.

The overtime costs are the consequences of scheduling. The total finished goods to be manufactured is the major factor for the capacity requirements. But lot sizing decisions can affect the total capacity required, because these decisions determine the unproductive setup times which absorbs the machining time and the waiting time of a part in the machine queue which increase the total manufacturing time. Overtime costs can be seen in Table 7 and 8 for the former and new systems, respectively.

Table 7. Overtime Costs for former system (000 TL)

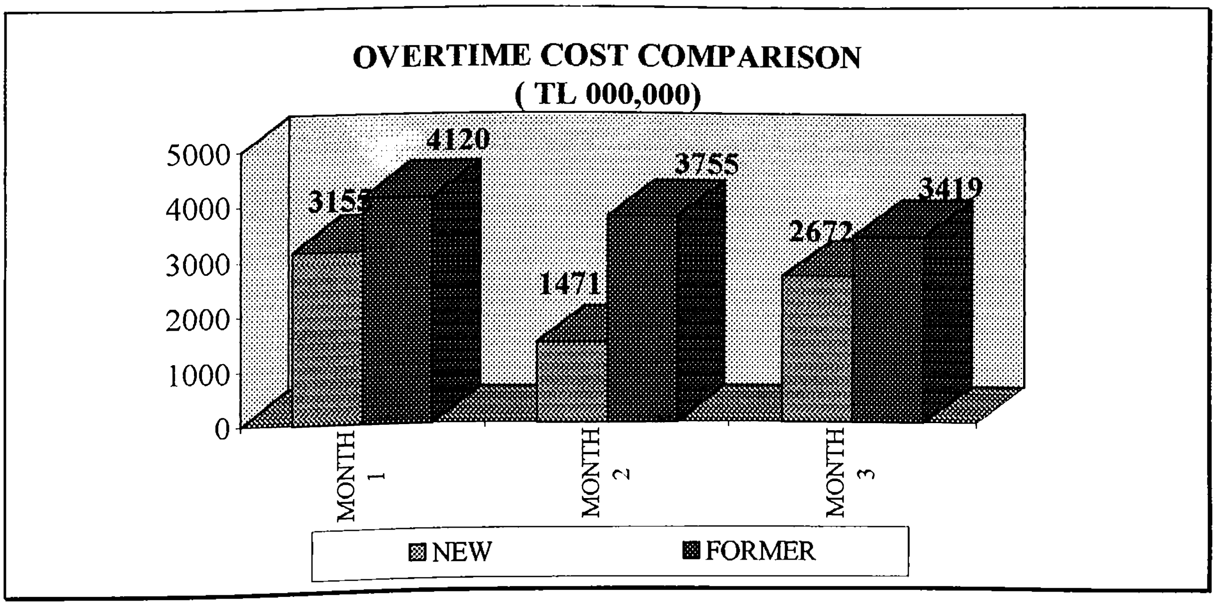
1. Month	2.Month	3.Month
2,989,111	2,672,464	2,408,126

Table 8. Overtime Costs for the new system (000 TL)

	MONTH 1				MONTH 2				MONTH 3			
WEEKS	1	2	3	4	5	6	7	8	9	10	11	12
COST	2,517,131	23,385	447,236	167,654	703,600	14,729	729,614	23,439	534,422	78,330	305,155	6,154
TOTAL	3,155,406				1,471,382				924,061			

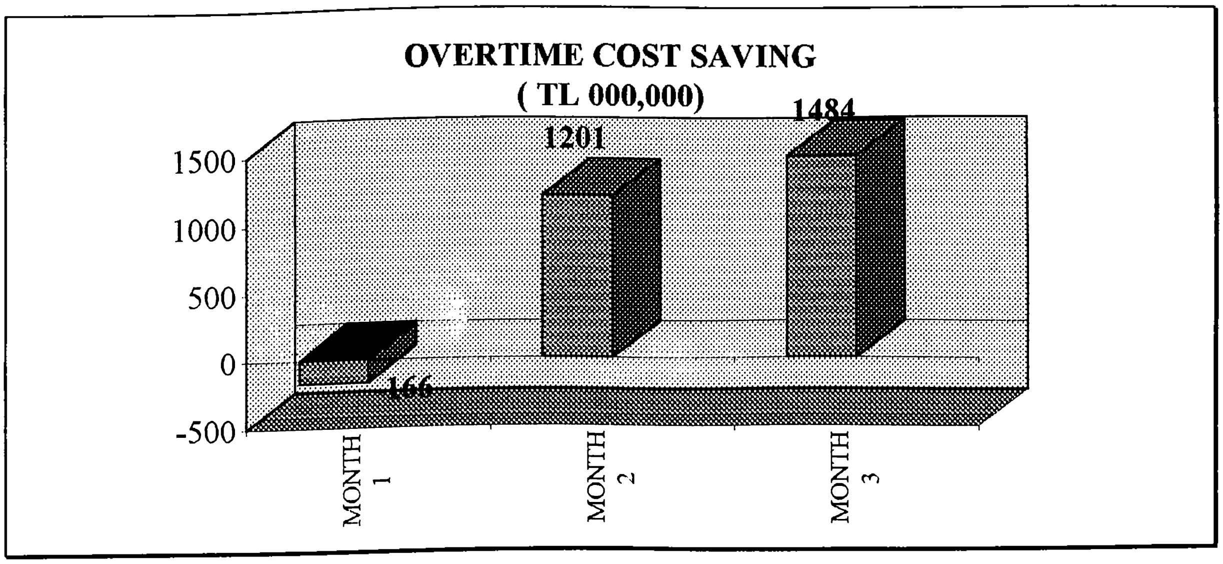
By the new lot sizing method, a significant saving is made in overtime costs. Increase in savings for overtime cost is mainly because of smaller lots. The parts which have high holding cost per unit are preferred to be manufactured in smaller lots if the setup cost is not so significant. These high holding cost parts are generally the ones which absorbs more resource in the machining shop, so that the total value add to the part is high. When these high resource requiring parts are manufactured in smaller sizes, the machines get loaded in higher frequencies but for smaller duration.

Exh 5. Overtime Costs Comparison



The savings from overtime increases as the first week load is absorbed by the machines. Savings from overtime cost can be seen in Exh 6.

Exh.6. Overtime Cost Saving



The Planner is free to change the production schedule to balance capacities. Here it is very important that, the planner, if he knows the machines, can think that some machines can be used by some parts although standard routings do not mention about it. So, grouping of machines can be done by the planner and capacity loads to the groups of machines can be analyzed.

The possible production schedules can easily be evaluated by a button click. This gives the planner the option of experiencing some of the alternatives to find out the best fit to the current situation.

The capacity allocation is not done by the computer itself. The main reasons are the computational difficulties with the capacity costs trade off between setup and inventory costs. It is not easy to find an algorithm which will trade off between these three factors. Also, the new solution of the computer can eliminate some bottlenecks while creating new ones. An infinite loop can be inevitable to find out the solution.

Another reason is the impossibility of communicating the experience of the planner to the computer. The groups of machines which can do each others' work, the machines which have higher breakdowns whenever high loaded, the utilization of machines, the quality problems at the raw materials, although possible, are not loaded to the computer.

VII.6. Total Costs

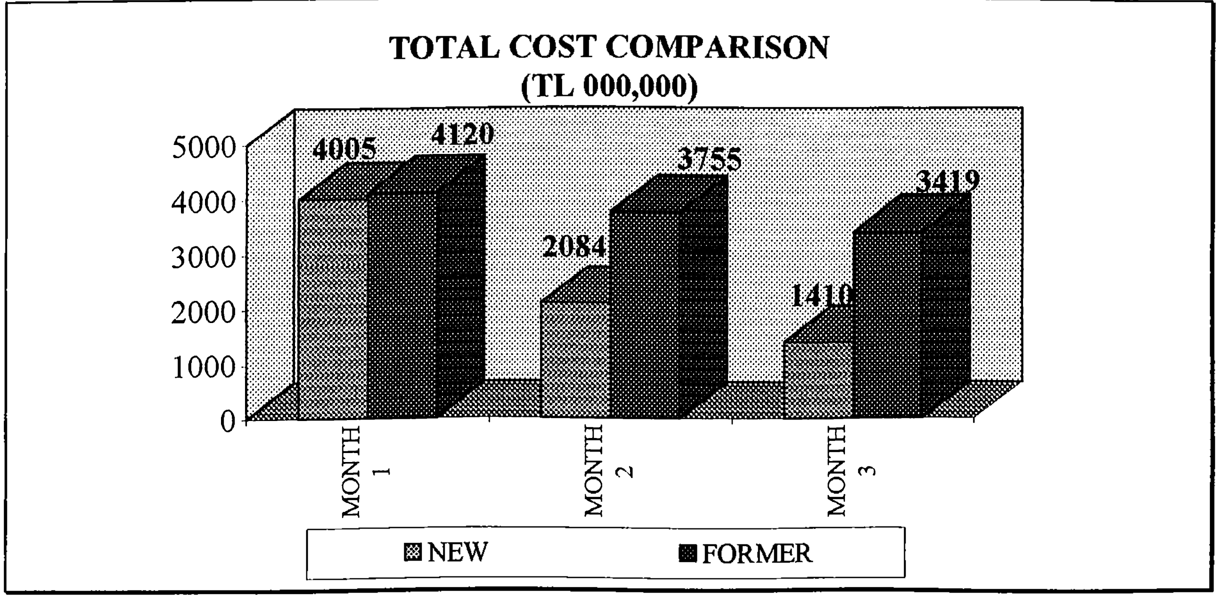
Silver-Meal Heuristic is used to decrease total cost by trade off between setup and inventory holding costs.Total Cost summary can be seen in Table 9.

Table 9 Total Costs of Parts

	1st Month		2nd Month		3rd Month	
	Silver-Meal	Former	Silver-Meal	Former	Silver-Meal	Former
Setup Cost	540,634,107	397,343,564	360,995,026	385,531,302	208,860,844	358,785,223
Inventory Holding	309,704,447	734,026,206	243,822,973	697,750,705	201,912,180	652,878,557
Subtotal	**850,338,554**	**1,131,369,770**	**613,318,999**	**1,083,282,007**	**486,507,024**	**1,011,663,680**
Overtime	3,155,466,000	2,989,111,219	1,471,382,000	2,672,464,314	924,061,000	2,408,126,049
Total Cost	**4,005,804,554**	**4,120,480,989**	**2,084,700,999**	**3,755,746,321**	**1,410,568,024**	**3,419,789,729**

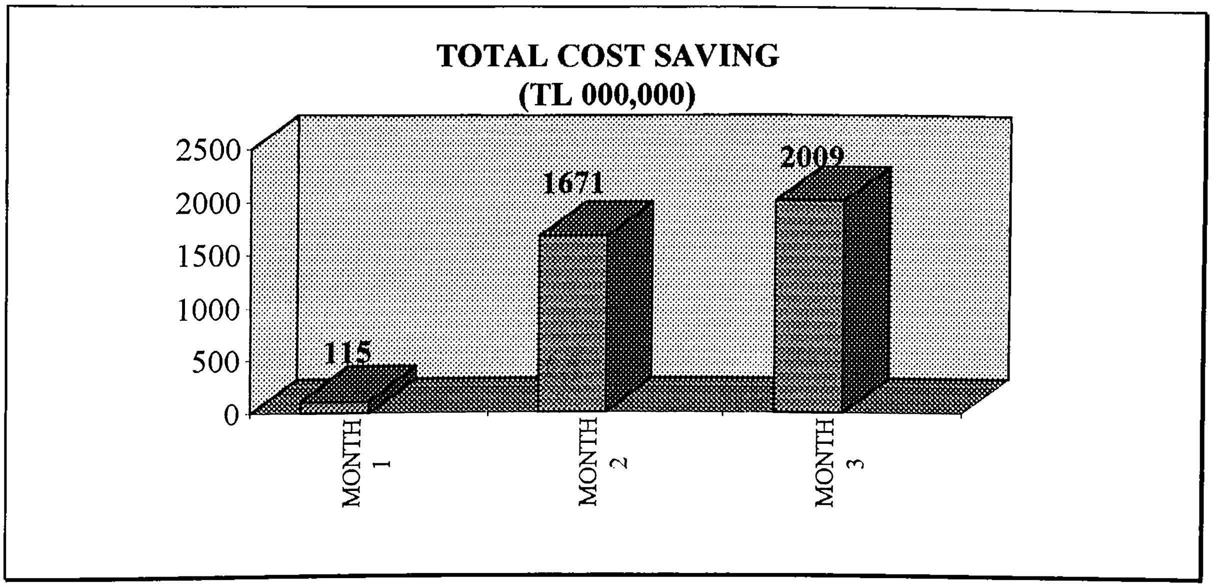
The total cost comparison shows that, there is a significant cost reduction by the Silver-Meal Heuristic. This can be seen in Exh. 7.

Exh 7. Total Cost Comparison



The savings from the new scheduling process is increasing, because of the increases in savings of the setup, inventory and capacity costs, analyzed in earlier sections. The savings can be seen in Exh 8.

Exh 8. Total Cost Saving



The total costs are proved to be better in Silver-Meal Lot Sizing Module. And as the system reach to the steady state, costs are decreasing and the savings from the new lot sizing is increasing. Since there is no former production which satisfied the first week's demand, the module gives production in the first week for all parts. So, the capacity load is maximum and overtime cost is very high at the first week. But as the system reaches to the third month, the overtime costs are even smaller than the former system.

The decrease in overtime costs are just because of smaller lot sizes for the high inventory cost items, and bigger lot sizes for less inventory cost items. Since, the inventory holding costs are directly proportional with the processing times, it means that the long processed items are divided into smaller lots, so the queuing time for the waiting parts are decreasing. And for the low inventory holding costs, the items processed in short time, and setups absorb more time, then by the aggregation of these smaller lots, the other waiting parts do not queue while frequent setups are made for the part.

Also, the real figures of WIP inventory which considers the raw material, labor and overhead costs show that at 31st March 1995, the WIP was TL163 billion, whereas at 2nd May 1995, the WIP stock was decreased to TL85 billion.

CHAPTER VIII

CONCLUSION

VIII.1. Database Usage

The technological developments in computer technology enables individuals to increase their effectiveness by the very customized and flexible applications in microcomputers. One of these application areas is the production planning activities.

In this study, the production planning activities of Türk Traktör Fabrikası Machining Shop is designed in MS Excel Ver.5. and MS Access Ver.2. to utilize the Silver-Meal Heuristic within the company's MRP module. The data is transferred from the IBM A6 system, and designed in MS Access Ver.2. By the usage of some special functions and macros of MS Excel Ver.5, the data processed and the queries designed help the planner to see effects of the scheduling, on machine capacities, inventories and setup times.

The Silver-Meal Heuristic is used for trade off between setup and inventory holding costs. The algorithm, although is not an optimization, determine the lot sizes to decrease total cost of inventory holding and setup. The new lot sizes also help to increase the throughput while decreasing inventories. The new lot sizes divide the capacity requirements to the planning horizon if the process times are longer and inventory holding costs are higher for that reason. By division of lot sizes into smaller batches, the total waiting time of the other parts requiring the same resource decrease and throughput increase. Also by the increased throughput and smaller batches of high cost parts, the inventory levels decrease.

The new Lot Sizing Module is designed to be very flexible and helping. The schedules of parts production, the capacity requirements of each part from necessary resources, the total capacity requirements on resources, the inventories, inventory holding costs and overtime costs can be seen by button clicks. The changes can be done to the schedule after the analysis of planner, and the effects to the system can be seen immediately.

By this new system, planning activities take shorter time. To access a central database, to take process time from the resource and wait for the printer to get document takes approximately one day. By the help of database, the total time to run the program and get the reports is one hour. Since the capacity and cost analysis is not done in the current system, there is a 90% improvement in the planning time. The analysis and changes in schedules can be done by the planner on his/her own computer or even on a laptop for the remaining of the day.

VIII.2. Effects On Cost Figures

The cost figures analyzed in this study are the setup, inventory holding and overtime costs. By the use of Silver-Meal Heuristic, setup costs are tried to be decreased by producing as many parts as possible to gain from setups, for the parts which have high setup costs. But while increasing lot sizes, the inventory holding costs can increase because of increase in average inventory level. So, for high value add parts, by releasing material just enough for satisfying demands of shorter periods, inventories are decreased. Silver-Meal Heuristic determines for which part the lots must be increased, and for which parts, lot sizes must be decreased. The cost reduction obtained from the system was about 2 billion in the last month.

And as the system reach to the steady state, costs are decreasing and the savings from the new lot sizing is increasing. The main reason, the Silver Meal gives production for every part in the first week, because there is no former production which satisfied the first week's demand. So, the capacity load is maximum and overtime cost is very high. But as the system reaches to the third month, the overtime costs are even smaller than the former system.

The decrease in overtime costs are just because of smaller lot sizes for the high inventory cost items, and bigger lot sizes for less inventory cost items. Since, the inventory holding costs are directly proportional with the processing times, it means that the long processed items are divided into smaller lots, so the queuing time for the waiting parts are decreasing. And for the low inventory holding costs, the items processed in short time, and setups absorb more time, then by the aggregation of these smaller lots, the other waiting parts do not queue while frequent setups are made for the part.

Also, the real figures of WIP inventory which considers the raw material, labor and overhead costs show that at 31st March 1995, the WIP was TL163 billion, whereas at 2nd May 1995, the WIP stock was decreased to TL85 billion.

V.III. Further Research Items

MRP lot sizing module uses setup and inventory holding cost for determining lot sizes. After the determination of lot sizes, planner search for capacity overloads and try to divide lots to decrease overtime costs. The lot sizing module can be improved by dynamic lead time configuration. The schedules can be converted into waiting times for each machine and lead time for a particular lot can be estimated. After the determination of lead times, forward scheduling can be done to obtain real capacity overloads and overtime costs.

A simulation model can be integrated to the module, which will get input of order releases, processing times and setups from the module, and gives out average inventory levels, machine utilization rates, queues and waiting times in front of the machines as output. By this way, the effect of lot sizing decisions on throughput and inventory levels can be analyzed.

REFERENCES

1. Baudin Michel, Manufacturing Systems Analysis, Yourdon Press Computing Series, 1991.

2. Coleman and McKnew , An Improved heuristic for Multilevel Lot Sizing in MRP, Decision Sciences, Vol 22., 1991.

3. Dauzere Peres, and Lassare J.B., Integration Of Lot Sizing and Scheduling, European Journal Of Operations Research, Vol. 75. pp 413-426, 1994.

4. Dillenberger , C., Escudero L.F., Resource Allocation and Scheduling with Setup Times, European Journal of Operations Research, Vol 75, pp275-286, 1994.

5.Eppen, G., Martin K., and Schrage L., A Scenerio Approach to Capacity Planning, Operations Research Society of America, July-August 1989.

6. Kuik, and Salomon, Multi-level Lot Sizing Problem : Evaluation of a simulated-annealing Heuristic, European Journal Of Operations Research, Vol 45. pp.25-37, 1991

7. Olhager, J., and Rapp, B., Balancing Capacity and Lot Sizes, European Journal Of Operations Research, Vol. 19, pp 337-344, 1985.

8. Steinberg and Napier, Optimal Multi-level Lot Sizing for Requirements Planning Systems, Management Science, 1980.

9. Tempelmeier, H., and Helber S., Dynamic Multi-item Multi-level Capacitated Lot Sizing, European Journal of Operations Research, Vol.75, pp427-440, 1994.

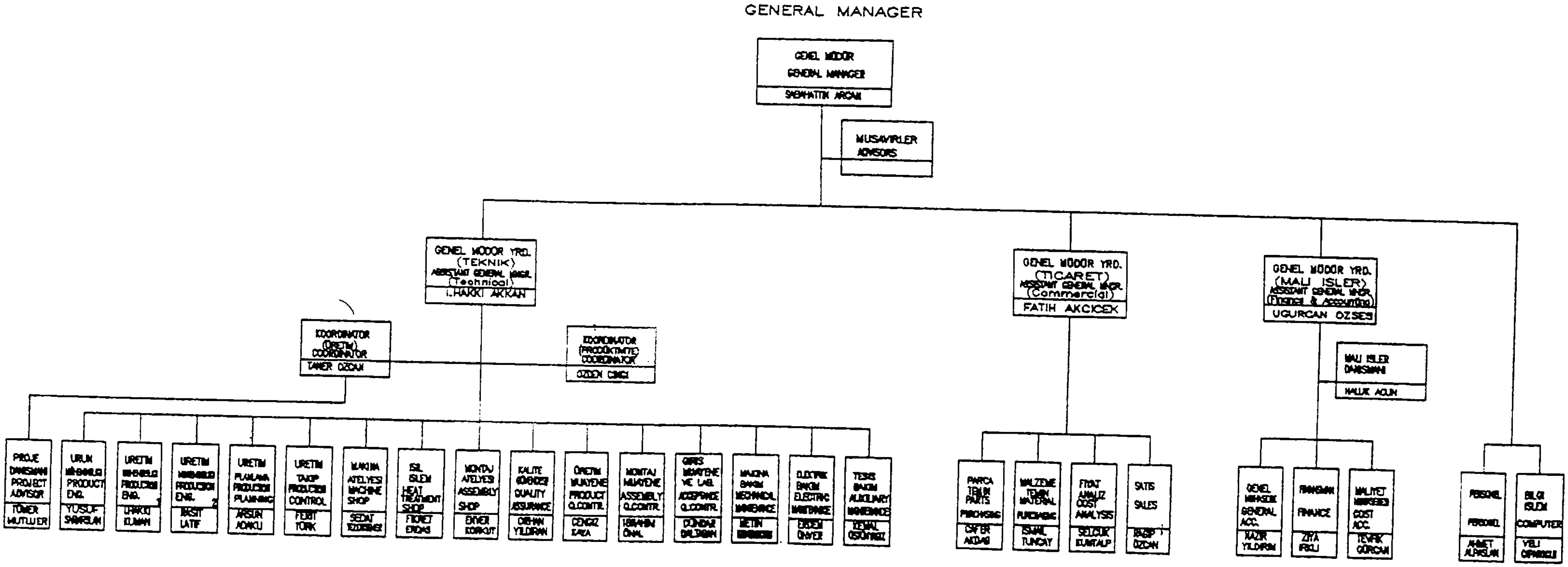
10. Trigeiro, Thomas and McClain, Capacitated lot Sizing with Set Up Times, Management Science, March 1989.

APPENDICES

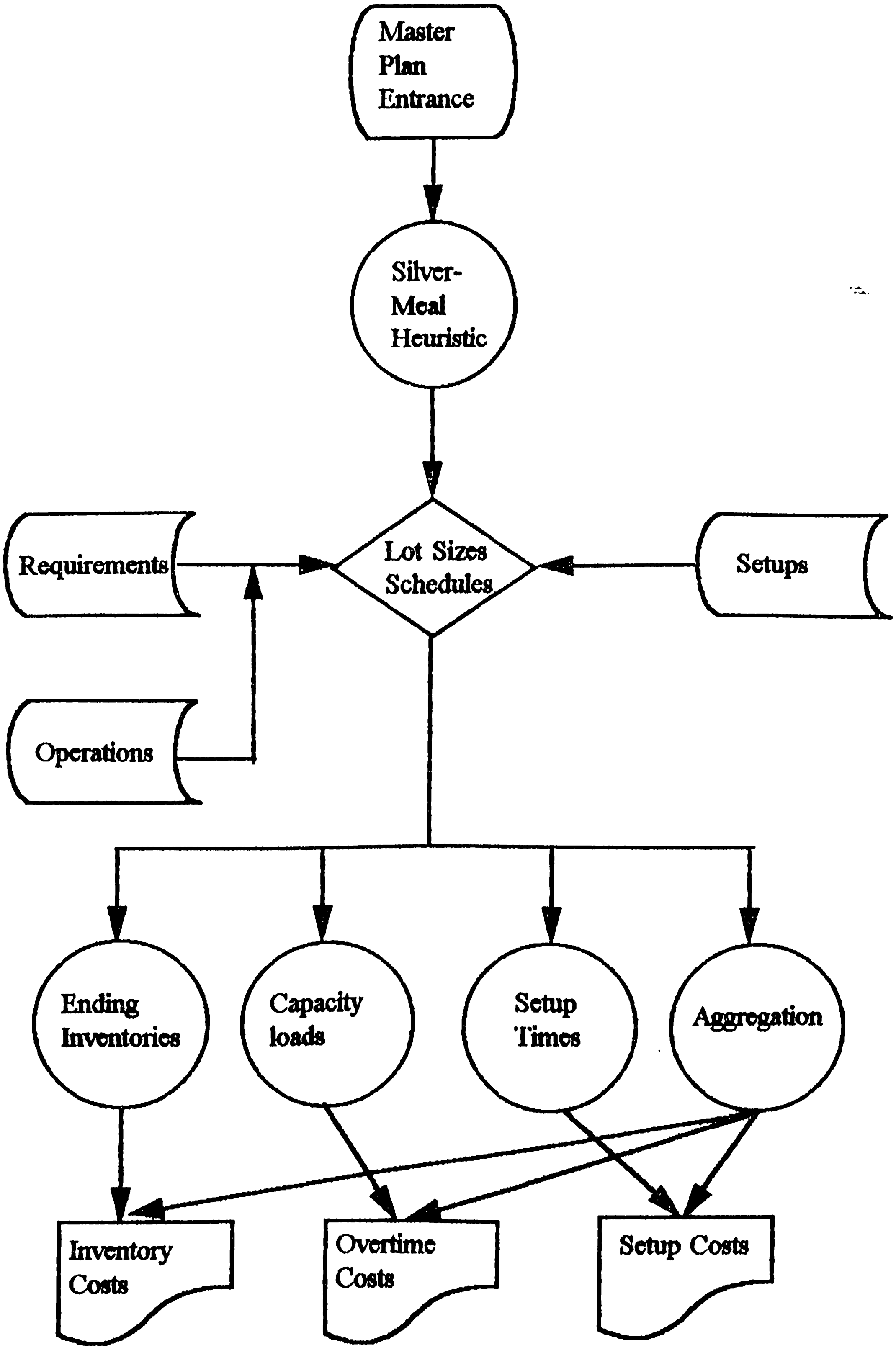
APPENDIX A

TURK TRAKTOR ve ZIRAAT MAKINELERI A.S.
YONETIM SEMASI
ORGANIZATION CHART

1/1/1983



App A1. Organization Structure Of Türk Traktör Fabrikası



App. A2. General Framework Of MRP Lot Sizing Module

```
Silver-Meal  Macro
'Macro recorded 2/9/1994 by Berkhan N. Esmer
'
Sub SMALG()
  Range("AD2:AO529").Select
  Selection.ClearContents
  lastweek = 12
  timehor = 18
  interest = 0.025
  t_parts = 531
  For n = 2 To t_parts
  released = 0
  lastrel = 1
    For j = 1 To lastweek
    If released = 0 Then
      periodcost = ActiveSheet.Cells(n, 2).Value
      released = ActiveSheet.Cells(n, j + 3).Value
      prevcost = periodcost
      ActiveSheet.Cells(n, j + lastweek + 4).Value = periodcost
      If j = lastweek Then
      ActiveSheet.Cells(n, lastrel + 29).Value = released
      GoTo son
      End If
    Else
      periodcost = ((prevcost * (j - lastrel)) + ((ActiveSheet.Cells(n, j + 3).Value) * (j - lastrel)
* (ActiveSheet.Cells(n, 3).Value * interest))) / (j - lastrel + 1)
      If periodcost < prevcost Then
        prevcost = periodcost
        released = released + ActiveSheet.Cells(n, j + 3).Value
      ActiveSheet.Cells(n, j + lastweek + 4).Value = periodcost
      If j = lastweek Then
        ActiveSheet.Cells(n, lastrel + 29).Value = released
        GoTo son
      End If
    Else ActiveSheet.Cells(n, lastrel + 29).Value = released
    lastrel = j
    released = 0
    j = j - 1
    End If
  End If
  Next j
son:  Next n
End Sub
```

App. A3. Silver-Meal Heuristic Code

APPENDIX B

FIGURES

KOD	54JN	54SP	5556	6056	7056	70DT	8066	80DT	CR1	CR2	CR3	CR4	CR5	CR7	CR8
110052 5085945			1	1	1	1	1	1							
117039 598307	0	0													
119143 5086381	0	0													
119144 599155	0	0													
121008 4973385															
121008 5085359			1	1	1	1			1	1	1	1	1	1	1
121008 5087104	1	1													
121008 5157941							1	1							
121008 599905	0	0													
121009 5149253			0	0	0	0			1	1	1	1	1	1	1
121009 599824	1	1	1	1	1	1									
121015 5086765							1								
121573 5118742			1	1	1	1	1	1							
121573 5118743			1	1	1	1	1	1							
121596 5085923			1	1											
121596 5085924			0		1										
121596 5085925						1									
121596 5085926							1	1							
121596 5086672	0	1													
125000 340362															
125000 340364															
125000B5085401															
125004 4974473															
125004 5086996	0	0	0	0	0	0									
125005 5085405	0	0	0	0	0	0									
125006 4968735															
125006 5085417	1	1	1	1	1	1	1	1							
125007 341438	1	1	1	1	1	1	1	1							
125007 5085419	0	0	0	0	0	0	0	0							
125007 596247	0	0	0	0	0	0	0	0							
125008 5085406	0	0	0	0	0	0									
125009 596251															
125011 5085413															
125012 5085414															
125015 5085407	0	0	0	0	0	0									
125015 596258															
125017 341437	1	1	1	1	1	1	1	1							
125017 4974474	0	0	0	0	0	0	0	0							

Fig. 1a. BOM Structure

Properties

Date Created:	27/2/95 1:37:43 PM	Def. Updatable:	Yes
Last Updated:	4/4/95 12:20:17 PM	Record Count:	649

Columns

Name	Type	Size
KOD	Text	255
MLZ	Text	255
54JN	Number (Double)	8
54SP	Number (Double)	8
5556	Number (Double)	8
6056	Number (Double)	8
7056	Number (Double)	8
70DT	Number (Double)	8
8066	Number (Double)	8
80DT	Number (Double)	8
CR1	Number (Double)	8
CR2	Number (Double)	8
CR3	Number (Double)	8
CR4	Number (Double)	8
CR5	Number (Double)	8
CR7	Number (Double)	8
CR8	Number (Double)	8

Fig. 1b. BOM Table Structure

Properties

Date Created:	14/3/95 7:12:42 AM	Def. Updatable:	Yes
Last Updated:	14/3/95 9:14:35 AM	Record Count:	7962

Columns

Name	Type	Size
Part No	Text	255
Op	Number (Double)	8
M/C	Text	255
Setup	Number (Double)	8

Fig.3. Setup Table Structure

Properties

Date Created:	15/3/95 11:41:45 AM	Def. Updatable:	Yes
Last Updated:	15/3/95 11:41:46 AM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW SETUP.[Part No], SETUP.Op, SETUP.[M/C], SETUP.Setup, tezgahişç.İŞÇİLİK,
tezgahişç.GGO, ([tezgahişç]![İŞÇİLİK])*([SETUP]![Setup])*(1+[TEZGAHYÜK]![GGO]) AS işç
FROM tezgahişç RIGHT JOIN SETUP ON tezgahişç.TEZGAH = SETUP.[M/C];
```

Columns

Name	Type	Size
Part No	Text	255
Op	Number (Double)	8
M/C	Text	255
Setup	Number (Double)	8
İŞÇİLİK	Number (Double)	8
GGO	Number (Double)	8
işç	Number (Double)	8

Fig.4. Query Structure for Setup Costs Of Operations

Properties

Date Created:	15/3/95 11:51:02 AM	Def. Updatable:	Yes
Last Updated:	15/3/95 2:31:01 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW setupcostofop.[Part No], Sum(setupcostofop.işç) AS SumOfişç
FROM setupcostofop
GROUP BY setupcostofop.[Part No]
ORDER BY Sum(setupcostofop.işç);
```

Columns

Name	Type	Size
Part No	Text	255
SumOfişç	Number (Double)	8

Fig.5. Query Structure for Setup Cost Of Parts

Properties

Date Created:	6/3/95 7:42:56 AM	Def. Updatable:	Yes
Last Updated:	8/5/95 2:24:11 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW OPERASYON.PARÇA AS PART, Sum(OPERASYON.BİRİKİMLİ) AS
HOLDINGCOST
FROM OPERASYON
GROUP BY OPERASYON.PARÇA
ORDER BY OPERASYON.PARÇA;
```

Columns

Name	Type	Size
PART	Text	255
HOLDINGCOST	Number (Double)	8

Fig. 6. Query Structure for Holding Cost Of Parts

Properties

Date Created:	8/5/95 3:57:19 PM	Def. Updatable:	Yes
Last Updated:	8/5/95 4:42:05 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW MACHINELOADBYWEEK.TEZGAH,
IIf([MACHINELOADBYWEEK]![SumOf1LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf1LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST1,
IIf([MACHINELOADBYWEEK]![SumOf1LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf1LOADSET]-
4800),0) AS OVERTIME1,
IIf([MACHINELOADBYWEEK]![SumOf2LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf2LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST2,
IIf([MACHINELOADBYWEEK]![SumOf2LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf2LOADSET]-
4800),0) AS OVERTIME2,
IIf([MACHINELOADBYWEEK]![SumOf3LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf3LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST3,
IIf([MACHINELOADBYWEEK]![SumOf3LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf3LOADSET]-
4800),0) AS OVERTIME3,
IIf([MACHINELOADBYWEEK]![SumOf4LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf4LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST4,
IIf([MACHINELOADBYWEEK]![SumOf4LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf4LOADSET]-
4800),0) AS OVERTIME4,
IIf([MACHINELOADBYWEEK]![SumOf5LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf5LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST5,
IIf([MACHINELOADBYWEEK]![SumOf5LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf5LOADSET]-
4800),0) AS OVERTIME5,
IIf([MACHINELOADBYWEEK]![SumOf6LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf6LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST6,
IIf([MACHINELOADBYWEEK]![SumOf6LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf6LOADSET]-
4800),0) AS OVERTIME6,
IIf([MACHINELOADBYWEEK]![SumOf7LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf7LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST7,
IIf([MACHINELOADBYWEEK]![SumOf7LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf7LOADSET]-
4800),0) AS OVERTIME7,
IIf([MACHINELOADBYWEEK]![SumOf8LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf8LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST8,
IIf([MACHINELOADBYWEEK]![SumOf8LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf8LOADSET]-
4800),0) AS OVERTIME8,
IIf([MACHINELOADBYWEEK]![SumOf9LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf9LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST9,
IIf([MACHINELOADBYWEEK]![SumOf9LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf9LOADSET]-
4800),0) AS OVERTIME9,
IIf([MACHINELOADBYWEEK]![SumOf10LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf10LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST10,
IIf([MACHINELOADBYWEEK]![SumOf10LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf10LOADSET]-
4800),0) AS OVERTIME10,
IIf([MACHINELOADBYWEEK]![SumOf11LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf11LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST11,
IIf([MACHINELOADBYWEEK]![SumOf11LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf11LOADSET]-
4800),0) AS OVERTIME11,
IIf([MACHINELOADBYWEEK]![SumOf12LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf12LOADSET]-
4800)*[tezgahişç]![İŞÇİLİK]*(1+[tezgahişç]![GGO]),0) AS OVERCOST12,
IIf([MACHINELOADBYWEEK]![SumOf12LOADSET]>4800,([MACHINELOADBYWEEK]![SumOf12LOADSET]-
4800),0) AS OVERTIME12
FROM tezgahişç INNER JOIN MACHINELOADBYWEEK ON tezgahişç.TEZGAH =
MACHINELOADBYWEEK.TEZGAH;
```

Fig.7a. Query Structure For Overtime Costs

Columns

Name	Type	Size
TEZGAH	Text	255
OVERCOST1	Number (Double)	8
OVERTIME1	Number (Double)	8
OVERCOST2	Number (Double)	8
OVERTIME2	Number (Double)	8
OVERCOST3	Number (Double)	8
OVERTIME3	Number (Double)	8
OVERCOST4	Number (Double)	8
OVERTIME4	Number (Double)	8
OVERCOST5	Number (Double)	8
OVERTIME5	Number (Double)	8
OVERCOST6	Number (Double)	8
OVERTIME6	Number (Double)	8
OVERCOST7	Number (Double)	8
OVERTIME7	Number (Double)	8
OVERCOST8	Number (Double)	8
OVERTIME8	Number (Double)	8
OVERCOST9	Number (Double)	8
OVERTIME9	Number (Double)	8
OVERCOST10	Number (Double)	8
OVERTIME10	Number (Double)	8
OVERCOST11	Number (Double)	8
OVERTIME11	Number (Double)	8
OVERCOST12	Number (Double)	8
OVERTIME12	Number (Double)	8

Fig.7b. Query Fields For Overtime Costs

Properties

Date Created:	18/4/95 8:26:38 AM	Def. Updatable:	Yes
Last Updated:	18/4/95 4:24:34 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW OPERASYON.PARÇA, OPERASYON.OP, OPERASYON.TEZGAH,
([IHT]![1IHT])*([operasyon]![SÜRE]) AS 1WKLOAD, ([IHT]![2IHT])*([operasyon]![SÜRE]) AS 2WKLOAD,
([IHT]![3IHT])*([operasyon]![SÜRE]) AS 3WKLOAD, ([IHT]![4IHT])*([operasyon]![SÜRE]) AS 4WKLOAD,
([IHT]![5IHT])*([operasyon]![SÜRE]) AS 5WKLOAD, ([IHT]![6IHT])*([operasyon]![SÜRE]) AS 6WKLOAD,
([IHT]![7IHT])*([operasyon]![SÜRE]) AS 7WKLOAD, ([IHT]![8IHT])*([operasyon]![SÜRE]) AS 8WKLOAD,
([IHT]![9IHT])*([operasyon]![SÜRE]) AS 9WKLOAD, ([IHT]![10IHT])*([operasyon]![SÜRE]) AS 10WKLOAD,
([IHT]![11IHT])*([operasyon]![SÜRE]) AS 11WKLOAD, ([IHT]![12IHT])*([operasyon]![SÜRE]) AS 12WKLOAD
FROM IHT INNER JOIN OPERASYON ON IHT.[PART NO] = OPERASYON.PARÇA;
```

Columns

Name	Type	Size
PARÇA	Text	255
OP	Number (Double)	8
TEZGAH	Text	255
1WKLOAD	Number (Double)	8
2WKLOAD	Number (Double)	8
3WKLOAD	Number (Double)	8
4WKLOAD	Number (Double)	8
5WKLOAD	Number (Double)	8
6WKLOAD	Number (Double)	8
7WKLOAD	Number (Double)	8
8WKLOAD	Number (Double)	8
9WKLOAD	Number (Double)	8
10WKLOAD	Number (Double)	8
11WKLOAD	Number (Double)	8
12WKLOAD	Number (Double)	8

Fig.8. Query Structure For Machine Load

Properties

Date Created:	18/4/95 4:37:21 PM	Def. Updatable:	Yes
Last Updated:	21/4/95 4:12:45 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW machineload.PARÇA, machineload.OP, machineload.TEZGAH,
IIf([machineload]![1wkload]>0,[SETUP]![Setup]+[machineload]![1WKLOAD],0) AS 1LOADSET,
IIf([machineload]![2wkload]>0,[SETUP]![Setup]+[machineload]![2WKLOAD],0) AS 2LOADSET,
IIf([machineload]![3wkload]>0,[SETUP]![Setup]+[machineload]![3WKLOAD],0) AS 3LOADSET,
IIf([machineload]![4wkload]>0,[SETUP]![Setup]+[machineload]![4WKLOAD],0) AS 4LOADSET,
IIf([machineload]![5wkload]>0,[SETUP]![Setup]+[machineload]![5WKLOAD],0) AS 5LOADSET,
IIf([machineload]![6wkload]>0,[SETUP]![Setup]+[machineload]![6WKLOAD],0) AS 6LOADSET,
IIf([machineload]![7wkload]>0,[SETUP]![Setup]+[machineload]![7WKLOAD],0) AS 7LOADSET,
IIf([machineload]![8wkload]>0,[SETUP]![Setup]+[machineload]![8WKLOAD],0) AS 8LOADSET,
IIf([machineload]![9wkload]>0,[SETUP]![Setup]+[machineload]![9WKLOAD],0) AS 9LOADSET,
IIf([machineload]![10wkload]>0,[SETUP]![Setup]+[machineload]![10WKLOAD],0) AS 10LOADSET,
IIf([machineload]![11wkload]>0,[SETUP]![Setup]+[machineload]![11WKLOAD],0) AS 11LOADSET,
IIf([machineload]![12wkload]>0,[SETUP]![Setup]+[machineload]![12WKLOAD],0) AS 12LOADSET
FROM machineload LEFT JOIN SETUP ON (machineload.TEZGAH = SETUP.[M/C]) AND (machineload.OP =
SETUP.Op) AND (machineload.PARÇA = SETUP.[Part No]);
```

Columns

Name	Type	Size
PARÇA	Text	255
OP	Number (Double)	8
TEZGAH	Text	255
1LOADSET	Number (Double)	8
2LOADSET	Number (Double)	8
3LOADSET	Number (Double)	8
4LOADSET	Number (Double)	8
5LOADSET	Number (Double)	8
6LOADSET	Number (Double)	8
7LOADSET	Number (Double)	8
8LOADSET	Number (Double)	8
9LOADSET	Number (Double)	8
10LOADSET	Number (Double)	8
11LOADSET	Number (Double)	8
12LOADSET	Number (Double)	8

Fig.9. Query Structure For Machine Load and Setup

Properties

Date Created:	18/4/95 9:14:33 AM	Def. Updatable:	Yes
Last Updated:	21/4/95 4:50:44 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW MACHINELOADANDSETUP.TEZGAH,
Sum(MACHINELOADANDSETUP.[1LOADSET]) AS SumOf1LOADSET,
Sum(MACHINELOADANDSETUP.[2LOADSET]) AS SumOf2LOADSET,
Sum(MACHINELOADANDSETUP.[3LOADSET]) AS SumOf3LOADSET,
Sum(MACHINELOADANDSETUP.[4LOADSET]) AS SumOf4LOADSET,
Sum(MACHINELOADANDSETUP.[5LOADSET]) AS SumOf5LOADSET,
Sum(MACHINELOADANDSETUP.[6LOADSET]) AS SumOf6LOADSET,
Sum(MACHINELOADANDSETUP.[7LOADSET]) AS SumOf7LOADSET,
Sum(MACHINELOADANDSETUP.[8LOADSET]) AS SumOf8LOADSET,
Sum(MACHINELOADANDSETUP.[9LOADSET]) AS SumOf9LOADSET,
Sum(MACHINELOADANDSETUP.[10LOADSET]) AS SumOf10LOADSET,
Sum(MACHINELOADANDSETUP.[11LOADSET]) AS SumOf11LOADSET,
Sum(MACHINELOADANDSETUP.[12LOADSET]) AS SumOf12LOADSET
FROM MACHINELOADANDSETUP
GROUP BY MACHINELOADANDSETUP.TEZGAH;
```

Columns

Name	Type	Size
TEZGAH	Text	255
SumOf1LOADSET	Number (Double)	8
SumOf2LOADSET	Number (Double)	8
SumOf3LOADSET	Number (Double)	8
SumOf4LOADSET	Number (Double)	8
SumOf5LOADSET	Number (Double)	8
SumOf6LOADSET	Number (Double)	8
SumOf7LOADSET	Number (Double)	8
SumOf8LOADSET	Number (Double)	8
SumOf9LOADSET	Number (Double)	8
SumOf10LOADSET	Number (Double)	8
SumOf11LOADSET	Number (Double)	8
SumOf12LOADSET	Number (Double)	8

Fig.10. Query Structure For Weekly Loads on Machines

PRODUCTION PLAN

	54JN	54 SP	55 56	60 56	70 56	70 DT	80 66	80 DT	CR1	CR2	CR3	CR4	CR5	CR7	CR8
1 WEEK	15	55	30	40	55	15	0	10	5	5	0	3	5		3
2 WEEK	15	55	30	40	55	15	0	10	5	5	0	3	5		3
3 WEEK	15	55	30	40	55	15	0	10	5	5	0	3	5		3
4 WEEK	15	55	30	40	55	15	0	10	5	5	0	3	5		3
5 WEEK	15	65	25	37	45	0	13		5	4	1	9	5		0
6 WEEK	15	65	25	37	45	0	13		5	4	1	9	5		0
7 WEEK	15	65	25	37	45	0	13		5	4	1	9	5		0
8 WEEK	15	65	25	37	45	0	13		5	4	1	9	5		0
9 WEEK	10	75	35	25	35	0	0	0	7	3	5	7	5		2
10 WEEK	10	75	35	25	35	0	0	0	7	3	5	7	5		2
11 WEEK	10	75	35	25	35	0	0	0	7	3	5	7	5		2
12 WEEK	10	75	35	25	35	0	0	0	7	3	5	7	5		2

Fig.11. Production Plan For the Application Period

PART NO	SETUP	HC	1IHT	2IHT	3IHT	4IHT	5IHT	6IHT	7IHT	8IHT	9IHT	10IHT	11IHT	12IHT
110028 568635	631066.852	1941.74	0	0	0	0	0	0	0	0	0	0	0	0
110052 5085945	713917.461	50240.1	0	0	0	0	0	0	0	0	0	0	0	0
117039 598307	631066.052	8543.66	0	0	0	0	0	0	0	0	0	0	0	0
119143 5086381	1019415.684	9611.6	0	0	0	0	0	0	0	0	0	0	0	0
119144 599155	1310677.308	11553.34	161	161	161	161	131	131	131	131	124	124	124	124
121008 4973385	427440.132	236472.5	70	70	70	70	80	80	80	80	85	85	85	85
121008 5085359	1454764.416	275580.72	10	10	10	10	13	13	13	13	0	0	0	0
121000 5087104	2735008.222	399003.53	0	0	0	0	0	0	0	0	0	0	0	0
121008 5157941	2607440.955	460216.64	21	21	21	21	24	24	24	24	29	29	29	29
121000 599905	3506508.642	399003.53	210	210	210	210	187	187	187	187	180	180	180	180
121009 5149253	1971847.992	232205.96	0	0	0	0	13	13	13	13	0	0	0	0
121009 599824	1971847.992	204626.72	150	150	150	150	120	120	120	120	95	95	95	95
121015 5086765	759559.68	48324.44	150	150	150	150	120	120	120	120	95	95	95	95
121573 5118742	0	14563.08	70	70	70	70	62	62	62	62	60	60	60	60
121573 5118743	1125726.507	70502.79	55	55	55	55	45	45	45	45	35	35	35	35
121596 5085923	1127304.355	27737.16	15	15	15	15	0	0	0	0	0	0	0	0
121596 5085924	1173416.03	20843.83	10	10	10	10	13	13	13	13	0	0	0	0

Fig.12. A Range From Gross Requirements of Parts

PART NO	1 WEEK	2 WEEK	3 WEEK	4 WEEK	5 WEEK	6 WEEK	7 WEEK	8 WEEK	9 WEEK	10 WEEK	11 WEEK	12 WEEK
110028 568635	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9
110052 5085945	713917.5	713917.5	713917.5	713917.5	713917.5	713917.5	713917.5	713917.5	713917.5	713917.5	713917.5	713917.5
117039 598307	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9	631066.9
119143 5086381	1019416	1019416	1019416	1019416	1019416	1019416	1019416	1019416	1019416	1019416	1019416	1019416
119144 599155	1310677	673939.5	474094.2	383472	330993.4	301052.6	283990.6	274977.8	269893.4	268691.1	1310677	669664.8
121008 4973385	427440.1	379250.8	427440.1	379250.8	427440.1	402898.1	427440.1	402898.1	427440.1	414721.7	427440.1	414721.7
121008 5085359	1454764	754940.3	540037.6	446365.3	414413	405053.4	1454764	763207.7	508805.1	381603.9	305283.1	254402.6
121008 5087104	2735808	2735808	2735808	2735808	2735808	2735808	2735808	2735808	2735808	2735808	2735808	2735808
121008 5157941	2607441	1400366	1062438	941796.7	930160.6	2607441	1414172	1090051	1017732	2607441	1437183	1136073
121008 599905	3506509	2591162	3506509	2591162	3506509	2499391	3506509	2499391	3506509	2471461	3506509	2471461
121009 5149253	1971848	1971848	1971848	1971848	1971848	1693518	1503336	1368245	1216218	1094596	995087.6	912163.7
121009 599824	1971848	1292864	1271163	1971848	1231476	1148387	1971848	1231476	1080178	1971848	1180319	1046073
121015 5086765	759559.7	452266.5	398159.9	759559.7	437769.2	369165.2	363857.9	759559.7	425688.1	345003	327614.6	759559.7
121573 5118742	0	0	0	0	0	0	0	0	0	0	0	0
121573 5118743	1125727	606039.8	461595.2	410961.2	385291	379952.9	1125727	598189.5	435427.6	367784.7	338189.3	327617.7
121596 5085923	1127304	567812.8	384089.3	294307.8	235446.2	196205.2	168175.9	147153.9	130803.5	117723.1	107021	98102.6

Fig.13. A Range From Silver-Meal Period Costs

PART NO	1 WEEK	2 WEEK	3 WEEK	4 WEEK	5 WEEK	6 WEEK	7 WEEK	8 WEEK	9 WEEK	10 WEEK	11 WEEK	12 WEEK
110028 568635	0											
110052 5085945	0											
117039 598307	0											
119143 5086381	0											
119144 599155	1416										248	
121008 4973385	140		140		160		150		170		170	
121008 5085359	66						26					
121008 5087104	0											
121008 5157941	108					101				87		
121008 599905	420		420		374		374		380		360	
121009 5149253	52											
121009 599824	450			390			335			285		
121015 5086765	450			510				405				95
121573 5118742	70	70	70	70	62	62	62	62	60	60	60	60
121573 5118743	310						230					
121596 5085923	60											
121596 5085924	92											

Fig.14. A Range From Silver-Meal Lot Sizes

Properties

Date Created:	7/3/95 7:13:18 AM	Last Updated:	14/3/95 1:45:39 PM
Owner:	admin		

Actions

Name	Condition	Action	Argument	Value
		RunApp	Command Line:	c:\msoffice\excel\excel.exe c:\imalat\master.xls

Fig.15. Macro For Transfer Of Data Between Applications

Properties

Date Created:	14/3/95 1:45:26 PM	Last Updated:	8/5/95 12:14:28 PM
Owner:	admin		

Actions

Name	Condition	Action	Argument	Value
		OpenQuery	Query Name:	DELIHT
			View:	Datasheet
			Data Mode:	Edit
		OpenQuery	Query Name:	DELINV
			View:	Datasheet
			Data Mode:	Edit
		TransferSpreadsheet	Transfer Type:	Import
			Spreadsheet Type:	5
			Table Name:	IHT
			File Name:	C:\IMALAT\MASTER.XLS
			Has Field Names:	No
			Range:	wwrange
		TransferSpreadsheet	Transfer Type:	Import
			Spreadsheet Type:	5
			Table Name:	INV
			File Name:	C:\IMALAT\MASTER.XLS
			Has Field Names:	No
			Range:	inventories

Fig.16. Macro For Transfer of Data to MS Access Tables

Columns

Name	Type	Size
PART NO	Text	50
1IHT	Number (Double)	8
2IHT	Number (Double)	8
3IHT	Number (Double)	8
4IHT	Number (Double)	8
5IHT	Number (Double)	8
6IHT	Number (Double)	8
7IHT	Number (Double)	8
8IHT	Number (Double)	8
9IHT	Number (Double)	8
10IHT	Number (Double)	8
11IHT	Number (Double)	8
12IHT	Number (Double)	8

Fig.17. Table Structure To Hold Silver-Meal Lot Sizes

Properties

Date Created:	8/5/95 11:52:15 AM	Def. Updatable:	Yes
Last Updated:	8/5/95 12:15:23 PM	Record Count:	534

Columns

Name	Type	Size
part no	Text	50
holdcost	Number (Double)	8
1 week	Number (Double)	8
2 week	Number (Double)	8
3 week	Number (Double)	8
4 week	Number (Double)	8
5 week	Number (Double)	8
6 week	Number (Double)	8
7 week	Number (Double)	8
8 week	Number (Double)	8
9 week	Number (Double)	8
10 week	Number (Double)	8
11 week	Number (Double)	8
12 week	Number (Double)	8
TOTALCOST	Number (Double)	8

Fig.18. Table Structure To Hold Inventories

Properties

Date Created:	29/4/95 2:22:45 PM	Def. Updatable:	Yes
Last Updated:	29/4/95 2:25:43 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW IHT.[PART NO], IHT.[1IHT], IHT.[2IHT], IHT.[3IHT], IHT.[4IHT], IHT.[5IHT],
IHT.[6IHT], IHT.[7IHT], IHT.[8IHT], IHT.[9IHT], IHT.[10IHT], IHT.[11IHT], IHT.[12IHT]
FROM IHT
WHERE ((IHT.[PART NO]=[Forms]![PARTSCHEDULEFORM]![Field16]));
```

Query Parameters

Name	Type
Forms!PARTSCHEDULEFORM!Field16	Text

Columns

Name	Type	Size
PART NO	Text	50
1IHT	Number (Double)	8
2IHT	Number (Double)	8
3IHT	Number (Double)	8
4IHT	Number (Double)	8
5IHT	Number (Double)	8
6IHT	Number (Double)	8
7IHT	Number (Double)	8
8IHT	Number (Double)	8
9IHT	Number (Double)	8
10IHT	Number (Double)	8
11IHT	Number (Double)	8
12IHT	Number (Double)	8

Fig. 19. Query For Reporting Required part Schedules

Properties

Date Created:	29/4/95 11:53:51 AM	Def. Updatable:	Yes
Last Updated:	8/5/95 2:41:44 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW IHT.[PART NO], MACHINELOADANDSETUP.TEZGAH, SETUP.Setup,
MACHINELOADANDSETUP.OP, MACHINELOADANDSETUP.[1LOADSET],
MACHINELOADANDSETUP.[2LOADSET], MACHINELOADANDSETUP.[3LOADSET],
MACHINELOADANDSETUP.[4LOADSET], MACHINELOADANDSETUP.[5LOADSET],
MACHINELOADANDSETUP.[6LOADSET], MACHINELOADANDSETUP.[7LOADSET],
MACHINELOADANDSETUP.[8LOADSET], MACHINELOADANDSETUP.[9LOADSET],
MACHINELOADANDSETUP.[10LOADSET], MACHINELOADANDSETUP.[11LOADSET],
MACHINELOADANDSETUP.[12LOADSET]
FROM (((SETUP RIGHT JOIN MACHINELOADANDSETUP ON SETUP.[Part No] =
MACHINELOADANDSETUP.PARÇA) LEFT JOIN setupcostofop ON (MACHINELOADANDSETUP.TEZGAH =
setupcostofop.[M/C]) AND (MACHINELOADANDSETUP.OP = setupcostofop.Op) AND
(MACHINELOADANDSETUP.TEZGAH = SETUP.[M/C]) AND (MACHINELOADANDSETUP.OP = SETUP.Op)
AND (MACHINELOADANDSETUP.PARÇA = setupcostofop.[Part No])) LEFT JOIN IHT ON
MACHINELOADANDSETUP.PARÇA = IHT.[PART NO]) LEFT JOIN HOLDINGCOST ON
MACHINELOADANDSETUP.PARÇA = HOLDINGCOST.PART
WHERE ((IHT.[PART NO]=[Forms]![PARTDETAILFORM]![Field16]));
```

Query Parameters

Name	Type
Forms!PARTDETAILFORM!Fi eld16	Text

Columns

Name	Type	Size
PART NO	Text	50
TEZGAH	Text	255
Setup.Setup	Number (Double)	8
OP	Number (Double)	8
1LOADSET	Number (Double)	8
2LOADSET	Number (Double)	8
3LOADSET	Number (Double)	8
4LOADSET	Number (Double)	8
5LOADSET	Number (Double)	8
6LOADSET	Number (Double)	8
7LOADSET	Number (Double)	8
8LOADSET	Number (Double)	8
9LOADSET	Number (Double)	8
10LOADSET	Number (Double)	8
11LOADSET	Number (Double)	8
12LOADSET	Number (Double)	8

Fig.20. Query Structure Treport Loads on Required Machines

Properties

Date Created:	8/5/95 4:31:31 PM	Def. Updatable:	Yes
Last Updated:	8/5/95 6:04:31 PM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW machineload.TEZGAH, machineload.PARÇA, machineload.OP, setupcostofop.işç,
machineload.[1WKLOAD], OVERTIMECOST.OVERTIME1, OVERTIMECOST.OVERCOST1,
machineload.[2WKLOAD], OVERTIMECOST.OVERTIME2, OVERTIMECOST.OVERCOST2,
machineload.[3WKLOAD], OVERTIMECOST.OVERTIME3, OVERTIMECOST.OVERCOST3,
machineload.[4WKLOAD], OVERTIMECOST.OVERTIME4, OVERTIMECOST.OVERCOST4,
machineload.[5WKLOAD], OVERTIMECOST.OVERTIME5, OVERTIMECOST.OVERCOST5,
machineload.[6WKLOAD], OVERTIMECOST.OVERTIME6, OVERTIMECOST.OVERCOST6,
machineload.[7WKLOAD], OVERTIMECOST.OVERTIME7, OVERTIMECOST.OVERCOST7,
machineload.[8WKLOAD], OVERTIMECOST.OVERTIME8, OVERTIMECOST.OVERCOST8
FROM (OVERTIMECOST RIGHT JOIN machineload ON OVERTIMECOST.TEZGAH =
machineload.TEZGAH) LEFT JOIN setupcostofop ON (machineload.TEZGAH = setupcostofop.[M/C]) AND
(machineload.OP = setupcostofop.Op) AND (machineload.PARÇA = setupcostofop.[Part No])
WHERE ((machineload.TEZGAH=[MACHINE ID]));
```

Query Parameters

Name	Type
[MACHINE ID]	Text

Columns

Name	Type	Size
TEZGAH	Text	255
PARÇA	Text	255
OP	Number (Double)	8
işç	Number (Double)	8
1WKLOAD	Number (Double)	8
OVERTIME1	Number (Double)	8
OVERCOST1	Number (Double)	8
2WKLOAD	Number (Double)	8
OVERTIME2	Number (Double)	8
OVERCOST2	Number (Double)	8
3WKLOAD	Number (Double)	8
OVERTIME3	Number (Double)	8
OVERCOST3	Number (Double)	8
4WKLOAD	Number (Double)	8
OVERTIME4	Number (Double)	8
OVERCOST4	Number (Double)	8
5WKLOAD	Number (Double)	8
OVERTIME5	Number (Double)	8
OVERCOST5	Number (Double)	8
6WKLOAD	Number (Double)	8
OVERTIME6	Number (Double)	8
OVERCOST6	Number (Double)	8
7WKLOAD	Number (Double)	8

Fig. 21a. Query Structure For Overtime Necessary For Machines

OVERCOST7	Number (Double)	8
8WKLOAD	Number (Double)	8
OVERTIME8	Number (Double)	8
OVERCOST8	Number (Double)	8

Fig.21b. Query Structure For Overtime Necessary For Machines

Properties

Date Created:	9/5/95 11:04:06 AM	Def. Updatable:	Yes
Last Updated:	9/5/95 11:04:07 AM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW Sum(SETUPCOSTSOFSCHEDULE.[1SETUP]) AS SumOf1SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[2SETUP]) AS SumOf2SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[3SETUP]) AS SumOf3SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[4SETUP]) AS SumOf4SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[5SETUP]) AS SumOf5SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[6SETUP]) AS SumOf6SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[7SETUP]) AS SumOf7SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[8SETUP]) AS SumOf8SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[9SETUP]) AS SumOf9SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[10SETUP]) AS SumOf10SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[11SETUP]) AS SumOf11SETUP,
Sum(SETUPCOSTSOFSCHEDULE.[12SETUP]) AS SumOf12SETUP
FROM SETUPCOSTSOFSCHEDULE;
```

Columns

Name	Type	Size
SumOf1SETUP	Number (Double)	8
SumOf2SETUP	Number (Double)	8
SumOf3SETUP	Number (Double)	8
SumOf4SETUP	Number (Double)	8
SumOf5SETUP	Number (Double)	8
SumOf6SETUP	Number (Double)	8
SumOf7SETUP	Number (Double)	8
SumOf8SETUP	Number (Double)	8
SumOf9SETUP	Number (Double)	8
SumOf10SETUP	Number (Double)	8
SumOf11SETUP	Number (Double)	8
SumOf12SETUP	Number (Double)	8

Fig.22. Query Structure For Overall Setup Cost Of The Schedule

Properties

Date Created:	9/5/95 10:56:42 AM	Def. Updatable:	Yes
Last Updated:	9/5/95 10:56:42 AM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW Sum(HOLDINGCOSTOFSCHEDULE.[1INV]) AS SumOf1INV,
Sum(HOLDINGCOSTOFSCHEDULE.[2INV]) AS SumOf2INV, Sum(HOLDINGCOSTOFSCHEDULE.[3INV]) AS
SumOf3INV, Sum(HOLDINGCOSTOFSCHEDULE.[4INV]) AS SumOf4INV,
Sum(HOLDINGCOSTOFSCHEDULE.[5INV]) AS SumOf5INV, Sum(HOLDINGCOSTOFSCHEDULE.[6INV]) AS
SumOf6INV, Sum(HOLDINGCOSTOFSCHEDULE.[7INV]) AS SumOf7INV,
Sum(HOLDINGCOSTOFSCHEDULE.[8INV]) AS SumOf8INV, Sum(HOLDINGCOSTOFSCHEDULE.[9INV]) AS
SumOf9INV, Sum(HOLDINGCOSTOFSCHEDULE.[10INV]) AS SumOf10INV,
Sum(HOLDINGCOSTOFSCHEDULE.[11INV]) AS SumOf11INV, Sum(HOLDINGCOSTOFSCHEDULE.[12INV])
AS SumOf12INV
FROM HOLDINGCOSTOFSCHEDULE;
```

Query Parameters

Name	Type
INTERESTRATE	Text

Columns

Name	Type	Size
SumOf1INV	Number (Double)	8
SumOf2INV	Number (Double)	8
SumOf3INV	Number (Double)	8
SumOf4INV	Number (Double)	8
SumOf5INV	Number (Double)	8
SumOf6INV	Number (Double)	8
SumOf7INV	Number (Double)	8
SumOf8INV	Number (Double)	8
SumOf9INV	Number (Double)	8
SumOf10INV	Number (Double)	8
SumOf11INV	Number (Double)	8
SumOf12INV	Number (Double)	8

Fig.23. Query Structure For Overall Holding Cost

Properties

Date Created:	9/5/95 10:47:38 AM	Def. Updatable:	Yes
Last Updated:	9/5/95 10:51:56 AM	ODBC Timeout:	60
Record Locks:	No Locks	Returns Records:	Yes
Type:	Select		

SQL

```
SELECT DISTINCTROW Sum(OVERTIMECOST.OVERCOST1) AS SumOfOVERCOST1,
Sum(OVERTIMECOST.OVERCOST2) AS SumOfOVERCOST2, Sum(OVERTIMECOST.OVERCOST3) AS
SumOfOVERCOST3, Sum(OVERTIMECOST.OVERCOST4) AS SumOfOVERCOST4,
Sum(OVERTIMECOST.OVERCOST5) AS SumOfOVERCOST5, Sum(OVERTIMECOST.OVERCOST6) AS
SumOfOVERCOST6, Sum(OVERTIMECOST.OVERCOST7) AS SumOfOVERCOST7,
Sum(OVERTIMECOST.OVERCOST8) AS SumOfOVERCOST8, Sum(OVERTIMECOST.OVERCOST9) AS
SumOfOVERCOST9, Sum(OVERTIMECOST.OVERCOST10) AS SumOfOVERCOST10,
Sum(OVERTIMECOST.OVERCOST11) AS SumOfOVERCOST11, Sum(OVERTIMECOST.OVERCOST12) AS
SumOfOVERCOST12
FROM OVERTIMECOST;
```

Columns

Name	Type	Size
SumOfOVERCOST1	Number (Double)	8
SumOfOVERCOST2	Number (Double)	8
SumOfOVERCOST3	Number (Double)	8
SumOfOVERCOST4	Number (Double)	8
SumOfOVERCOST5	Number (Double)	8
SumOfOVERCOST6	Number (Double)	8
SumOfOVERCOST7	Number (Double)	8
SumOfOVERCOST8	Number (Double)	8
SumOfOVERCOST9	Number (Double)	8
SumOfOVERCOST10	Number (Double)	8
SumOfOVERCOST11	Number (Double)	8
SumOfOVERCOST12	Number (Double)	8

Fig.24. Query Structure For Overall Overtime Cost Of The Schedule

PART NO	1HT	2HT	3HT
126201 599781	0	0	0
126202 5111588	880	800	720
126213 4951407	840	748	720
126213 4993814	0	0	0
126213 5124803	0	0	0
126213 599113	880	800	720
126214 4951409	0	0	0
126214 5147787	0	0	0
126214 5147820	0	0	0
126214A4951410	80	104	0
126214A5147819	924	844	836
126224 5111589	0	0	0
126224 596244	40	52	0
126224 596417	924	844	836
126224A5111587	0	0	0
126224A596252	0	0	0
126224A599784	40	52	0
126243 563512	280	320	340
126243B5128070	0	0	0
126502 599933	280	320	340
126502A4970496	0	0	0
126502A5087105	280	320	340
126504 4970495	560	428	380
126504 5087106	40	52	0
126510 5085463	280	320	340
126510 5111596	0	0	0
126510 599937V	880	800	720
126513 599115	0	0	0
126514 4993816	0	0	0
126514 599116	560	428	380
126514A599120	840	748	720
126524 5085462	40	52	0
126524 599941	280	320	340
126524A5111595	840	748	720
126524A599942	40	52	0
126530 5087033	0	0	0
126530 5087034	0	0	0
126530 597148	0	0	0
126540 4976483	924	844	836
128104 581724	0	0	0
128104 598174	0	0	0
128104A581725	924	844	836
128121 4996273-1	40	52	0
128121 5147816	0	0	0
128121 5162671	924	844	836
128127 4991966	0	0	0
129004 598293	0	0	0
129005 588708	924	844	836
129006 593351-1	924	844	836
129007 5085116	0	0	0
129008 598294	0	0	0
129010 588709	1848	1688	1672
129011 588710	0	0	0
129011 593670	0	0	0
129015 4971673	0	0	0
129016 563438	80	104	0
129017 4971674	924	844	836
129017 5117500	0	0	0
129018 5109831	0	0	0
129018 588713	924	844	836

Fig.25a. Production Schedule of Former System

PART NO	1IHT	2IHT	3IHT	4IHT	5IHT	6IHT	7IHT	8IHT	9IHT	10IHT	11IHT	12IHT
126213 5124803	0	0	0	0	0	0	0	0	0	0	0	0
126213 599113	440	0	440	0	400	0	580	0	0	360	0	180
126214 4951409	0	0	0	0	0	0	0	0	0	0	0	0
126214 5147787	0	0	0	0	0	0	0	0	0	0	0	0
126214 5147820	0	0	0	0	0	0	0	0	0	0	0	0
126214A4951410	20	20	20	20	26	26	26	26	0	0	0	0
126214A5147819	231	231	231	231	211	211	211	211	209	209	209	209
126224 5111589	0	0	0	0	0	0	0	0	0	0	0	0
126224 596244	92	0	0	0	0	0	0	0	0	0	0	0
126224 596417	231	231	231	231	211	211	211	211	209	209	209	209
126224A5111587	0	0	0	0	0	0	0	0	0	0	0	0
126224A596252	0	0	0	0	0	0	0	0	0	0	0	0
126224A599784	92	0	0	0	0	0	0	0	0	0	0	0
126243 563512	520	0	0	0	0	0	0	420	0	0	0	0
126243B5128070	0	0	0	0	0	0	0	0	0	0	0	0
126502 599933	210	0	0	230	0	0	245	0	0	255	0	0
126502A4970496	0	0	0	0	0	0	0	0	0	0	0	0
126502A5087105	210	0	0	230	0	0	245	0	0	255	0	0
126504 4970495	140	140	140	140	107	107	107	107	95	95	95	95
126504 5087106	92	0	0	0	0	0	0	0	0	0	0	0
126510 5085463	70	70	70	70	80	80	80	80	85	85	85	85
126510 5111596	0	0	0	0	0	0	0	0	0	0	0	0
126510 599937V	220	220	220	220	200	200	200	200	180	180	180	180
126513 599115	0	0	0	0	0	0	0	0	0	0	0	0
126514 4993816	0	0	0	0	0	0	0	0	0	0	0	0
126514 599116	420	0	0	354	0	0	309	0	0	285	0	0
126514A599126	210	210	210	210	187	187	187	187	180	180	180	180
126524 5085462	66	0	0	0	0	0	26	0	0	0	0	0
126524 599941	280	0	0	0	320	0	0	0	340	0	0	0
126524A5111595	420	0	420	0	374	0	374	0	360	0	360	0
126524A599942	66	0	0	0	0	0	26	0	0	0	0	0
126530 5087033	0	0	0	0	0	0	0	0	0	0	0	0
126530 5087034	0	0	0	0	0	0	0	0	0	0	0	0
126530 597148	0	0	0	0	0	0	0	0	0	0	0	0
126540 4976483	1135	0	0	0	0	1051	0	0	0	0	418	0
128104 581724	0	0	0	0	0	0	0	0	0	0	0	0
128104 598174	0	0	0	0	0	0	0	0	0	0	0	0
128104A581725	231	231	231	231	211	211	211	211	209	209	209	209
128121 4996273-1	92	0	0	0	0	0	0	0	0	0	0	0
128121 5147816	0	0	0	0	0	0	0	0	0	0	0	0
128121 5162671	924	0	0	0	844	0	0	0	836	0	0	0
128127 4991966	0	0	0	0	0	0	0	0	0	0	0	0
129004 598293	0	0	0	0	0	0	0	0	0	0	0	0
129005 588708	693	0	0	653	0	0	631	0	0	627	0	0
129006 593351-1	693	0	0	653	0	0	631	0	0	627	0	0
129007 5085116	0	0	0	0	0	0	0	0	0	0	0	0
129008 598294	0	0	0	0	0	0	0	0	0	0	0	0
129010 588709	924	0	924	0	844	0	844	0	836	0	836	0
129011 588710	0	0	0	0	0	0	0	0	0	0	0	0
129011 593670	0	0	0	0	0	0	0	0	0	0	0	0
129015 4971673	0	0	0	0	0	0	0	0	0	0	0	0
129016 563438	132	0	0	0	0	0	52	0	0	0	0	0
129017 4971674	693	0	0	653	0	0	631	0	0	627	0	0
129017 5117500	0	0	0	0	0	0	0	0	0	0	0	0
129018 5109831	0	0	0	0	0	0	0	0	0	0	0	0
129018 588713	693	0	0	653	0	0	631	0	0	627	0	0
129019 4971675	0	0	0	0	0	0	0	0	0	0	0	0
129028 593675	462	0	462	0	422	0	422	0	418	0	418	0
129030 584785	0	0	0	0	0	0	0	0	0	0	0	0
129047 593693	0	0	0	0	0	0	0	0	0	0	0	0

fig.25b. Production Schedule For Silver-Meal

PART NO	1SETUP	2SETUP	3SETUP
126201 599781	0	0	0
126202 5111588	0	0	0
126213 4951407	2377246.14	2377246.1	2377246.14
126213 4993814	0	0	0
126213 5124803	0	0	0
126213 599113	2261954.658	2261954.7	2261954.658
126214 4951409	0	0	0
126214 5147787	0	0	0
126214 5147820	0	0	0
126214A4951410	0	0	0
126214A5147819	0	0	0
126224 5111589	0	0	0
126224 596244	1027070.3508	1027070.4	0
126224 596417	0	0	0
126224A5111587	0	0	0
126224A596252	0	0	0
126224A599784	206006.388	206006.39	0
126243 563512	6447584.544	6447584.5	6447584.544
126243B5129070	0	0	0
126502 599933	1549606.947	1549606.9	1549606.947
126502A4970496	0	0	0
126502A5087105	162742.6944	162742.69	162742.6944
126504 4970495	0	0	0
126504 5087106	203042.28	203042.28	0
126510 5085463	0	0	0
126510 5111596	0	0	0
126510 599937V			
126513 599115	0	0	0
126514 4993816	0	0	0
126514 599116	2314514.0827	2314514.1	2314514.0827
126514A599126	0	0	0
126524 5085462	206006.388	206006.39	0
126524 599941	1027070.3508	1027070.4	1027070.3508
126524A5111595	454640.394	454640.39	454640.394
126524A599942	206006.388	206006.39	0
126530 5087033	0	0	0
126530 5087034	0	0	0
126530 597148	0	0	0
126540 4976483	582523.248	582523.25	582523.248
128104 581724	0	0	0
128104 598174	0	0	0
128104A581725	0	0	0
128121 4996273-1	4154921.1192	4154921.1	0
128121 5147816	0	0	0
128121 5162671	1755111.048	1755111.0	1755111.048
128127 4991966	0	0	0
129004 598293	0	0	0
129005 588708	319479.612	319479.61	319479.612
129006 593351-1	3238788.444	3238788.4	3238788.444
129007 5085116	0	0	0
129008 598294	0	0	0
129010 588709	7295254.895	7295254.9	7295254.895
129011 588710	0	0	0
129011 593670	0	0	0
129015 4971673	0	0	0
129016 563438	625284.4752	625284.48	0
129017 4971674	2299439.9136	2299439.9	2299439.9136
129017 5117500	0	0	0
129018 5109831	0	0	0
129018 588713	1488550.2076	1488550.2	1488550.2076

Fig.26a. Setup Cost Of Former System

PART NO	1SETUP	2SETUP	3SETUP	4SETUP	5SETUP	6SETUP	7SETUP	8SETUP	9SETUP	10SETUP	11SETUP	12SETUP
126213 6124803	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126213 599113	2261954.658	0	2261954.658	0	2261954.658	0	2261954.658	0	0	2261954.658	0	2261954.65[illegible]
126214 4951409	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126214 5147787	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126214 5147820	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126214A4951410	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126214A5147819	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126224 5111589	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126224 598244	1027070.3508	0	0	0	0	0	0	0	0	0	0	[illegible]
126224 598417	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126224A6111587	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126224A598252	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126224A599784	206006.388	0	0	0	0	0	0	0	0	0	0	[illegible]
126243 583512	6447584.544	0	0	0	0	0	0	6447584.54	0	0	0	[illegible]
126243B5128070	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126502 599933	1549806.947	0	0	1549806.947	0	0	1549806.947	0	0	1549806.947	0	[illegible]
126502A4970498	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126502A5087105	162742.6944	0	0	162742.6944	0	0	162742.6944	0	0	162742.6944	0	[illegible]
126504 4970495	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126504 5007106	203042.28	0	0	0	0	0	0	0	0	0	0	[illegible]
126510 6085463	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126510 6111598	0	0	0	0	0	0	0	0	0	0	0	[illegible]
126510 699937V												
126513 599115	0	0	0	0	0	0	0	0	0	0	0	0
126514 4993816	0	0	0	0	0	0	0	0	0	0	0	0
126514 599116	2314514.0827	0	0	2314514.0827	0	0	2314514.0827	0	0	2314514.0827	0	0
126514A599128	0	0	0	0	0	0	0	0	0	0	0	0
126524 6085462	206006.388	0	0	0	0	0	206006.388	0	0	0	0	0
126524 599941	1027070.3508	0	0	0	1027070.3508	0	0	0	1027070.3508	0	0	0
126524A5111595	454640.394	0	454640.394	0	454640.394	0	454640.394	0	454640.394	0	454640.394	0
126524A599942	206006.388	0	0	0	0	0	206006.388	0	0	0	0	0
126530 6087033	0	0	0	0	0	0	0	0	0	0	0	0
126530 6087034	0	0	0	0	0	0	0	0	0	0	0	0
126530 597149	0	0	0	0	0	0	0	0	0	0	0	0
126540 4978483	582523.248	0	0	0	0	582523.248	0	0	0	0	582523.248	0
128104 581724	0	0	0	0	0	0	0	0	0	0	0	0
128104 598174	0	0	0	0	0	0	0	0	0	0	0	0
128104A581725	0	0	0	0	0	0	0	0	0	0	0	0
128121 4998273-1	4154921.1192	0	0	0	0	0	0	0	0	0	0	0
128121 5147816	0	0	0	0	0	0	0	0	0	0	0	0
128121 5162671	1755111.048	0	0	0	1755111.048	0	0	0	1755111.048	0	0	0
128127 4991966	0	0	0	0	0	0	0	0	0	0	0	0
129004 598293	0	0	0	0	0	0	0	0	0	0	0	0
129005 598708	319479.812	0	0	319479.812	0	0	319479.812	0	0	319479.812	0	0
129006 593359-1	3238788.444	0	0	3238788.444	0	0	3238788.444	0	0	3238788.444	0	0
129007 5085116	0	0	0	0	0	0	0	0	0	0	0	0
129008 598294	0	0	0	0	0	0	0	0	0	0	0	0
129010 598709	7295254.895	0	7295254.895	0	7295254.895	0	7295254.895	0	7295254.895	0	7295254.895	0
129011 598710	0	0	0	0	0	0	0	0	0	0	0	0
129011 593870	0	0	0	0	0	0	0	0	0	0	0	0
129015 4971673	0	0	0	0	0	0	0	0	0	0	0	0
129016 583438	625284.4752	0	0	0	0	0	625284.4752	0	0	0	0	0
129017 4971674	2299439.9136	0	0	2299439.9136	0	0	2299439.9136	0	0	2299439.9136	0	0
129017 5117500	0	0	0	0	0	0	0	0	0	0	0	0
129018 5109831	0	0	0	0	0	0	0	0	0	0	0	0
129019 598713	1488550.2076	0	0	1488550.2076	0	0	1488550.2076	0	0	1488550.2076	0	0
129019 4971675	0	0	0	0	0	0	0	0	0	0	0	0
129028 593675	2138691.812	0	2138691.812	0	2138691.812	0	2138691.812	0	2138691.812	0	2138691.812	0
129030 584785	0	0	0	0	0	0	0	0	0	0	0	0
129047 593693	0	0	0	0	0	0	0	0	0	0	0	0

Fig.26b. Setup Costs Of Silver-Meal Schedule

part no	holdcost	1 week	2 week	3 week	TOTAL COST
125006 5085417	0	0	0	0	0
125007 341438	136056.03	0	0	0	0
125007 5085419	119564.52	0	0	0	0
125007 596247	231055.58	0	0	0	0
125008 5085406	19708.71	0	0	0	0
125009 596251	5431.19	0	0	0	0
125011 5085413	41472.45	0	0	0	0
125012 5085414	2769.68	0	0	0	0
125015 5085407	57972.87	440	400	360	1,739,186
125015 596258	98627.39	0	0	0	0
125017 341437	507860.1	0	0	0	0
125017 4974474	596905.24	0	0	0	0
125017 5085420	253529.18	440	400	360	7,605,875
125018 4968736	94077.49	0	0	0	0
125018A5085403	115436.69	0	0	0	0
125019 5085411	202102.7	0	0	0	0
125022 4994299	0	440	400	360	0
125024 4974475	4366.24	440	400	360	130,987
125025 4968738	0	0	0	0	0
125027 341435	152942.19	0	0	0	0
125027 5085421	146162.62	0	0	0	0
125027 596248	222270.01	0	0	0	0
125031 4957587	0	880	800	720	0
125031 5086997	0	440	400	360	0
125031A5085416	0	0	0	0	0
125066 5085418	2524.27	0	0	0	0
125069 4968739	4466.01	0	0	0	0
125207 4967878	16469.02	140	160	170	193,511
125313 567425	58074.67	462	422	418	1,890,331
126036 596325	3858.75	0	0	0	0
126201 596322	0	20	26	0	0
126201 599781	0	0	0	0	0
126202 5111588	0	440	400	360	0
126213 4951407	204399.34	420	374	360	5,896,921
126213 4993814	167898.36	0	0	0	0
126213 5124803	159132.46	0	0	0	0
126213 599113	213347.6	440	400	360	6,400,428
126214 4951409	194977.85	0	0	0	0
126214 5147787	183570.22	0	0	0	0
126214 5147820	261260.51	0	0	0	0
126214A4951410	9168.42	40	52	0	21,087
126214A5147819	9168.42	462	422	416	298,432
126224 5111589	85621.94	0	0	0	0
126224 596244	85590.22	20	26	0	98,429
126224 596417	19168.78	462	422	418	623,944
126224A5111587	62271.85	0	0	0	0
126224A596252	45378.68	0	0	0	0
126224A599784	0	20	26	0	0
126243 563512	171472.51	140	160	170	2,014,802
126243B5128070	137231.84	0	0	0	0
126502 599933	274238.28	140	160	170	3,222,300
126502A4970496	22664.7	0	0	0	0
126502A5087105	22664.7	140	160	170	266,310
126504 4970495	2981.79	280	214	190	50,989
126504 5087106	5722.87	20	26	0	6,581
126510 5085463	0	140	160	170	0
126510 5111596	0	0	0	0	0
126510 599937V	0	440	400	360	0
126513 599115	212916.41	0	0	0	0
126514 4993816	151595.84	0	0	0	0

Fig.27a. Inventory Costs For Former System

part no	holdcost	1 week	2 week	3 week	4 week	5 week	6 week	7 week	8 week	9 week	10 week	11 week	12 week	TOTALCOST
125008 5085406	19708.71	0	0	0	0	0	0	0	0	0	0	0	0	0
125009 596251	5431.19	0	0	0	0	0	0	0	0	0	0	0	0	0
125011 5085413	41472.45	0	0	0	0	0	0	0	0	0	0	0	0	0
125012 5085414	2769.68	0	0	0	0	0	0	0	0	0	0	0	0	0
125015 5085407	57972.87	440	220	0	400	200	0	380	180	0	360	180	0	3,420,399
125015 596258	99627.39	0	0	0	0	0	0	0	0	0	0	0	0	0
125017 341437	597860.1	0	0	0	0	0	0	0	0	0	0	0	0	0
125017 4974474	596905.24	0	0	0	0	0	0	0	0	0	0	0	0	0
125017 5085420	253529.18	440	220	0	600	400	200	0	540	360	180	0	0	10,634,395
125018 4960736	94077.49	0	0	0	0	0	0	0	0	0	0	0	0	0
125018A5085403	115436.69	0	0	0	0	0	0	0	0	0	0	0	0	0
125019 5085411	202102.7	0	0	0	0	0	0	0	0	0	0	0	0	0
125022 4994299	0	0	0	0	0	0	0	0	0	0	0	0	0	0
125024 4974475	4366.24	0	0	0	0	0	0	0	0	0	0	0	0	0
125025 4968738	0	0	0	0	0	0	0	0	0	0	0	0	0	0
125027 341435	152942.19	0	0	0	0	0	0	0	0	0	0	0	0	0
125027 5085421	146162.62	0	0	0	0	0	0	0	0	0	0	0	0	0
125027 596248	222270.01	0	0	0	0	0	0	0	0	0	0	0	0	0
125031 4957587	0	0	0	0	0	0	0	0	0	0	0	0	0	0
125031 5086997	0	0	0	0	0	0	0	0	0	0	0	0	0	0
125031A5085416	0	0	0	0	0	0	0	0	0	0	0	0	0	0
125066 5085418	2524.27	0	0	0	0	0	0	0	0	0	0	0	0	0
125069 4968739	4466.01	0	0	0	0	0	0	0	0	0	0	0	0	0
125307 4967878	16469.03	370	300	230	160	80	0	420	340	255	170	85	0	992,259
125313 567425	58074.67	462	231	0	633	422	211	0	627	418	209	0	0	4,664,848
126036 596325	3858.75	0	0	0	0	0	0	0	0	0	0	0	0	0
126201 596322	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126201 599781	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126202 5111588	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126213 4951407	204399.34	210	0	397	187	0	374	187	0	360	180	0	0	9,683,419
126213 4993814	167898.36	0	0	0	0	0	0	0	0	0	0	0	0	0
126213 5124803	159132.46	0	0	0	0	0	0	0	0	0	0	0	0	0
126213 599113	213347.6	220	0	220	0	200	0	380	180	0	180	0	0	7,360,492
126214 4951409	194977.85	0	0	0	0	0	0	0	0	0	0	0	0	0
126214 5147787	183570.22	0	0	0	0	0	0	0	0	0	0	0	0	0
126214 5147820	261260.51	0	0	0	0	0	0	0	0	0	0	0	0	0
126214A4951410	9168.42	0	0	0	0	0	0	0	0	0	0	0	0	0
126214A5147819	9168.42	0	0	0	0	0	0	0	0	0	0	0	0	0
126224 5111589	85821.94	0	0	0	0	0	0	0	0	0	0	0	0	0
126224 596244	85590.22	82	72	62	52	39	26	13	0	0	0	0	0	740,355
126224 596417	19168.78	0	0	0	0	0	0	0	0	0	0	0	0	0
126224A5111507	62271.85	0	0	0	0	0	0	0	0	0	0	0	0	0
126224A596252	45378.68	0	0	0	0	0	0	0	0	0	0	0	0	0
126224A5997[illegible]4	0	82	72	62	52	39	26	13	0	0	0	0	0	0
126243 563512	171472.51	450	380	310	240	160	80	0	340	255	170	85	0	10,588,427
126243B5128070	137231.84	0	0	0	0	0	0	0	0	0	0	0	0	0
126502 599933	274238.28	140	70	0	160	80	0	165	85	0	170	85	0	6,547,439
126502A4970498	22664.7	0	0	0	0	0	0	0	0	0	0	0	0	0
126502A5087105	22664.7	140	70	0	160	80	0	165	85	0	170	85	0	541,120
126504 4970495	2981.79	0	0	0	0	0	0	0	0	0	0	0	0	0
126504 5087106	5722.87	82	72	62	52	39	26	13	0	0	0	0	0	49,503
126510 5085463	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126510 5111596	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126510 599937V	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126513 599115	212918.41	0	0	0	0	0	0	0	0	0	0	0	0	0
126514 4993816	161595.84	0	0	0	0	0	0	0	0	0	0	0	0	0
126514 599116	222438.33	280	140	0	214	107	0	202	95	0	190	95	0	7,357,148
126514A599126	9168.42	0	0	0	0	0	0	0	0	0	0	0	0	0
126524 5085462	44893.27	56	46	36	26	13	0	13	0	0	0	0	0	213,243
126524 599941	87504.25	210	140	70	0	240	160	80	0	255	170	85	0	3,084,525

Fig.27b. Inventory Costs Of Silver-Meal

1/5/95

TEZGAH	OVERCO	OVERTIM	OVERCO	OVERTI	OVERCO	OVERTI	OVERCO	OVERTI	OVERCO	OVERTI	OVERCO	OVERTI	OVERCO	OVERTI	OVERCO	OVERTIME	OVERCO	OVERTI	OVERCO	OVERTI	OVERCO	OVERTIM	OVERCO	OVERTI
00IDB	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
103ILB001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
103ILB003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
103ILB004	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
103ILB005	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
103ILB006	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
103ILB007	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
104ILB001	53806119	5542	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
109ILB001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
109ILB002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS002	46008245	2773	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS003	51172719	7478	0	0	0	0	0	0	6803212	994	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS004	85657873	6094	0	0	0	0	0	0	65063546	4629	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS005	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS006	68714057	5599	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS007	29942719	2221	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS008	7215360	469	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS009	11072068	720	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10IKS011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
112ILB001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
113ILB002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
113ILB003	29037522	2991	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKB009	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKB023	3521240	458	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKC010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKC019	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKD005	23206990	1929	0	0	20579694	1710	0	0	12275010	1020	0	0	14672248	1219	0	0	9747802	810	0	0	3747802	310	0	0
11IKD006	03409580	9155	0	0	03409580	9155	0	0	91096829	8065	0	0	91096829	8065	0	0	89844409	7954	0	0	89844409	7954	0	0
11IKD024	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKE025	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKF008	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKL004	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11IKL020	34310596	2933	0	0	0	0	0	0	5603200	479	0	0	0	0	0	0	7356774	629	0	0	0	0	0	0
11IKR022	0	0	0	0	0	0	5469229	593	0	0	0	0	0	0	721724	78	0	0	0	0	0	0	0	0
12IKB004	8760614	952	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12IKG006	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12IKL001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12IKR005	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12ILD002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13IKD001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13IKH002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17IKS001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17IKS002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17IKS003	57678771	4048	0	0	57678771	4048	0	0	49496997	3474	0	0	49496997	3474	0	0	47604008	3341	0	0	47604008	3341	0	0
17IKT005	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17IKT006	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17IKT008	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17IKT010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17IKT023	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1IKD003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20IKC002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20IKD001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20IKD003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23IKF002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23IKL003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23IKL004	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23IKM001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25IKL001	39291375	3312	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig. 29b. Overtime Cost For Silver-Meal

TEZGAH	SumOf1LOADSET	SumOf2LOADSET	SumOf3LOADSET
	0	0	0
00IDB	0	0	0
103ILB001	121.64	112.84	111.96
103ILB003	516.8	526	487.92
103ILB004	1274.12	1179.2	1052.08
103ILB005	1729.76	1586.56	1569.84
103ILB006	1727.36	1581.04	1428.24
103ILB007	332	171	380
104ILB001	7096.6	6773.16	5346.8
109ILB001	0	0	0
109ILB002	0	0	0
10IKS001	1106.44	1008.88	813.76
10IKS002	7464.24	6796.6	6295.16
10IKS003	12518.2	12406.84	12146.72
10IKS004	13885.16	11986.2	10934.64
10IKS005	1083.6	997.2	781.2
10IKS006	9103.48	8456.08	7808.52
10IKS007	6575.52	6279.04	6104.32
10IKS008	5814.24	5522.08	5449.2
10IKS009	6266.04	5947.6	5794.28
10IKS010	826	837.24	806.2
10IKS011	15414	14314	13391
112ILB001	0	0	0
113ILB002	510.16	470.84	448.44
113ILB003	5625.52	5402.56	4734.52
11IKB009	3478.28	3159.56	3161.32
11IKB023	8811.88	7683.88	7279.72
11IKC010	8867.44	8019.72	7751.96
11IKC019	6762.72	6192.32	5948.08
11IKD005	13081.2	11692	11242.8
11IKD006	27320.24	25140.08	24918.32
11IKD024	4506.44	5201.32	4614.68
11IKE025	0	0	0
11IKF008			0
11IKL004	7870	6220	5370
11IKL020	8512.68	8384.36	8346.68
11IKR022	5778.4	5264	4749.6
12IKB004	6709.12	6149	5697.28
12IKG006	0	0	0
12IKL001	5212.4	5450.72	5659
12IKR005	0	0	0
12ILD002	2190	2357.84	2228
13IKD001	4668.4	4393.08	4233.4
13IKH002	809.88	912.72	1084.12
17IKS001	3280.12	3822.16	3371.48
17IKS002	0	0	0
17IKS003	17546.4	16397.92	16132.2
17IKT005	805.6	2588.16	1079.84
17IKT006	2887.2	2580.84	2487.6
17IKT008	2826	2616.2	1811.8
17IKT010	4269.6	3859.2	3038.4
17IKT023	0	0	0
1IKD003			0
20IKC002	4858.32	4325	4050.68
20IKD001	6738.64	6235.12	6037.56
20IKD003	4078.8	3927.96	3768.88
23IKF002			0
23IKL003	2304	2496	2592
23IKL004	8122.8	8586.96	8896.4
23IKM001	1888	1782.2	1612

Fig.28a. Machineloads For Former System

9/5/95

MACHINELOADBYWEEK

TEZGAH	SumOf1LOADSET	SumOf2LOADSET	SumOf3LOADSET	SumOf4LOADSET	SumOf5LOADSET	SumOf6LOADSET	SumOf7LOADSET	SumOf8LOADSET	SumOf9LOADSET	SumOf10LOADSET	SumOf11LOADSET	SumOf12LOADSET
	0	0	0	0	0	0	0	0	0	0	0	0
00IDB	0	0	0	0	0	0	0	0	0	0	0	0
103ILB001	237.47	0	0	0	0	0	0	0	0	88.97	0	0
103ILB003	568.36	0	128.4	0	125.04	0	403.8	0	124.56	0	124.56	0
103ILB004	1360.57	0	0	127.1	467.76	704.95	142.64	0	443.12	0	181.54	97.72
103ILB005	1638.44	0	540.2	0	749.4	0	496.2	0	752.2	0	715.72	0
103ILB006	1469.14	0	401.6	147.3	369.6	999.4	523.92	0	266.4	0	525.14	149.91
103ILB007	902.5	0	0	0	0	0	0	0	0	309.5	0	0
104ILB001	10342.04	0	0	0	2348.94	0	1378.6	1757.66	1992	[illegible]	0	0
109ILB001	0	0	0	0	0	0	0	0	0	0	0	0
109ILB002	0	0	0	0	0	0	0	0	0	0	0	0
10IKS001	1822.34	0	0	0	0	668.82	0	0	0	0	228.96	148.96
10IKS002	7572.83	0	0	1701.04	1866.16	3075.96	1616.74	0	1849.04	2526.55	0	367.68
10IKS003	12278.07	0	1098	2258.64	5794.18	2037.12	2495.6	4038.58	3867.72	685.5	1692.6	1065.95
10IKS004	10894.24	0	3331.8	0	9429.03	848.4	2973.46	0	2864.4	3306.07	2864.4	175.2
10IKS005	1310.4	0	0	0	0	1288.8	0	0	0	0	0	262.8
10IKS006	10399.29	0	0	1704.9	2285.04	1191.16	4337.66	1178.22	2267.36	0	1123.2	166.24
10IKS007	7021.19	0	1551.12	0	2771.68	0	1420.72	618.45	2279.01	510	1107.68	0
10IKS008	5269.38	0	392.4	1550	1610.22	0	2055.23	1470.19	250.2	1212	372.2	0
10IKS009	5520.27	0	0	3673.07	1000	0	3067.34	460.18	1025	3056.56	0	150.48
10IKS010	825	0	0	0	637.24	0	0	0	666.2	0	0	0
10IKS011	4176	3711	3711	4266	3391	3391	3796	3391	3359	3359	3359	3359
112ILB001	0	0	0	0	0	0	0	0	0	0	0	0
113ILB002	735.37	0	0	65.9	0	59.42	72.56	417.39	0	0	43.04	25.76
113ILB003	7790.87	0	390.18	15.1	2068.68	55.91	1489.48	1027.24	1401.58	991.68	469.24	10.64
11IKB009	3319.46	0	0	1952.76	0	0	1722	1036.06	0	1680	54.88	0
11IKB023	5257.98	0	3149.4	2278.82	2219.28	621	3347.06	943.68	2739.2	1045.7	2139.2	495.18
11IKC010	4763.41	0	4733.24	361.95	2789.8	1539.6	3199.19	120	4215.76	0	2929.88	548
11IKC019	4572.29	0	1224	3167.09	1120	0	4655.43	0	0	4065.01	0	548
11IKD005	6728.4	0	6510	208.8	5820	0	6013.2	0	5610	139.6	5610	0
11IKD006	13955.12	0	13955.12	0	12665.04	0	12665.04	0	12754.16	0	12754.16	0
11IKD024	4610.4	0	500	237.6	4408.28	0	507.4	0	2030.76	420	277	1115
11IKE025	0	0	0	0	0	0	0	0	0	0	0	0
11IKF008		0	0	0	0	0	0	0	0	0	0	0
11IKL004	2807.5	1622.5	1622.5	2682.5	1272.5	1272.5	1957.5	2147.5	935	1910	935	935
11IKL020	7733.35	0	956	693.41	5279.04	1046.2	1640.87	0	5428.96	1477.89	988	0
11IKR022	4363.8	0	0	5392.6	0	0	0	4676.2	0	0	0	1277.4
12IKB004	5752.34	0	0	4020.68	795.72	0	3948.58	0	102	3594.32	493.26	0
12IKG006	0	0	0	0	0	0	0	0	0	0	0	0
12IKL001	4660.2	0	1079.2	0	2526.01	0	2127.20	[illegible]	2600	0	1165.2	0
12IKR005	0	0	0	0	0	0	0	0	[illegible]	0	0	0
12ILD002	2416.4	0	0	1095.68	515.2	0	0	1035.7	1064	0	0	256.8
13IKD001	3184.96	0	1529	697	1506.2	703.24	2212.6	0	1492	405	1924.68	0
13IKH002	1015.56	0	0	0	0	955.57	0	0	0	835.59	0	0
17IKS001	4398.48	0	0	0	0	2951.03	565.64	0	0	2558.61	0	0
17IKS002	0	0	0	0	0	0	0	0	0	0	0	0
17IKS003	8848.2	0	8848.2	0	8273.96	0	8273.96	0	8141.1	0	8141.1	0
17IKT005	805.6	0	0	0	3548	0	0	0	0	0	0	0
17IKT006	2197.9	0	0	2034.72	0	0	1934.92	0	0	1888.2	0	0
17IKT009	2477.9	0	0	1957.9	0	0	1604.26	0	0	1372.95	0	0
17IKT010	3232.2	0	0	[illegible]	0	0	2719.2	0	0	[illegible]	0	0
17IKT023	0	0	0	0	0	0	0	0	0	0	0	0
1IKD003		0	0	0	0	0		0	0	0	0	0
20IKC002	2577.01	610.5	1031.42	1720.8	1250.48	948.26	558	1655.5	681.6	559.68	505.2	935.25
20IKD001	4337.96	0	2780.4	515.22	2302.62	1021.88	2587.94	202.4	2085	1097.5	1471	590.5
20IKD003	3411.6	0	336.4	1729.92	1075.8	464	2098.88	0	1077.52	1314.72	506.5	0
23IKF002		0	0	0	0		0	0	0	0	0	0
						0	1640	1488	0	0		0

Fig.28b. Machine Load For Silver-Meal

TEZGAH	OVERCOST1	OVERTIME1	OVERCOST2	OVERTIME2	OVERCOST3
00IDB	0	0	0	0	0
103ILB001	0	0	0	0	0
103ILB003	0	0	0	0	0
103ILB004	0	0	0	0	0
103ILB005	0	0	0	0	0
103ILB006	0	0	0	0	0
103ILB007	0	0	0	0	0
10[illegible]ILB001	0	0	0	0	0
109ILB001	0	0	0	0	0
109ILB002	0	0	0	0	0
10IKS001	0	0	0	0	0
10IKS002	0	0	0	0	0
10IKS003	17232139	2518	16470098	2407	14690087
10IKS004	54608047	3885	27917126	1986	13136876
10IKS005	0	0	0	0	0
10IKS006	0	0	0	0	0
10IKS007	0	0	0	0	0
10IKS008	0	0	0	0	0
10IKS009	0	0	0	0	0
10IKS010	0	0	0	0	0
10IKS011	63195783	5414	50355857	4314	39581991
112ILB001	0	0	0	0	0
113ILB002	0	0	0	0	0
113ILB003	0	0	0	0	0
11IKB009	0	0	0	0	0
11IKB023	0	0	0	0	0
11IKC010	0	0	0	0	0
11IKC019	0	0	0	0	0
11IKD005	37080159	3081	20241733	1682	14956258
11IKD006	195636839	17320	171011337	15140	168500497
11IKD024	0	0	0	0	0
11IKE025	0	0	0	0	0
11IKF008	0	0	0	0	0
11IKL004	0	0	0	0	0
11IKL020	0	0	0	0	0
11IKR022	0	0	0	0	0
12IKB004	0	0	0	0	0
12IKG006	0	0	0	0	0
12IKL001	0	0	0	0	0
12IKR005	0	0	0	0	0
12ILD002	0	0	0	0	0
13IKD001	0	0	0	0	0
13IKH002	0	0	0	0	0
17IKS001	0	0	0	0	0
17IKS002	0	0	0	0	0
17IKS003	107521140	7546	91157592	6398	87371613
17IKT005	0	0	0	0	0
17IKT006	0	0	0	0	0
17IKT008	0	0	0	0	0
17IKT010	0	0	0	0	0
17IKT023	0	0	0	0	0
1IKD003	0	0	0	0	0
20IKC002	0	0	0	0	0
20IKD001	0	0	0	0	0
20IKD003	0	0	0	0	0
23IKF002	0	0	0	0	0
23IKL003	0	0	0	0	0
23IKL004	0	0	0	0	0
23IKM001	0	0	0	0	0
25IKL001	0	0	0	0	0

Fig. 29a. Overtime Cost for The Former System

VITA

Berkhan N. Esmer was born in 1971, Ankara. By the graduation of Ankara Atatürk Anadolu High School in1989, he joined Bilkent University Faculty of Engineering Department of Industrial Engineering where he got his BS degree in 1993. He is currently working at Türk Traktör Fabrikasý as Industrial Engineer in Production Planning Department.