

**PERANCANGAN SISTEM PENGENDALIAN LANTAI
PRODUKSI YANG TERINTEGRASI DAN
TERKOMPUTERISASI DENGAN *MICROSOSFT VISUAL
BASIC* DAN *MICROSOFT ACCESS***

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

**AQUINALDO PRIMA INDRA
0404070123**



**DEPARTEMEN TEKNIK INDUSTRI
FAKULTAS TEKNIK
UNIVERSITAS INDONESIA
DEPOK
JULI 2008**



LEMBAR PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Aquinaldo Prima Indra
NPM : 0404070123
Program Studi : Sarjana Reguler Departemen Teknik Industri
Judul Skripsi : Perancangan Sistem Pengendalian Lantai
Produksi yang Terintegrasi dan Terkomputerisasi
dengan *Microsoft Visual Basic* dan
Microsoft Access

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Teknik Industri Fakultas Teknik Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Ir. Amar Rachman, MEIM

Penguji : Ir. Boy Nurtjahyo, MSIE

Penguji : Ir. Isti Surjandari MA, MT, Ph.D

Ditetapkan di : Depok
Tanggal : 18 Juli 2008

RIWAYAT HIDUP PENULIS

Nama : Acquinaldo Prima Indra

Tempat, Tanggal Lahir : Batusangkar, 22 Februari 1986

Alamat : Jl. Panai, Sumanik – Batusangkar, Sumatera Barat

Pendidikan :

a.	SD	:	SDN Kompleks No 12 Sumanik (1992 – 1998)
b.	SLTP	:	SMPN 3 Salimpaung (1998 – 2001)
c.	SMU	:	SMAN 1 Padang Panjang (2001 – 2004)
d.	S-1	:	Departemen Teknik Industri, Fakultas Teknik Universitas Indonesia (2004 – 2008)

UCAPAN TERIMA KASIH

Segala puji dan syukur kepada Allah SWT atas Rahmat, Karunia, dan Hidayah-Nya sehingga skripsi ini dapat terselesaikan pada waktunya. Selain itu, penulis juga ingin mengucapkan terima kasih kepada pihak-pihak yang telah membantu dalam penyusunan skripsi ini. Adapun pihak-pihak itu adalah:

1. Papa, Mama, Rido, Sonya, Ari, Ayang, One, Om Yon, Tck Upik, Pa Zul, Nte Ita, Nte Lin dan Seluruh keluarga besar penulis atas doa, perhatian, semangat, dan berbagai dukungan lainnya yang diberikan kepada penulis.
2. Bapak Ir. Amar Rachman, MEIM. selaku dosen pembimbing skripsi untuk segala bantuan dan pengarahan kepada penulis.
3. Bapak Akhmad Hidayatno ST, MBT, selaku pembimbing akademis selama masa studi penulis.
4. Seluruh dosen pengajar Teknik Industri yang telah memberikan ilmu yang tak ternilai selama masa kuliah.
5. Rian, Arie, Heri dan Eko atas semua suka dan duka yang pernah kita lalui bersama.
6. Fahmi, Azis, Surip, Danu, Alpha, Mela, Ria, Arli, Dee, Romad, Anwar, Dewi, Oka, Dafid, Ape, Ucok, Susi, Willia, WE 7, Bronjay, Teratai Hitam atas persahabatannya, atas waktu kita bersama.
7. Seluruh sahabat TIUI angkatan 2004 yang senantiasa memberikan keceriaan dan kebersamaan yang tidak akan terlupakan selamanya.
8. Pak M, Iwan, Mbak Ana, Mbak Har, Mas Dodi, Mbak Fat, Mas Latif.
9. Dan semua pihak yang telah membantu hingga skripsi ini dapat terselesaikan.

Penulis menyadari bahwa skripsi ini jauh dari sempurna mengingat keterbatasan penulis. Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan. Penulis berharap bahwa skripsi ini dapat memberikan manfaat bagi semua pihak yang membacanya.

Depok, Juli 2008

Penulis

**LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Acquinaldo Prima Indra
NPM/NIP : 0404070123
Program Studi : Teknik Industri
Fakultas : Teknik
Jenis karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia Hak Bebas Royalti Non- Eksklusif (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul :

**Perancangan Sistem Pengendalian Lantai Produksi Yang Terkomputerisasi
Dan Terintegrasi Dengan *Microsoft Visual Basic* Dan *Microsoft Access***

Beserta perangkat yang ada (bila diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/format-kan, mengelolanya dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 18 Juli 2008
Yang menyatakan :

(Acquinaldo Prima Indra)

ABSTRAK

Aquinaldo Prima Indra
Program Sarjana Teknik Industri
Universitas Indonesia

Perancangan Sistem Pengendalian Lantai Produksi Yang Terkomputerisasi Dan Terintegrasi Dengan Microsoft Visual Basic Dan Microsoft Access.

Pengendalian lantai produksi merupakan bagian dari *sub sistem* MPC (*Manufacturing Planning and Control*) seperti Rencana Produksi dan MRP (*Material Resource Planning*) yang dipakai kegiatan produksinya, yang bertujuan untuk mengatur aliran material agar MPS (*Master Production Schedule*) terlaksana dengan seefisien mungkin. Tujuan dari pengendalian lantai produksi pada penelitian ini adalah menghitung penjadwalan produksi harian di lini produksi dengan memperhitungkan biaya listrik yang dikeluarkan oleh perusahaan dalam memenuhi permintaan dari konsumen.

Pada penelitian ini akan dirancang suatu sistem pengendalian lantai produksi untuk penjadwalan produksi harian di lini produksi menggunakan *Microsoft Visual Basic* dan *Microsoft Access* serta terintegrasi dengan penjadwalan induk produksi (*Master Production Scheduling*) dan perencanaan kapasitas sumberdaya (*Capacity Resources Planning*) sebagai bagian dari perencanaan dan pengendalian produksi.

Kata kunci : Penjadwalan produksi, pengendalian lantai produksi, *Microsoft Visual Basic* dan *Microsoft Access*

ABSTRACT

Acquinaldo Prima Indra
Program Sarjana Teknik Industri
Universitas Indonesia

Computerized System Design of Shop Floor Control and Integrated Using
Microsoft Visual Basic and Microsoft Access

Shop Floor control is a part from MPC (Manufacturing Planning and Control) sub system like production planning and MRP (Material of Resource Planning) that will be used in production activity, where to manage material flow to be MPS (Master of Production Schedule) done efficiently. This reserach will be conducted to design the production activity control to calculate scheduling of daily production in production line with considering the electrical cost that company spend to fullfill demand from consumer.

This research will be conducted to design a shop floor control system using Microsoft Visual Basic and Microsoft Access and also integrated witihin Master Production Scheduling and Capacity Resources Planning as production planning and controlling substance.

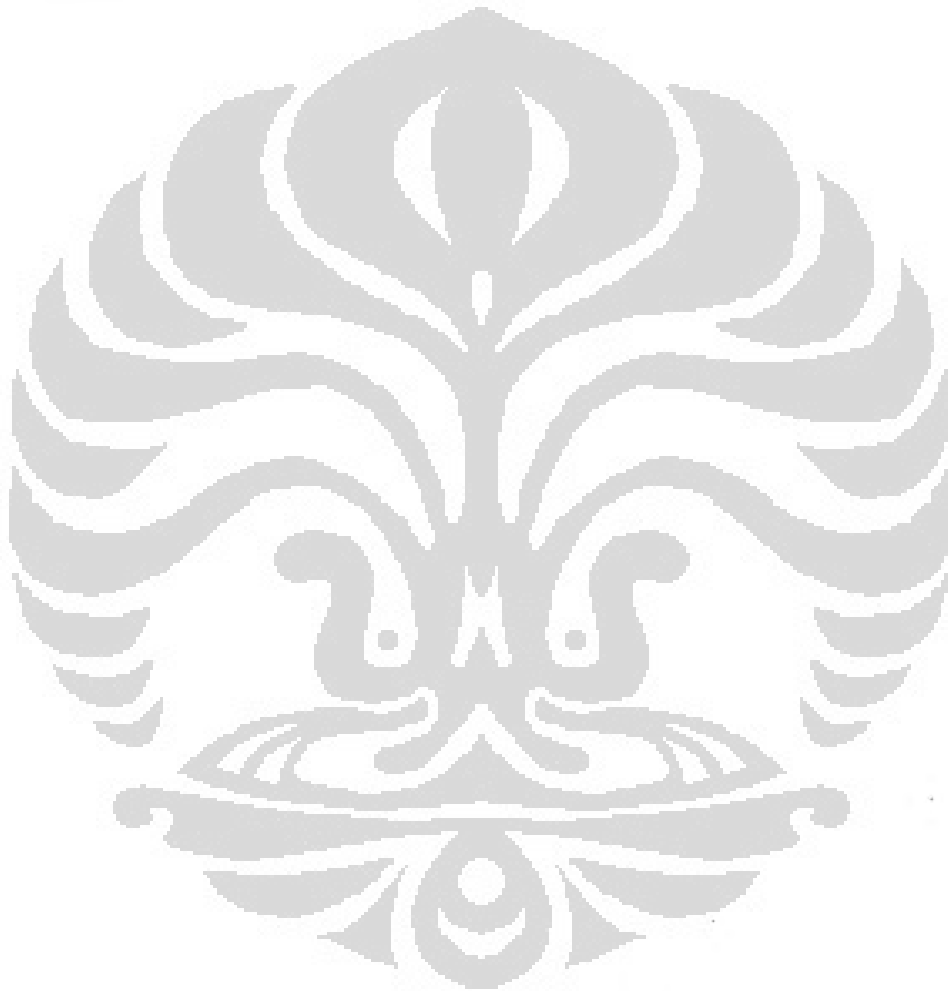
Keyword : Production scheduling, shop floor control, *Microsoft Visual Basic and Microsoft Access*.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
LEMBAR PENGESAHAN	iii
RIWAYAT HIDUP PENULIS.....	iv
UCAPAN TERIMAKASIH	v
LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS	vi
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xii
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN	xiv
1. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Diagram Keterkaitan Masalah	3
1.3. Perumusan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Ruang Lingkup Penelitian	5
1.6 Metodologi Penelitian.....	5
1.7 Sistematika Penelitian.....	7
2. TINJAUAN LITERATUR	8
2.1 Perencanaan Dan Pengendalian Produksi	8
2.2 Manajemen Sistem Produksi	10
2.3 Pengendalian Produksi.....	15
2.4 <i>Shop Floor Control</i> (Pengendalian Lantai Produksi)	16
2.4.1 Definisi <i>Shop Floor Control</i>	16
2.4.2 <i>Manufacturing Systems</i>	19
2.4.3 <i>Data Requirements</i>	20
2.4.4 <i>Order Preparation</i>	22
2.4.5 <i>Scheduling</i>	22
2.4.6 <i>Scheduling Techniques</i>	23
2.4.7 <i>Operation Sequencing</i>	24
2.4.8 <i>Production Reporting</i>	26
2.4.9 <i>Arsitektur Sistem Informasi Dalam Shop Floor Control</i>	27

2.4.9.1 Sistem Terpusat	27
2.4.9.2 Sistem Hierarkhi	27
2.4.9.3 Sistem Heterarkhi.....	28
2.4.10 Shoop Floor Control Integration	28
2.5 Siklus Produksi	29
2.5.1. Siklus Fabrikasi	29
2.5.2. Siklus Penjadwalan.....	29
2.6 Persoalan Umum Penjadwalan	32
2.6.1 Pengurutan (<i>sequencing</i>) produk.....	32
2.6.2 Penggiliran Kerja.....	33
3. PENGUMPULAN DAN PENGOLAHAN DATA	35
3.1 Pemahaman Bisnis PT.X	35
3.2 Konsep Tujuan.....	41
3.3 Identifikasi Data.....	42
3.4 Pengumpulan dan Hasil Data.....	42
3.4.1 Klasifikasi <i>Work Centre</i> di Masing-Masing Lini Produksi.....	42
3.4.2. Jumlah Mesin dan Pekerja di Setiap Work Centre.....	43
3.4.3 Jam Kerja Produksi	44
3.4.4 Waktu Proses, Kapasitas, dan Biaya	44
3.4.5 Jadwal induk Produksi (<i>Mater Production Scheduling</i>)	45
3.4.6 Penjadwalan Produksi Harian	46
4. PENGOLAHAN DATA DAN ANALISIS	48
4.1 PENYUSUNAN ALGORITMA	48
4.1.1 Langkah – Langkah Penyusunan Algoritma Sistem	52
4.1.1.1 <i>Pembuatan Draft Sistem</i>	52
4.1.2 Pengelompokkan Jenis Data.....	54
4.1.3 Pengidentifikasian dan Pemvisualisasian Interface Sistem	55
4.1.4 Memformulasikan Rumus	57
4.1.5 Penyusunan Sistem.....	60
4.2 Validasi dan Verifikasi Sistem	61
4.3 Cara Kerja Sistem	71
4.4 Analisis	75
4.4.1 Analisis Sistem Pengendalian Lantai Produksi	75
4.4.2 Analisis Rancangan Sistem Pengendalian Lantai Produksi	76
4.4.3 Analisis Penyusunan Sistem.....	77

4.4.4 Analisis Hasil	78
4.5 Kelebihan dan Kekurangan Sistem Pengendalian Lantai Produksi.....	78
4.5.1 Kelebihan Sistem.....	78
4.5.1.1 Kegunaan dan Fungsi Sistem.....	78
4.5.1.2 Rancangan Sistem.....	79
4.5.1.3 Aplikasi Sistem.....	79
4.5.2 Improvement Sistem.....	79
5. KESIMPULAN	80
DAFTAR REFERENSI	82



DAFTAR GAMBAR

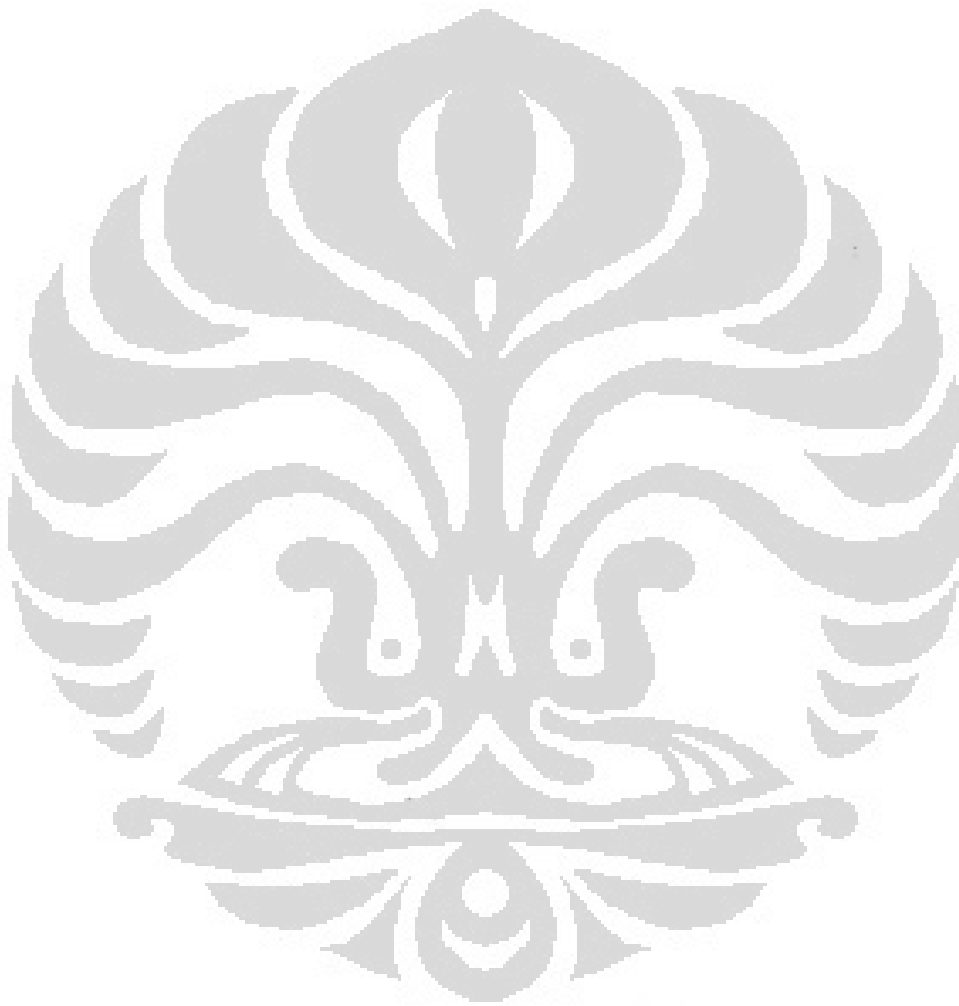
Gambar 1. 1 Diagram Keterkaitan Masalah	4
Gambar 1. 2 Diagram Alir Metodologi Penelitian.....	6
Gambar 2. 1 Skema Sistem Produksi.....	8
Gambar 2. 2 Hubungan antara Setiap Tahapan Manajemen Produksi	13
Gambar 2. 3 <i>Priority Planning</i> dan <i>Production Activity Control</i>	18
Gambar 2. 4 <i>Manufacturing Lead Time</i>	23
Gambar 2. 5 <i>Shoop Floor Control Integration</i>	28
Gambar 2. 6 Siklus Fabrikasi.....	29
Gambar 2. 7 Siklus Penjadwalan	30
Gambar 2. 7 Waktu kerja efektif dan tidak efektif pada manufaktur	31
Gambar 4. 1 Flowchart Hubungan dan interaksi antara ketiga sistem.....	50
Gambar 4. 2 Flowchart aktivitas utama pada sistem pengendalian rantai produksi	51
Gambar 4. 3 <i>Interface</i> untuk <i>Input General Data</i>	55
Gambar 4. 4 <i>Interface Input Data Perintah Produksi</i>	56
Gambar 4. 5 <i>Interface</i> untuk <i>General Information</i> dalam Sistem	56
Gambar 4. 6 <i>Interface</i> Pengolahan Data Penjadwalan Produksi Harian.....	57
Gambar 4. 7 Hasil setelah <i>Input General Data</i>	71
Gambar 4. 8 <i>Production Order</i> dari MPS.....	72
Gambar 4. 9 Pemrosesan Jadwal Produksi Harian	73
Gambar 4. 10 Hasil Akhir Penjadwalan Produksi Harian	74

DAFTAR TABEL

Tabel 2.1 Tabel Kebijakan Operasi dalam Sebuah Industri.....	17
Tabel 2.2 Hubungan Antara <i>Critical Ratio</i> dan <i>Prioritas Order</i>	27
Tabel 3.1 Produk-Produk PT.X	36
Tabel 3.2 Jumlah Mesin di Masing-Masing <i>Work Centre</i>	40
Tabel 3.3 Kondisi Kapasitas Masing-masing <i>Work Centre</i> PT X	41
Tabel 3.4 Klasifikasi <i>Work Centre</i> di Masing-Masing Lini Produksi	43
Tabel 3.5 Jumlah Mesin dan Pekerja di Setiap <i>Work Centre</i>	43
Tabel 3.6 Operasi Produksi Pekerja di Seluruh <i>Work Centre</i>	44
Tabel 3.7 Waktu Proses, Kapasitas, dan Biaya.....	44
Tabel 3.8 Jadwal induk Produksi (<i>Mater Production Scheduling</i>).....	45
Tabel 3.9 Jadwal harian produksi <i>work centre A</i>	47
Tabel 4.1 Jenis Data Input dan Sumber nya ..	53
Tabel 4.2 Jenis Data Informasi	53
Tabel 4.3 Pengelompokkan Jenis Data	54
Tabel 4.4 Perhitungan Penalti Produksi.....	59
Tabel 4.5 Data Umum <i>Work Centre</i> (Lembur).....	63
Tabel 4.6 Data Umum <i>Work Centre</i> (Normal)	63
Tabel 4.7 Data Informasi Tambahan (Lembur)	63
Tabel 4.8 Data Variabel Penalti	64
Tabel 4.9 Hasil penjadwalan dengan 400 kali random	64
Tabel 4.10 Hasil penjadwalan dengan 550 kali random	65
Tabel 4.11 Hasil Penjadwalan dengan 400 kali random (1)	66
Tabel 4.12 Hasil Penjadwalan dengan 400 kali random (2)	66
Tabel 4.13 Hasil Penjadwalan dengan 450 kali random (1)	67
Tabel 4.14 Hasil Penjadwalan dengan 450 kali random (2)	67
Tabel 4.15 Hasil Penjadwalan dengan 500 kali random (1)	68
Tabel 4.16 Hasil Penjadwalan dengan 500 kali random (2)	68
Tabel 4.17 Hasil Penjadwalan dengan 550 kali random (1)	69
Tabel 4.18 Hasil Penjadwalan dengan 550 kali random (2)	69
Tabel 4.19 Penjadwalan Produksi Harian yang Optimal	70

DAFTAR LAMPIRAN

LAMPIRAN 4.1	83
LAMPIRAN 4.2	92



merupakan salah satu sarana manajemen untuk mencapai tujuan yang telah ditetapkan, karena itu setiap tingkat manajemen dalam organisasi sangat membutuhkan aktivitas perencanaan. Perencanaan kapasitas merupakan salah satu hal yang memiliki peranan penting dalam dunia industri manufaktur maupun jasa.

Dalam implementasinya, perencanaan yang telah dibuat tersebut pada kondisi tertentu sangat mungkin asumsi-asumsi tersebut tidak dapat digunakan, Seperti beban produksi yang kelebihan kapasitas. Oleh karena itu perlu sebuah tindakan penyesuaian yang dikenal dengan istilah pengendalian. Pengendalian adalah sebuah tindakan penyesuaian rencana dan pelaksanaan, agar kegiatan operasional dan performansi sistem manufaktur tetap *acceptable*, meskipun perlu perubahan-perubahan dalam rencana¹.

Shop Floor Control merupakan aktivitas untuk melaksanakan segala rencana yang dibuat. Aktivitas ini merupakan sub sistem dari MPC (*Manufacturing Planning and Control*) seperti Rencana Produksi dan MRP (*Material Resource Planning*) yang dipakai saat terdapat proses transformasi dalam kegiatan produksinya, yang bertujuan untuk mengatur aliran material agar MPS (*Master Production Schedule*) terlaksana dengan mengutamakan efisiensi material, buruh tool, dan waktu mesin. Pembuatan rencana menggunakan beberapa asumsi: mesin selalu tersedia, material datang tepat waktu, waktu proses tertentu, tenaga kerja produktif, tidak ada perubahan jumlah demand dan due date, dan lain-lain

Aktivitas utama dari setiap industri manufaktur seperti industri farmasi adalah aktivitas-aktivitas fisik yang dilakukan di dalam *shop floor* (jalur produksi). Baik tidaknya suatu industri manufaktur sangat bergantung pada kinerja jalur produksinya. Oleh karena itu setiap industri manufaktur selalu memberi perhatian khusus terhadap perbaikan dan pengendalian jalur produksi agar seluruh aktivitas produksi dapat dilakukan dengan cara-cara yang lebih baik. Perencanaan dan pengendalian produksi merupakan suatu aktivitas membuat perencanaan, melaporkan hasil operasi, dan merevisi rencana yang dibutuhkan untuk mencapai hasil yang diinginkan. Perencanaan dan pengendalian produksi dilakukan untuk menentukan penugasan pekerjaan pada mesin, ekspektasi waktu menunggu pada

¹ Bedworth D.D., Bailey J.E., *Integrated Production Control System*, John Wiley & Sons, 1987.

mesin stasiun kerja, penjadwalan pekerjaan pada mesin dan penugasan tenaga kerja pada mesin. Kegiatan perencanaan dan pengendalian produksi meliputi perencanaan dan penjadwalan produksi, pembagian kerja, pengendalian proses, akuisisi data, analisis operasi, alokasi sumber daya, jaminan kualitas dan manajemen perawatan².

Pada penelitian ini akan dicari solusi untuk menyelesaikan masalah Pengendalian rantai produksi dengan menggunakan bahasa pemrograman *Visual Basic 6.0* yang akan membentuk logika dari sistem pengendalian rantai produksi yang akan dirancang dan *Microsoft Access* yang akan menyimpan setiap data yang dibutuhkan dan dihasilkan selama sistem ini berjalan.

1.2. Diagram Keterkaitan Masalah

Untuk melihat bagaimana keterkaitan antara setiap faktor-faktor yang berpengaruh pada permasalahan yang akan diteliti dan memberikan gambaran sistematis yang lebih menyeluruh, maka disusun suatu diagram keterkaitan permasalahan seperti terlihat pada gambar 1.1 berikut.

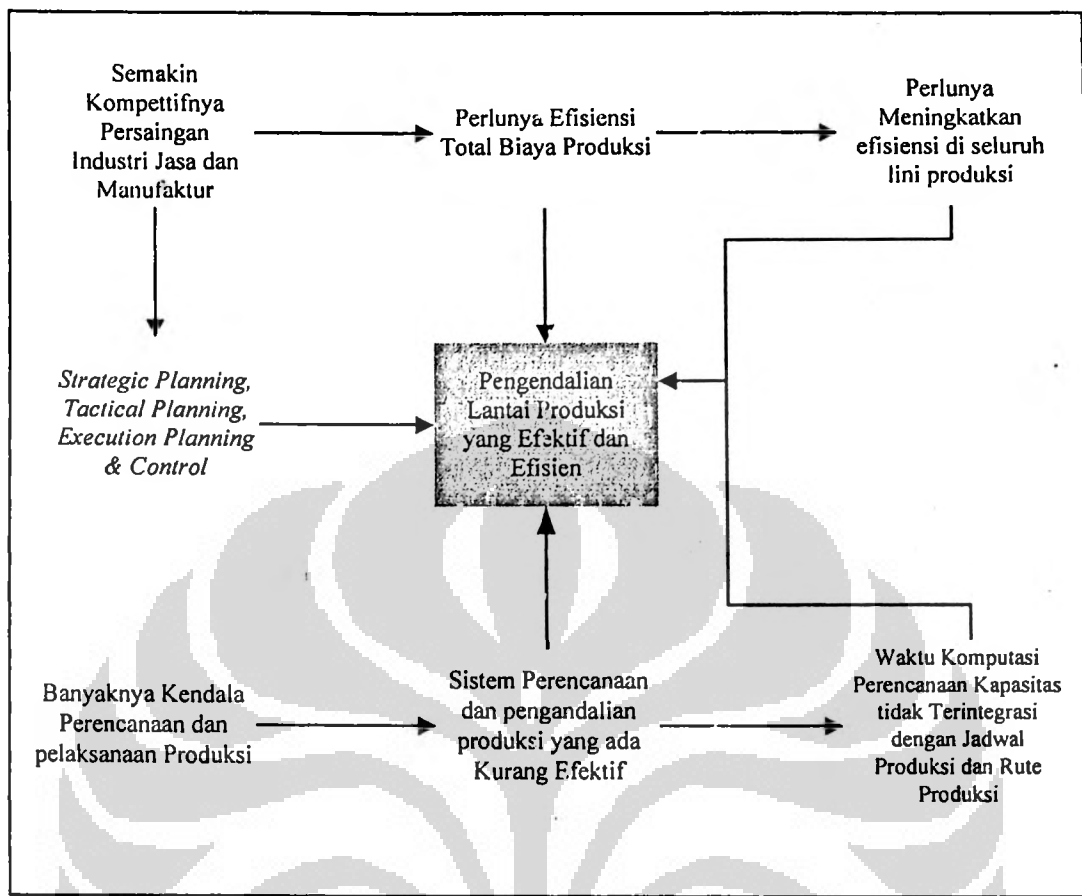
1.3. Perumusan Masalah

Berdasarkan latar belakang dan diagram keterkaitan masalah, maka pokok permasalahan yang akan dibahas dalam penelitian ini adalah belum adanya sebuah sistem yang mengintegrasikan antara MPS, *capacity resources planning* dan efisiensi setiap lini produksi dan merecord semua downtime dan kondisi aktual dari setiap lini produksi tersebut yang ada dalam industri farmasi.

1.4. Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah Memperoleh sebuah usulan perancangan sistem perencanaan dan pengendalian rantai produksi yang efektif dan efisien dengan mengoptimalkan *work flow* di setiap lini produksi melalui penerapan sistem yang terkomputerisasi.

² J.R. Tony Arnold, dan Stephen N. Chapman, *Introduction to Materials Management*, Pearson Education, Inc, 2004, hal 144



Gambar 1.1 Diagram Keterkaitan Masalah

1.5. Ruang Lingkup Penelitian

Untuk mendapatkan hasil penelitian yang spesifik dan terarah, maka ruang lingkup permasalahan dalam penelitian ini adalah sebagai berikut:

1. Aktivitas utama *dispatching*, antara lain *order sequencing*, *schedule maintenance*, *scheduling*, *down time*, dan utilisasi.
2. Penentuan urutan pekerjaan yang akan di proses di tiap mesin.
3. *Routing* / penugasan urutan tipe mesin yang diperlukan.
4. Data waktu *set up* mesin dan waktu proses produksi dianggap tetap.

1.6. Metodologi Penelitian

Berikut adalah metodologi penelitian yang dilakukan penulis dan digambarkan diagram alir pada Gambar 2

1. Pemilihan Topik Penelitian

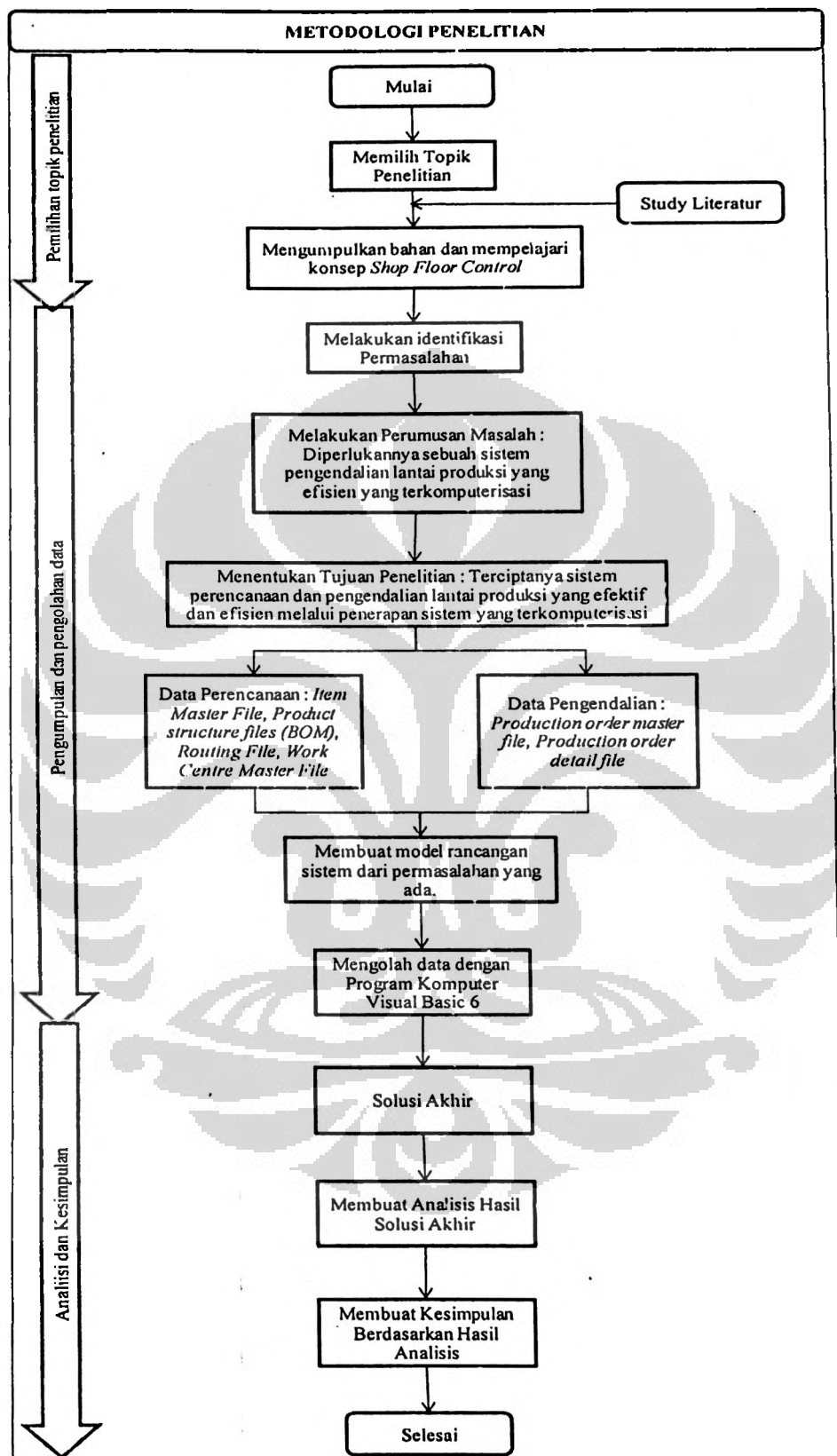
- a. Penentuan topik penelitian yang dibantu oleh dosen pembimbing penelitian.
- b. Mengumpulkan bahan dan mempelajari konsep *Shop Floor Control*, yang nantinya akan menjadi dasar teori penelitian.

2. Pengumpulan dan Pengolahan Data

- a. Melakukan identifikasi permasalahan dari topik yang akan diteliti.
- b. Melakukan perumusan masalah dari topik ini, yaitu: diperlukannya Sistem pengendalian jalur pengendalian produksi yang efisien yang terkomputerisasi dan terintegrasi dengan jadwal induk produksi dan perencanaan sumberdaya kapasitas.
- c. Menentukan tujuan penelitian dari perumusan permasalahan diatas, yaitu : Terciptanya sistem perencanaan dan pengendalian lantai produksi yang efektif dan efisien melalui penerapan sistem yang terkomputerisasi..
- d. Mengumpulkan data-data yang diperlukan, yaitu antara lain : Data Perencanaan : *Item Master File, Product structure files (BOM), Routing File, Work Centre Master File*, dan Data pengendalian : *Production order master file, Production order detail file*.
- e. Membuat model yang saling terintegrasi dan terkomputerisasi dari permasalahan yang ada.
- f. Mengolah data dengan program komputer Visual Basic 6.0 dari model program linear diatas.

3. Analisis dan Kesimpulan

- a. Membuat analisis hasil solusi akhir dengan tambahan saran dari dosen pembimbing penelitian
- b. Membuat kesimpulan berdasarkan hasil analisis dan saran diatas



Gambar 1. 2 Diagram Alir Metodologi Penelitian

1.7. Sistematika Penelitian

Pembahasan mengenai penelitian yang diketengahkan oleh peneliti disajikan dalam lima bab.

Bab 1 merupakan bab pendahuluan yang akan memberikan penjelasan mengenai latar belakang penelitian, diagram keterkaitan masalah, perumusan permasalahan, tujuan penelitian, batasan masalah, metodologi penelitian, dan sistematika penelitian.

Bab 2 menjelaskan secara terperinci mengenai teori dan konsep yang relevan dengan masalah yang telah dirumuskan untuk mencari pemecahan atas masalah tersebut. Adapun teori dan konsep yang relevan dengan masalah yang telah dirumuskan di atas antara lain adalah perencanaan produksi, sistem pengendalian rantai produksi, aktivitas *Dispatching, Sequencing dan routing/rerouting*.

Bab 3 membahas mengenai pengumpulan data. Data-data yang telah dikumpulkan merupakan data sekunder dari dokumen perusahaan adalah sebagai input dalam pengolahan data yang dilakukan pada tahap selanjutnya. Data yang diperlukan dalam penelitian ini antara lain, Data Perencanaan: *Item Master File, Product structure files (BOM), Routing File, Work Centre Master File* dan Data Pengendalian : *Production order master file, Production order detail file*.

Bab 4 merupakan pengolahan data dan analisis hasil yang diperoleh. Pada bab ini terdapat model matematis pengembangan program komputer Visual Basic 6.0 untuk model matematis tersebut beserta hasil eksekusi program dan analisis hasil program tersebut.

Bab 5 merupakan bab terakhir, dimana peneliti akan menyimpulkan secara keseluruhan dari uraian bab-bab sebelumnya.

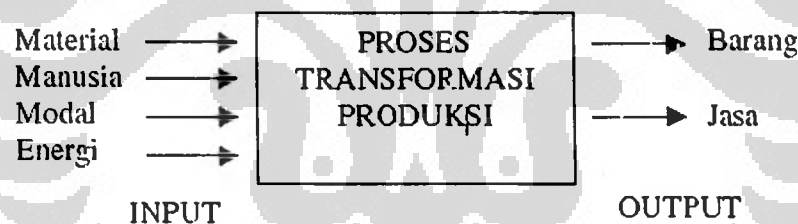
2. LANDASAN TEORI

2.1. Perencanaan Dan Pengendalian Produksi

Pada masa lalu pengertian produksi hanya dikaitkan dengan unit usaha fabrikasi yaitu yang menghasilkan barang – barang nyata seperti mobil, perabot, semen dan sebagainya. Namun pengertian produksi pada saat ini menjadi semakin meluas. Produksi dalam industri manufaktur sering diartikan sebagai aktivitas yang ditujukan untuk meningkatkan nilai masukan (*input*) menjadi keluaran (*output*). Dalam melakukan kegiatan produksi ada berbagai faktor (*input*) yang harus dikelola yang sering disebut sebagai faktor – faktor produksi yaitu:

1. Material atau bahan baku sebagai *input* utama dalam proses produksi
2. Mesin atau peralatan
3. Manusia atau karyawan
4. Modal atau uang
5. Manajemen yang akan memfungsionalisasikan keempat faktor yang lain.

Secara skematis sistem produksi³ dapat digambarkan sbb:



Gambar 2.1 Skema Sistem Produksi

Secara garis besar transformasi produksi dapat diklasifikasikan sebagai berikut;

1. Transformasi pabrikasi yaitu suatu transformasi yang bersifat diskrit dan menghasilkan produk nyata. Suatu transformasi dikatakan bersifat diskrit bila antara suatu operasi dan operasi yang lain dapat dibedakan dengan jelas seperti dijumpai pada pabrik mobil, misalnya.
2. Transformasi proses yaitu suatu transformasi yang bersifat kontinuedimana diantara operasi yang satu dengan operasi yang lain

³ M. P. Groover, *Fundamentals of Modern Manufacturing*, Prentice-Hall, 1996

3. kurang dapat dibedakan secara nyata, seperti dijumpai pada pabrik pupuk dan semen, misalnya.

Transformasi jasa yaitu suatu transformasi yang tidak mengubah secara fisik masukan menjadi keluaran; dalam hal ini secara fisik keluaran akan sama dengan masukan, namun transformasi jenis ini akan meningkatkan nilai masukannya, misalnya pada perusahaan angkutan. Sistem transformasi jasa sering disebut sebagai sistem operasi.

Dengan demikian maka kegiatan usaha jasa seperti dijumpai pada perusahaan angkutan, asuransi, bank, pos, telekomunikasi, dsb menjalankan juga kegiatan produksi. Secara skematis sistem produksi dapat digambarkan sebagai berikut:

Sekurang – kurangnya terdapat empat perbedaan pokok antara usaha jasa dan usaha pabrikasi, yaitu:

1. Dalam unit usaha pabrikasi keluarannya merupakan barang real sehingga produktivitasnya akan lebih mudah diukur bila dibandingkan dengan unit usaha jasa yang keluarannya berupa pelayanan
2. Kualitas produk yang dihasilkan dari usaha pabrikasi lebih mudah ditentukan standarnya
3. Kontak langsung dengan konsumen tidak selalu terjadi pada usaha pabrikasi sedangkan pada usaha jasa kontak langsung dengan konsumen merupakan suatu yang tidak dapat dielakkan
4. Tidak akan dijumpai adanya persediaan akhir di dalam usaha jasa sedang dalam usaha pabrikasi adanya persediaan sesuatu yang sulit dihindarkan.

Ditinjau dari kedatangan konsumen dan jumlah yang diminta, transformasi produksi dapat dibedakan atas:

1. *Job shop*, transformasi produksi bekerja bila ada pesanan saja. Jumlah pesanan relatif tidak terlalu besar dan jenis produk yang dipesan tidak standar sesuai dengan permintaan konsumen
2. *Flow shop*, transformasi produksi akan selalu bekerja baik ada pesanan maupun tidak. Jumlah pesanan biasanya relatif besar dan jenis produksinya standar. *Flow shop* dapat dibedakan atas :

- a. *Flow line / batch*
 - b. *Assembly line*
 - c. *Continuous*
3. *Project*, adalah bentuk spesial dari transformasi produksi dimana hanya ada satu atau beberapa pesanan yang spesifik dari konsumen.

2.2. Manajemen Sistem Produksi

Dalam melakukan kegiatan produksi, faktor (*input*) yang harus dikelola yang sering disebut sebagai faktor – faktor produksi seperti yang dijelaskan diatas harus dikelola dengan sebuah sistem yang aplikatif dan mudah dalam pelaksanaannya. Dengan demikian manajemen operasi berkaitan dengan pengelolaan faktor–faktor produksi sedemikian rupa sehingga keluaran (*output*) yang dihasilkan sesuai dengan permintaan konsumen baik kualitas, harga maupun waktu penyampaian.

Sebagaimana telah disebutkan di atas bahwa manajemen produksi operasi bertanggung jawab atas dihasilkannya keluaran (*output*) baik yang berupa produk maupun jasa yang sesuai dengan permintaan dan kebutuhan konsumen dengan kualitas yang baik dan harga yang terjangkau serta disampaikan tepat pada waktunya.

Bertitik tolak dari tanggung jawab ini maka ukuran kinerja suatu sistem operasi dapat diukur dari:

1. Biaya Produksi

Bila dikaitkan dengan tujuan suatu sistem usaha, maka ukuran kinerja sering diukur dengan keuntungan yang dapat dicapai, namun seperti diuraikan diatas bahwa sistem produksi hanyalah salah satu dari sub sistem yang ada dalam suatu sistem usaha, sehingga untuk mengukur seberapa besar kontribusi sistem operasi di dalam pencapaian keuntungan bukanlah hal yang mudah.

Oleh sebab itu untuk mengukur kinerja sistem produksi diambil ukuran waktu operasi tertentu, biasanya dalam waktu satu tahun, biaya produksi ini meliputi semua biaya yang dikeluarkan untuk menghasilkan produk/jasa ketangan konsumen. Dengan biaya produksi yang murah diharapkan bahwa produk / jasa dapat dipasarkan dengan harga yang dapat dijangkau oleh konsumen.

2. Kualitas Produk / Jasa.

Kenyataan menunjukkan bahwa konsumen tidak hanya memilih produk/jasa yang harganya murah namun juga produk/jasa yang berkualitas, oleh sebab itu baik buruknya suatu sistem produksi juga diukur dari kualitas produk/jasa yang dihasilkan. Ukuran kualitas produk yang dimaksudkan disini tentunya yang disesuaikan dengan selera konsumen bukan ukuran kualitas secara teknologi semata

3. Tingkat Pelayanan

Bagi konsumen untuk menilai baik buruknya suatu sistem produksi / operasi lebih dinilai dari pelayanan yang dapat diberikan oleh sistem produksi kepada konsumen itu sendiri. Berbicara mengenai tingkat pelayanan (*service level*) merupakan ukuran yang tidak mudah untuk diukur, sebab banyak dipengaruhi oleh faktor – faktor kualitatif, walaupun demikian beberapa ukuran obyektif yang sering digunakan antara lain:

1. Ketersediaan (*availability*) dan kemudahan untuk mendapatkan produk / jasa

Ukuran ketersediaan (*availability*) dan kemudahan untuk mendapatkan produk / jasa merupakan indikator yang secara langsung dapat menunjukkan tingkat pelayanan perusahaan kepada konsumennya. Hal ini dapat diukur dengan seberapa besar ketersediaan produk/jasa yang ditawarkan perusahaan di pasar. Selain itu, kemudahan mendapatkan produk/jasa tersebut merupakan faktor pendukung yang menunjukkan eksistensi perusahaan tersebut di pasar.

2. Kecepatan pelayanan baik yang berkaitan dengan waktu pengiriman (*delivery time*) maupun waktu pemrosesan (*processing time*)

Ukuran kecepatan pelayanan baik yang berkaitan dengan waktu pengiriman (*delivery time*) umumnya digunakan dalam industri jasa dimana perusahaan harus mampu melakukan pengiriman jasa yang ditawarkan kepada konsumen secara optimal sesuai dengan permintaan konsumen. Sedangkan ukuran kecepatan pelayanan baik yang berkaitan dengan waktu pemrosesan (*processing time*) sangat berkaitan dengan perusahaan manufaktur, dimana indikator ini sangat berperan penting dalam meningkatkan faktor ketersediaan (*availability*) dan kemudahan untuk mendapatkan produk mereka di perusahaan. Selain itu, *processing time*

sangat berpengaruh terhadap struktur biaya produksi perusahaan yang idealnya harus seminimal mungkin.

Pengelolaan sistem produksi (manajemen produksi) akan melibatkan serangkaian proses pengambilan keputusan operasional, keputusan-keputusan taktikal bahkan keputusan strategis. Dimana secara garis besar kegiatan dalam pengelolaan sistem produksi ini dapat dibagi menjadi tiga tahapan utama, antara lain :

1. Perencanaan manajemen atas

Pada tahapan ini, manajemen atas melaksanakan beberapa kegiatan yang berkaitan dengan peramalan permintaan, perencanaan kebutuhan, perencanaan sumber daya, distribusi dan penerimaan order.

2. Perencanaan manajemen operasi

Pada tahapan yang kedua ini, aktifitas yang dilakukan berkaitan dengan operasi perusahaan, antara lain perencanaan kapasitas, penjadwalan induk produksi, perencanaan kebutuhan material dan penjadwalan perakitan akhir.

3. Eksekusi manajemen operasi

Tahapan ini merupakan eksekusi dari perencanaan-perencanaan yang telah dilakukan diatas. Adapun aktivitas yang dilaksanakan pada tahapan ini antara lain pengontrolan aktivitas produksi dan segala aktivitas yang berkaitan dengan pengadaan bahan baku.

Gambar 2.2 Berikut ini merupakan aliran dan hubungan dari tahapan-tahapan dalam aktivitas produksi⁴ dalam sebuah perusahaan. Dimana Dalam setiap tahapan dalam manajemen produksi diatas, untuk menjalankan setiap aktivitas tersebut, maka masing-masing tahapan harus mengeluarkan beberapa keputusan-keputusan produksi. Secara umum ada lima jenis kategori keputusan penting yang harus diambil dalam manajemen produksi, yaitu keputusan yang berkaitan dengan beberapa hal berikut :

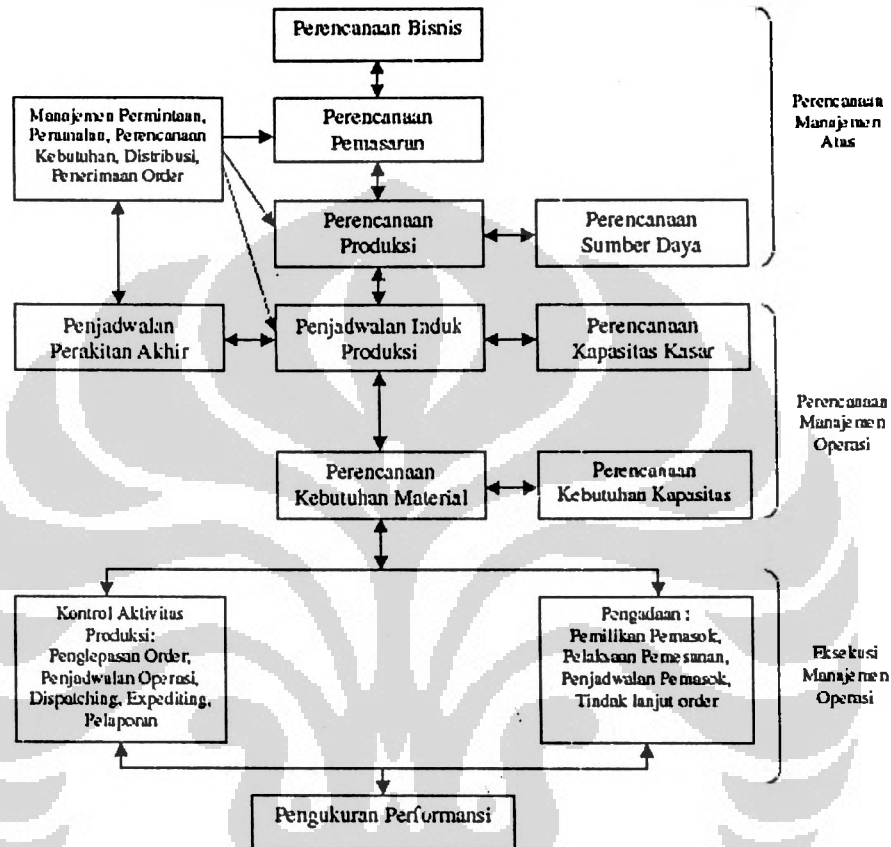
1. Proses produksi
2. Kapasitas
3. Persediaan

⁴ Amstead, B.H., Ostwald, Philip F., Begemen, Myrm, L. *Manufacturing Processes*. John Wiley and Sons, 1987

4. Tenaga kerja

5. Kualitas produksi

SISTEM PERENCANAAN DAN PENGENDALIAN



Gambar 2.2 Hubungan antara Setiap Tahapan Manajemen Produksi

1. Proses Produksi

Keputusan-keputusan yang termasuk dalam kategori ini pada prinsipnya berkaitan dengan penentuan wahana atau fasilitas fisik yang dipergunakan untuk terjadinya transformasi input menjadi produk / jasa. Keputusan tersebut antara lain:

- a. Teknologi produksi
- b. Tipe peralatan
- c. Jenis proses dan aliran proses produksi
- d. Tata letak fasilitas

Pada umumnya keputusan – keputusan yang diambil dalam kategori ini dilakukan pada tahap awal pendirian perusahaan (*initial business*) serta berdampak jangka

panjang dan tidak mudah diubah dalam waktu yang singkat (*long term strategic decision*).

2. Kapasitas

Keputusan - keputusan yang termasuk dalam kategori ini berkaitan dengan penentuan kemampuan sistem produksi untuk menghasilkan barang dalam jumlah dan waktu yang tepat (efisien). Dipandang dari sudut waktu dibedakan atas:

- a. Keputusan jangka panjang, antara lain penentuan kapasitas desain sistem produksi, ekspansi kapasitas, integrasi vertikal, integrasi horizontal, dan sebagainya.
- b. Keputusan jangka menengah, antara lain penentuan sub kontrak, penambahan mesin, rekrutasi tenaga kerja, dan sebagainya.
- c. Keputusan jangka pendek, pada prinsipnya berkaitan dengan pengalokasian pendayagunaan sumber – sumber yang tersedia untuk menghasilkan barang yang diminta konsumen. Keputusan ini diantaranya adalah penjadualan produksi (*scheduling & dispatching*), pengaturan mesin, dan sebagainya.

3. Persediaan (*Inventory*)

Keputusan yang termasuk dalam kategori ini pada hakekatnya berkaitan dengan pengaturan material yang diperlukan untuk keperluan produksi, mulai dari pengaturan bahan baku, barang setengah jadi maupun produk jadi. Ditinjau dari segi permasalahan yang dihadapi, keputusan ini dapat dibedakan atas keputusan tentang *operating system* persediaan dan keputusan tentang persediaan.

4. Tenaga Kerja

Mengelola orang merupakan pekerjaan terpenting yang perlu dibuat oleh seorang manajer mengingat tenaga kerja tidak hanya sebagai salah satu faktor produksi tetapi merupakan faktor penentu dari keberhasilan semua aktivitas didalam sistem produksi. Keputusan dalam kategori ini dimulai sejak proses seleksi karyawan sampai dengan pensiun. Adapun keputusan – keputusan rutin diantaranya penugasan karyawan, pengaturan lembur dan cuti, penggiliran kerja dan sebagainya.

5. Kualitas Produksi

Manajer produksi bertanggungjawab atas kualitas dari barang / jasa yang dihasilkan, oleh sebab itu manajer produksi wajib untuk melakukan kegiatan – kegiatan agar produk / jasa yang dihasilkan sesuai dengan standar yang telah ditetapkan. Kualitas hasil produksi tentunya memiliki *critical impact* yang cukup besar terhadap eksistensi produk/jasa yang dihasilkan perusahaan. Saat ini setiap perusahaan umumnya wajib memiliki suatu departemen yang secara khusus melakukan inspeksi terhadap kualitas produksi, sehingga kualitas produk yang dihasilkan dapat dipertanggungjawabkan dan bila perlu dilakukan *continues improvement* untuk meningkatkan kualitas produksi tersebut.

2.3. Pengendalian Produksi

Kebijakan operasi menyatakan tujuan operasi yang telah ditetapkan akan dapat dicapai. Kebijakan operasi ini harus dibuat untuk setiap kategori keputusan yang telah disebutkan terdahulu (proses, kapasitas, persediaan, tenaga kerja dan kualitas). Dengan demikian akan dapat dijumpai beberapa kebijaksanaan dalam suatu sistem produksi, tidak jarang bahwa kebijakan tersebut tidak selalu selaras bahkan saling bertentangan. Oleh karena itu penentuan kebijaksanaan operasi merupakan *trade off* dari berbagai pilihan yang ada dengan berpegang pada tujuan yang telah dinyatakan. Tabel⁵ berikut ini merupakan contoh dari suatu kebijaksanaan operasi.

Tabel 2.1 Tabel Kebijakan Operasi dalam Sebuah Industri

Policy Type	Policy Area	Strategic Choice
Process	Span of process Automation	Make or Buy Handmade or Machine made Flexible or hand automation Project batch line or continuous
Capacity	Facility size Location Investment	One large or several small facilities Near markets low cost or foreign Permanent or temporary
Inventory	Amount Distribution Control System	High levels or low levels of inventory Centralized or decentralized warehouse Control on great detail or less detail
Work House	Job Specialization Supervision Wage System Staffing	High or low specialization Highly decentralized or centralized type of pay incentives used Good paying or low paying Many or few staff
Quality	Approach Training Suppliers	Prevention or inspection Technical or managerial training Selected on quality or cost

⁵ Thomas M. Carol and Robert D. Dean “A Bayesian Approach to Plant – Location Decision “ Sciences.2004

2.4. Shop Floor Control (Pengendalian Lantai Produksi)

Shop Floor Control merupakan sebuah aktifitas untuk melaksanakan segala rencana yang telah dibuat. Aktivitas ini merupakan sub sistem dari *Manufacturing Planning and Control* (MPC) seperti Rencana Produksi dan *Material Resources Planning* (MRP) yang dipakai saat terdapat proses transformasi dalam kegiatan produksinya, yang bertujuan untuk mengatur aliran material agar *Master Production Schedule* (MPS) terlaksana dengan mengutamakan efisiensi material, buruh, mesin dan peralatan, dan waktu operasi mesin. Pembuatan rencana menggunakan beberapa asumsi: mesin selalu tersedia, material datang tepat waktu, waktu proses tertentu, tenaga kerja produktif, tidak ada perubahan jumlah *demand* dan *due date*, dan lain-lain

Dalam implementasinya, perencanaan yang telah dibuat tersebut pada kondisi tertentu sangat mungkin asumsi-asumsi tersebut tidak dapat digunakan, Seperti beban produksi yang kelebihan kapasitas. Oleh karena itu perlu sebuah tindakan penyesuaian yang dikenal dengan istilah pengendalian.

Pengendalian adalah sebuah tindakan penyesuaian rencana dan pelaksanaan, agar kegiatan operasional dan performansi sistem manufaktur tetap *acceptable*, meskipun perlu perubahan-perubahan dalam rencana.

2.4.1. Definisi Shop Floor Control

Dalam dunia industri manufaktur *Shop Floor Control* sering disebut juga sebagai *job shop control*, *Manufacturing activity control*, dan *production activity control*. Namun secara umum definisi⁶ *Shop floor Control* adalah aktifitas yang bertanggung jawab untuk melaksanakan rencana tahap sebelumnya dengan membuat jadual yang lebih rinci (dalam jangka pendek), memonitor dan melapor.

Dalam arti luas *Shop Floor Control* merupakan aktifitas merencanakan (jangka pendek), memonitor, mengendalikan, menekan *lead time*, antrian, dan penyesuaian. Dan dalam arti sebuah tindakan SFC merupakan aktivitas *dispatching* dan *reporting*.

⁶ J.R. Tony Arnold, dan Stephen N. Chapman, *Introduction to Materials Management*, Pearson Education, Inc, 2004, Hal 144

Shop Floor Control (SFC) merupakan bagian yang bertanggung jawab terhadap pelaksanaan *master production schedule (MPS)* dan *material requirements plan (MRP)*. Pada saat yang bersamaan, *Shop Floor Control (SFC)* juga bertanggung jawab terhadap utilisasi tenaga kerja dan mesin, meminimalisasi inventori *work in process*, dan tetap meningkatkan *customer service*. Kegiatan utama *Shop Floor Control (SFC)* adalah sebagai berikut:

- Mengeluarkan *work order (WO)* ke shop floor untuk pelaksanaan produksi.
- Melakukan kontrol terhadap *work order (WO)* dan memastikan pekerjaan diselesaikan.
- Menangani rencana detail *flow of orders* pada kegiatan produksi sesuai rencana, serta mengendalikan pekerjaan dalam proses sampai selesai.
- Mengelola kegiatan dari waktu ke waktu dan memberikan dukungan bila diperlukan.

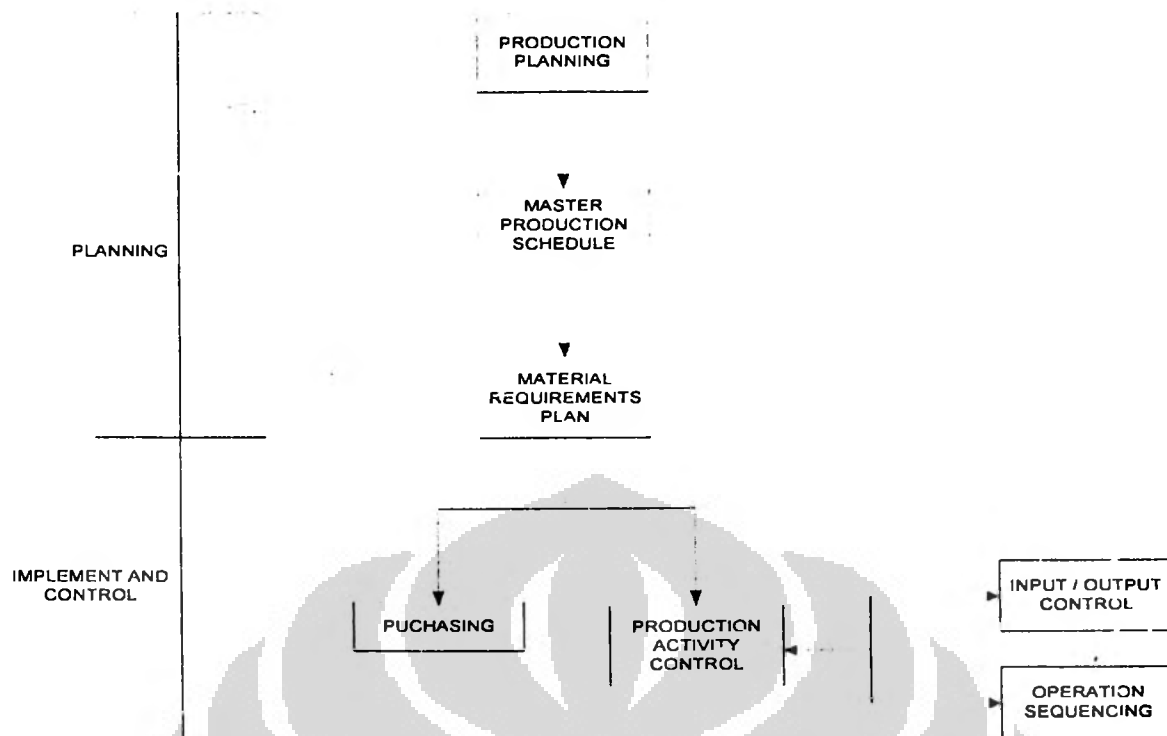
Shop Floor Control (SFC) memiliki enam fungsi utama, yaitu sebagai berikut :

1. Memiliki prioritas untuk tiap *shop order*.
2. Mendapatkan informasi kuantitas *work in process (WIP)*
3. Menyediakan informasi status *shop order* untuk kebutuhan informasi perusahaan.
4. Menyediakan data *output* (aktual) untuk mengendalikan kapasitas
5. Menyediakan *quantity by location by shop order* untuk mengetahui *WIP inventory*
6. Menyediakan ukuran efisiensi, utilisasi, dan produktivitas pekerja dan mesin.

Gambar⁷ 2.3 berikut adalah menunjukkan klasifikasi dari *Shop Floor Control (SFC)*. Dimana kegiatan *Shop Floor Control (SFC)* dapat diklasifikasikan menjadi tiga bagian yaitu:

1. Perencanaan / *Planning*
2. *Implementation*
3. Pengendalian / *Control*

⁷ J.R. Tony Arnold, dan Stephen N. Chapman, *Introduction to Materials Management*, Pearson Education, Inc, 2004, hal 145



Gambar 2.3 *Priority Planning dan Production Activity Control*

1. *Planning*

Pada bagian ini, *Shop Floor Control (SFC)* mesti melakukan beberapa hal antara lain memastikan bahwa material, personel, peralatan, serta informasi yang dibutuhkan tersedia pada saat diperlukan dan menetapkan waktu pelaksanaan dan proses kerja di setiap *work center*.

2. *Implementation*

Setelah perencanaan dibuat, *Shop Floor Control (SFC)* harus memberikan instruksi kepada *shop floor* apa yang harus dikerjakan berkaitan dengan rencana tersebut. Instruksi tersebut dapat berupa penyerahan perintah order kepada *shop floor* dalam bentuk *material requirements plan (MRP)*.

3. *Control*

Poduction activity control (PAC) dalam hal ini perlu melakukan kontrol terhadap apa yang terjadi di lapangan, seperti *work center efficiency*, *operation times*, *order quantities*, *lead times*, dan *work center queues*.

2.4.2. Manufacturing Systems

Proses manufaktur⁸ pada suatu perusahaan dapat dibagi dalam tiga kategori, yaitu sebagai berikut:

1. *Flow Manufacturing*

Flow manufacturing berhubungan dengan produksi produk yang memiliki *high volume standard*. Untuk produk dengan unit yang diskrit seperti mobil, prosesnya disebut *repetitive manufacturing*. Sedangkan produk yang diproduksi dalam *continous flow*, prosesnya disebut *continous manufacturing*. *Flow manufacturing* memiliki empat karakter utama, yaitu sebagai berikut:

- a. Rute aliran produksi tetap dan lokasi setiap *work center* telah diatur sesuai dengan rute produksi tersebut.
- b. Peralatan dan mesin-mesin yang digunakan telah didesain khusus untuk membuat produk tertentu (spesifik). Begitu juga dengan *work center* yang telah di-assign untuk memproduksi produk yang sama.
- c. Aliran material antar *workstation* menggunakan beberapa form.
- d. Kapasitas produksi ditentukan oleh *line* produksi.

2. *Intermittent Manufacturing*

Intermittent manufacturing merupakan proses manufaktur dengan desain produk, *process requirements*, dan *order quantities* yang bervariasi. *Intermittent manufacturing* memiliki beberapa karakteristik, yaitu sebagai berikut:

- a. Aliran kerja pada setiap *workstation* tidak seimbang. Hal ini disebabkan karena pekerjaan pada satu *workstation* bergantung pada part tertentu yang dihasilkan dari *workstation* lainnya.
- b. Mesin dan pekerja mesti fleksibel dalam mengerjakan pekerjaan yang bervariasi. Biasanya mesin-mesin dan *work center* dikelompokkan berdasarkan fungsi kerjanya.

⁸ J.R. Tony Arnold, dan Stephen N. Chapman, *Introduction to Materials Management*, Pearson Education, Inc, 2004, Hal 147

- c. Rata-rata waktu yang dibutuhkan untuk proses produksi pada umumnya panjang.
- d. Kapasitas produksi yang dibutuhkan bergantung pada particular mix produk yang telah dibuat dan sulit untuk diprediksi.

3. *Project Manufacturing*

Project manufacturing biasanya melibatkan kreasi dari satu atau beberapa unit produk. Dalam hal ini diperlukan koordinasi antara divisi *manufacturing*, *marketing*, *purchasing*, dan *engineering*.

2.4.3. *Data Requirements*

Untuk membuat perencanaan *material manufacturing*, *Shop Floor Control (SFC)* mesti memiliki informasi yang berkenaan dengan:

- Apa dan berapa banyak yang diproduksi ?
- Kapan parts dibutuhkan agar penyelesaian produksi tercapai?
- Apa operasi yang dibutuhkan untuk membuat produk, dan berapa lamanya?
- Apa kapasitas yang tersedia pada *work centers*?

Data yang dibutuhkan untuk menjawab pertanyaan-pertanyaan di atas biasanya dikumpulkan dalam suatu database. Database tersebut memuat dua file utama, yaitu *planning* dan *control*.

1. *Planning Files*

1.1. *Item Master File*:

- Nomor part
- Deskripsi part
- *Manufacturing lead time* (waktu normal untuk membuat part)
- Kuantitas di tangan
- Kuantitas tersedia
- Kuantitas dialokasikan
- *On-order quantities*
- *Lot-size quantity*

1.2. *Product Structure File (BOM)*:

- Daftar komponen
- Jumlah komponen untuk tiap 1 produk rakitan

1.3. Routing File:

- Operasi pembuatan produk, dan urutannya
- Deskripsi ringkas tiap operasi
- Peralatan dan asesoris untuk tiap operasi
- *Setup time* tiap operasi
- *Run time* (waktu standar) utk memproses satu unit mel. tiap operasi
- *Lead time* tiap operasi

1.4. Work Center Master File:

- Nomor *work center*
- Kapasitas
- Jumlah shift kerja tiap minggu
- Jumlah jam mesin per shift
- Jumlah jam pekerja per shift
- Efisiensi
- Utilisasi
- Waktu tunggu
- Adanya *work centers* alternatif.

2. Control Files

2.1. Shop Order Master File:

- Nomor *shop order*
- *Order quantity*
- Kuantitas diselesaikan
- Kuantitas skrap
- Kuantitas material direncanakan untuk di-*order*
- *Due date*
- Prioritas
- Kuantitas yg belum selesai
- Informasi biaya

2.2. Shop Order Detail File:

- Nomor operasi
- *Setup hours (planned dan actual)*
- *Run hours*
- Kuantitas selesai dilaporkan
- Kuantitas skrap dilaporkan
- *Due date*

2.4.4. Order Preparation

Setelah perintah untuk memproses suatu order diterima, maka *Shop Floor Control (SFC)* bertanggung jawab untuk membuat perencanaan dan mempersiapkan segala sesuatu yang diperlukan oleh *shop floor*. *Capacity requirements planning* berkaitan langsung dengan *material requirements planning* yang fokus pada *component parts* dan memperhatikan individual order pada setiap *work center*, *work center load*, serta kebutuhan pekerja untuk setiap periode waktu dan setiap *work center*. Pada bagian ini sering terjadi perbedaan antara *planned capacity* dan *actual available*.

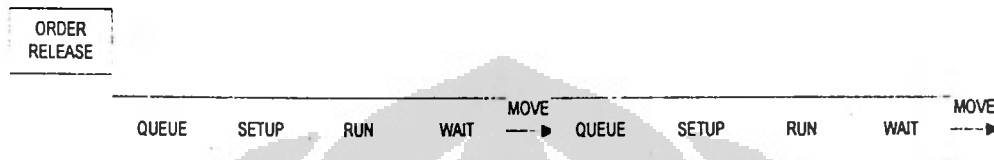
2.4.5. Scheduling

Proses *scheduling* merupakan proses untuk menentukan kapan pemrosesan suatu order dimulai dan selesai pada setiap *work center* sehingga *final due date* dapat ditentukan dengan memperhatikan informasi tentang *manufacturing lead time* setiap *work center*, *required and available capacity*. *Scheduling* bertujuan agar sumber daya yang terdapat pada *manufacturing* tersebut dapat digunakan secara optimal.

Manufacturing lead time merupakan waktu normal yang digunakan untuk memproduksi satu item dalam jumlah tertentu. *Manufacturing lead time* terdiri dari lima elemen, yaitu sebagai berikut :

1. *Queue time*, merupakan waktu yang diperlukan oleh suatu pekerjaan untuk menunggu di *work center* sebelum dikerjakan.
2. *Setup time*, merupakan waktu yang dibutuhkan untuk mempersiapkan operasi *work center*

3. *Run time*, merupakan waktu yang dibutuhkan dalam proses pembuatan *order*
4. *Wait time*, merupakan waktu yang dibutuhkan suatu pekerjaan untuk menunggu di *work center* sebelum dipindahkan ke *work center* berikutnya.
5. *Move time*, merupakan waktu normal yang diperlukan untuk memindahkan material dari satu *workstation* ke *workstation* lainnya.



Gambar 2.4 Manufacturing Lead Time

Dari kelima elemen *manufacturing lead time* di atas, *queue time* merupakan elemen dengan kapasitas terbesar hingga 85%-95% dari total *lead time*. Oleh karena itu, *Shop Floor Control (SFC)* harus *manage queue time* tersebut dengan mengatur aliran kerja yang masuk dan keluar *work center*.

2.4.6. Scheduling Techniques

1. Forward scheduling

Forward scheduling merupakan teknik penjadwalan yang mengasumsikan bahwa pengadaan material dan pelaksanaan operasi untuk sebuah komponen dimulai ketika *order* tersebut telah diterima. Waktu penyelesaian proses lebih awal dari *due date* dan ini menghasilkan inventori. Metode ini biasanya digunakan untuk memproduksi *order* yang paling awal diselesaikan serta menghitung berapa lama waktu penyelesaian operasi tersebut.

2. Backward scheduling

Backward scheduling merupakan teknik penjadwalan yang melaksanakan operasi terakhir pada rute kerja pada jadwal pertama dan dijadwalkan dapat diselesaikan pada saat *due date*. *Backward scheduling* digunakan untuk menentukan kapan operasi suatu *order* harus dimulai dan banyak digunakan oleh industri karena dapat mengurangi inventori *work in process*.

3. Infinite loading

Infinite loading merupakan teknik penjadwalan yang mengasumsikan bahwa setiap *workstation* pada operasi 1,2,3 memiliki kapasitas yang *available* pada saat dibutuhkan. Dan teknik ini tidak memperhitungkan keberadaan *shop order* lainnya.

4. Finite Loading

Finite Loading merupakan teknik penjadwalan yang mengharuskan adanya limit terhadap kapasitas yang *available* pada setiap *work station*. Suatu *order* dapat dipindahkan jadwalnya ke periode lain jika terjadi kekurangan kapasitas.

Terdapat dua metode⁹ untuk mengurangi total *manufacturing load time*, yaitu sebagai berikut:

1. Operation overlapping

Pada *operation overlapping*, operasi kedua dapat dimulai sebelum operasi pertama diselesaikan. Tentunya hal ini dapat mengurangi total *manufacturing load time*, karena terjadinya penghapusan *queue time* dari operasi pertama ke operasi kedua, dan seterusnya.

2. Operation splitting

Pada *operation overlapping*, operasi dilakukan dengan *split order* dalam dua atau lebih *lots* serta menjalankannya pada dua atau lebih mesin secara bersamaan. Hal ini menyebabkan pengurangan *run time* hingga setengahnya meskipun terjadi penambahan waktu *setup* serta memungkinkan satu operator dapat menjalankan satu atau lebih mesin pada satu waktu. *Operation overlapping* dapat dilakukan ketika $setup\ time < run\ time$.

2.4.7. Operation Sequencing

Operation sequencing merupakan metode perencanaan jangka pendek dari *actual jobs* yang akan dijalankan oleh setiap *workcenter* berdasarkan kapasitas dan prioritas. *Dispatching* menjalankan fungsi sebagai penyeleksi dan pengurutan pekerjaan yang *available* dijalankan pada *workcenter*. *Dispatch list* merupakan alat untuk mengontrol prioritas kerja yang berisi tentang:

⁹ J.R. Tony Arnold, dan Stephen N. Chapman, *Introduction to Materials Management*, Pearson Education, Inc, 2004, Hal 155

1. *Plant*, departemen, dan *workcenter*
2. *Part number*, *shop order number*, *operation number*, dan *operation description job*
3. *Standard hours*
4. Informasi prioritas
5. Pekerjaan yang akan datang ke setiap *work center*

Dispatching rule berfungsi untuk mengurangi *work in process inventory*, meminimalisasi jumlah keterlambatan *order*, atau memaksimumkan *output* dari setiap *workcenter*. Berikut adalah beberapa jenis *dispatching rule*:

1. *First Come First Served* (FCFS)

Setiap pekerjaan dilakukan sesuai urutan datangnya instruksi kerja serta tidak memperhitungkan *due dates* dan *processing time*. *First Come First Served* (FCFS) menguntungkan dan adil di mata *customers* yang merasa didahulukan jika lebih dulu datang. Dari segi kriteria penilaian (*average complete time*, *utilization*, *average number of job in system*, dan *average lateness*), *First Come First Served* (FCFS) kurang baik untuk semua kriteria.

2. *Earliest Job Due Date* (EDD)

Pada *rule* ini, pekerjaan yang didahulukan adalah pekerjaan dengan *due date* paling awal. *Earliest Job Due Date* (EDD) dapat meminimumkan *max tardiness* yg penting bila penalti tinggi. *Earliest Job Due Date* (EDD) baik untuk diterapkan bila *lateness* merupakan hal yang terpenting.

3. *Shortest Processing Time* (SPT)

Pekerjaan diurutkan berdasarkan *process time*, dimana pekerjaan dengan *process time* paling kecil dilakukan di awal. Pada *rule* ini *due date* diabaikan, menyebabkan *number of jobs processed* bertambah. Kelemahannya, pekerjaan yang berdurasi lama tdk diprioritaskan sehingga mempengaruhi pertimbangan *customers*. *Shortest Processing Time* (SPT) paling bermanfaat untuk “shop” yang paling padat (*highly congested*) karena dapat meminimumkan *mean flow time*, *mean number of jobs in the system* (WIP), dan *percent of late jobs*.

4. *Longest Processing Time* (LPT)

Pekerjaan diurutkan berdasarkan *process time*, dimana pekerjaan dengan *process time* paling besar dilakukan di awal. *Longest Processing Time* (LPT) diterapkan untuk pekerjaan subkontrak, dimana pekerjaan besar dan lama dikerjakan *in-house* dan pekerjaan jangka pendek keluar

Satu *rule* lain yang perlu diperhatikan adalah *critical ratio* (CR). *Critical ratio* (CR) merupakan index yang menunjukkan hubungan prioritas antara satu *order* dengan *order* lainnya pada satu *workcenter*. *Critical ratio* (CR) didasarkan pada perbandingan antara *actual time remaining* dengan *lead time remaining*. Berikut adalah tabel yang menunjukkan prioritas *order* berdasarkan nilai *critical ratio* (CR).

Tabel 2.2 Hubungan Antara *Critical Ratio* dan Prioritas *Order*

Critical Ratio	Condition	Priority
critical ratio < 1	actual time < lead time	order is behind schedule
critical ratio = 1	actual time = lead time	order is on schedule
critical ratio > 1	actual time > lead time	order is ahead schedule
critical ratio ≤ 0	today's date > due date	order is already late

2.4.8. Production Reporting

Production reporting memberikan *feedback* tentang apa yang sebenarnya terjadi pada *plant floor*. *Shop Floor Control* (SFC) memerlukan laporan ini untuk membuat prioritas *order* yang semestinya serta untuk menjawab pertanyaan yang berhubungan dengan *deliveries*, *status order*, *status inventori*, *department and workcenter performance*.

Sedangkan bagi manufacturing management, laporan ini berguna untuk membuat perencanaan operasi pabrik.

2.4.9. Arsitektur Sistem Informasi Dalam *Shop Floor Control*

Permasalahan dalam perencanaan dan pengendalian produksi telah ada sejak dimulainya proses produksi secara masal. Para pakar telah mengembangkan berbagai metode dan model untuk mendapatkan hasil yang mendekati optimal pada proses produksi. Permasalahan di perencanaan dan pengendalian produksi pada awal munculnya industri masih relatif sederhana sehingga model yang dikembangkan sesuai dengan permasalahan yang ada. Seiring dengan meningkatnya kompleksitas permasalahan dan volume pekerjaan yang besar serta tuntutan konsumen yang semakin kritis, memerlukan pengembangan model yang mampu mengatasi permasalahan yang ada. Dengan berkembangnya teknologi elektronika, telekomunikasi dan informasi dapat dimanfaatkan untuk mendapatkan hasil yang lebih baik.

Secara garis besar arsitektur sistem perencanaan dan pengendalian produksi¹⁰ dapat dikelompokkan menjadi tiga yaitu :

2.4.9.1. Sistem Terpusat

Sistem terpusat merupakan model yang pertama dikembangkan dalam proses perencanaan dan pengendalian produksi. Dalam sistem terpusat ini semua keputusan ditentukan oleh satu komando paling atas. Jika terjadi permasalahan di suatu bagian maka permasalahan akan dibawa ke pusat dan akan diputuskan di level atas. Sistem ini cukup baik jika permasalahan yang dihadapi pada sistem produksi masih relatif kecil. Akan tetapi untuk permasalahan yang cukup besar model ini akan menghadapi masalah kecepatan pemrosesan dan efektifitas kerja.

2.4.9.2. Sistem Hierarkhi

Dengan bertambahnya kompleksitas permasalahan dan volume data yang terjadi di sistem produksi, menyebabkan sistem terpusat kurang memberikan hasil yang baik sehingga dikembangkan sistem baru yaitu sistem hierarkhi. Dalam sistem ini permasalahan yang cukup besar dan kompleks dibagi menjadi beberapa bagian dan diurutkan berdasarkan level kepentingan dalam sistem produksi.

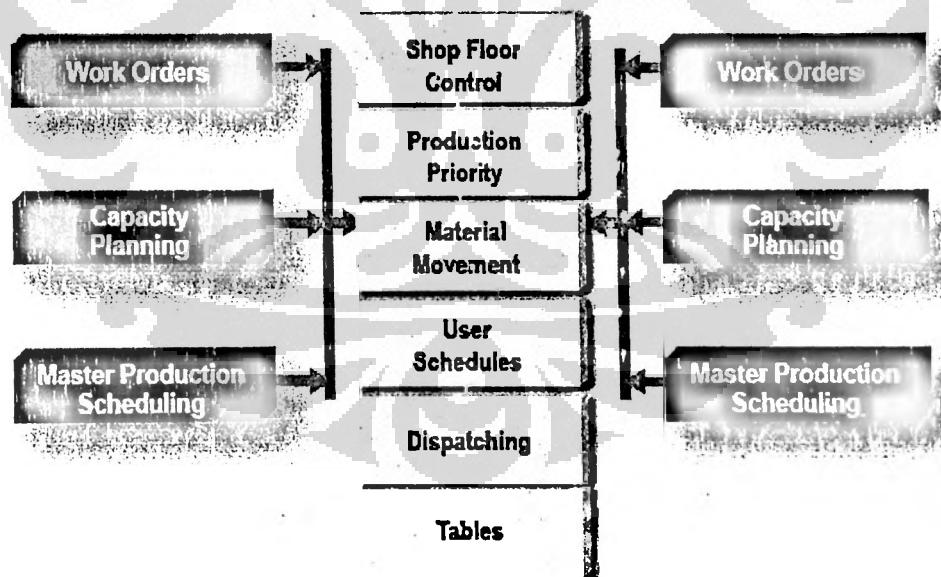
¹⁰ Arian Zwegers, On Systems Architecting : *A Study in Shop Floor Control to Determine architecting concepts and principles* Eindhoven : University Press Facilities . (1998.)

2.4.9.3. Sistem Heterarkhi

Sistem heterarkhi dikembangkan untuk menjawab kompleksitas permasalahan dan volume data yang makin besar serta untuk meningkatkan keandalan sistem. Dalam sistem heterarkhi permasalahan sistem produksi didistribusikan menjadi permasalahan di masing-masing bagian sistem produksi dan masing-masing bagian diberikan autonomi untuk menyelesaikan permasalahan di masing-masing bagian untuk mendapat hasil yang terbaik. Setelah solusi di masing-masing bagian didapatkan maka selanjutnya dari solusi-solusi per bagian tersebut dikoordinasikan apakah sudah sesuai dengan tujuan sistem produksi secara global dan apakah terjadi konflik antar bagian ataupun overlapping. Jika masih terjadi konflik ataupun overlap maka diperlukan pemecahan masalah yang baru dari masing-masing bagian sampai ditemukan solusi global yang mendekati optimal.

2.4.10. Shoop Floor Control Integration

Dalam implementasinya, sistem pengendalian lantai produksi ini berkaitan langsung dengan tiga sistem utama yaitu *work order*, *capacity planning* dan *master production scheduling*. Hubungan antara ketiga sistem tersebut dengan pengendalian lantai produksi dapat dilihat pada gambar 2.3 berikut.



Gambar 2.5 Shoop Floor Control Integration



Gambar 2.7 Siklus Penjadwalan

Dalam gambar diatas jelas terlihat bahwa penyusunan penjadwalan operasi dimulai dari penentuan besarnya volume permintaan barang / jasa yang diminta oleh konsumen yang kemudian dilanjutkan dengan:

- Rencana pengaturan tenaga kerja
- Rencana pengaturan mesin / peralatan
- Rencana pengaturan material

Selanjutnya begitu jadwal disusun maka akan dioperasionalisasikan dalam bentuk pelaksanaan. Dalam kenyataannya tidak selalu pelaksanaan sesuai dengan rencana. Apabila timbul perbedaan antara pelaksanaan dan rencana maka perlu dilakukan tindakan koreksi terhadap:

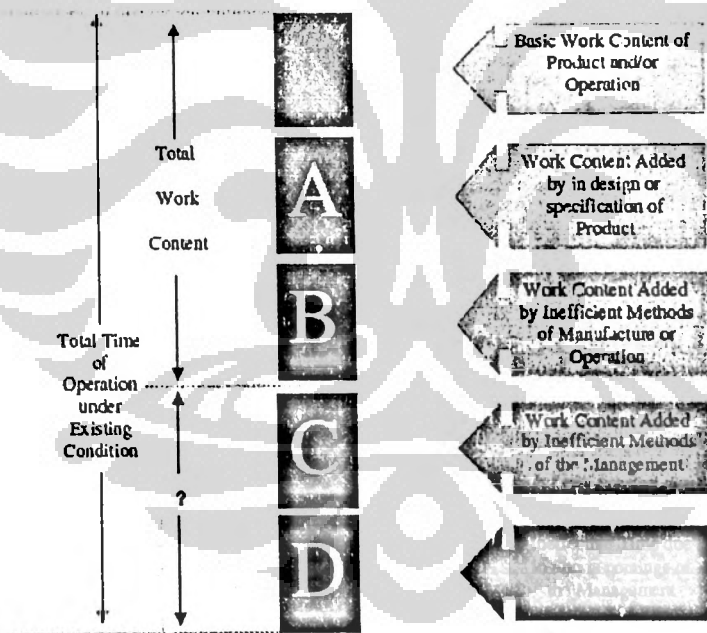
- Jadwal yang telah dibuat, ada kemungkinan rencana yang dibuat terlalu optimis sehingga sulit untuk dilaksanakan atau kemungkinan lain terjadi perubahan volume permintaan yang cukup berarti. Apabila hal ini terjadi maka perlu adanya perubahan rencana yang lebih realistis
- Pelaksanaan yang dilakukan, tidak jarang terjadi hambatan di dalam pelaksanaan baik yang berkaitan dengan manusianya maupun peralatan serta faktor – faktor eksternal lain yang mempengaruhinya. Apabila hal ini terjadi maka perlu diadakan perbaikan – perbaikan didalam pelaksanaannya. Dengan demikian akan terlihat bahwa antara proses perencanaan dan perbaikannya (pengendalian) akan selalu terjadi dan mengelinding secara kontinu. Oleh sebab itu antara perencanaan dan

Universitas Indonesia

pengendalian merupakan 2 kegiatan yang harus dilakukan secara simultan oleh orang yang bertanggungjawab atas kelancaran suatu sistem usaha. Dari urutan tersebut nampak bahwa jadwal operasi tidak selalu sama dengan volume permintaan barang / jasa, sebab tidak semua volume permintaan akan dipenuhi jika sumber daya yang diperlukan untuk merealisasikan tidak tersedia.

Dalam lingkup yang lebih luas, pihak manajemen pun harus bertanggung jawab untuk mengatasi pemborosan waktu kerja. Ketidakmampuan manajemen dalam mengelola sumber daya perusahaan, misalnya kurang baik pengaturan penjadwalan / rencana kerja, atau kurang tepatnya kebijakan sumber daya manusia pada umumnya dapat menyebabkan borosnya waktu kerja manufaktur.

Selain itu pihak manajemen juga harus bisa membedakan antar waktu kerja yang efektif dan yang kurang efektif. Gambar 2.6 dibawah ini menunjukkan bahwa terdapat lima jenis waktu kerja¹² dalam sebuah industri manufaktur.



Gambar 2.8 Waktu kerja efektif dan tidak efektif pada manufaktur

¹² Thomas M. Carol and Robert D. Dean "A Bayesian Approach to Plant – Location Decision" Sciences

2.6. Persoalan Umum Penjadwalan

2.6.1 Pengurutan (*sequencing*) produk

Conway mendefinisikan pengurutan (*sequencing*) sebagai proses membuat urutan produk pada satu mesin dalam jangka waktu tertentu, sedangkan penjadwalan (*scheduling*), didefinisikan sebagai proses pengurutan secara menyeluruh pada beberapa mesin. Kenneth R. Baker memberikan definisi penjadwalan sebagai proses pengalokasian sumber – sumber untuk memilih sekumpulan tugas dalam jangka waktu tertentu.

Dengan definisinya tersebut Baker menyatakan bahwa penjadwalan berfungsi sebagai alat pengambil keputusan yaitu untuk menetapkan suatu jadwal. Persoalan penjadwalan menjadi masalah jika ada tugas–tugas yang harus ditetapkan urutannya, yang mana yang harus didahulukan dan yang mana yang kemudian. Masalah ini semakin rumit jika ditemui n tugas untuk proses pada m mesin, yang berarti ada $(n) \times m$ buah jadwal yang mungkin ditemui dan satu diantaranya harus dipecahkan untuk memenuhi ukuran penampilan yang telah ditetapkan.

Dari struktur permasalahan, sistem penjadwalan operasional dipengaruhi oleh situasi tertentu. Model penjadwalan dapat dibedakan oleh beberapa keadaan berikut:

1. Proses dengan mesin tunggal atau proses dengan mesin ganda.
2. Pola aliran proses yang identik atau pola aliran proses yang sembarang
3. Sejumlah tertentu dan tetap dari pada pekerjaan atau kedatangan kontinu
4. Informasi yang lengkap atas pekerjaan dan mesin atau adanya ketidakpastian pada salah satu atau kedua elemen diatas.

Pada model pertama, sejumlah mesin dapat dibedakan atas mesin tunggal dan mesin ganda. Masalah mesin tunggal sangat mendasar untuk analisa menyeluruh dan masalah ini dapat diterapkan pada mesin ganda.

Pada model ke dua, pola aliran dapat dibedakan atas flow-shop dan job shop. Setiap pekerjaan dalam job-shop mempunyai aliran yang berbeda, sedangkan dalam flowshop hanya dijumpai pola aliran yang identik dari satu mesin ke mesin yang lain.

Walau pada flow-shop semua tugas akan mengalir pada jalur produksi yang sama, yang biasa dikenal dengan pure flow-shop, namun kadang kala dapat berbeda pola alirannya. Pertama, disebabkan suatu shop dapat menangani tugas yang bervariasi. Kedua, tugas yang datang ke dalam flow-shop tidak harus dikerjakan pada semua mesin. Jadi mungkin saja suatu proses dilewati. Jenis flow-shop diatas disebut dengan general flow-shop.

Pada model ke tiga, pola kedatangan tugas dapat dibedakan atas pola kedatangan statis dan dinamis. Pada pola statis, tugas datang secara bersamaan dan siap dikerjakan pada mesin – mesin yang tidak bekerja. Dipihak lain pola dinamis mempunyai sifat kedatangan tugas tidak tentu, jadi di jumpai adanya variabel waktu.

Pada model ke empat, perilaku elemen – elemen penjadwalan dapat dibedakan atas deterministik dan stokastik. Model deterministik dapat dilihat dengan adanya kepastian atas informasi tentang elemen – elemen yang ada. Sedangkan pada model stokastik, mengandung unsur ketidakpastian. Dengan demikian informasi ini hanya dapat diramal dengan metoda statistik. Elemen – elemen yang dimaksud adalah sebagai berikut :

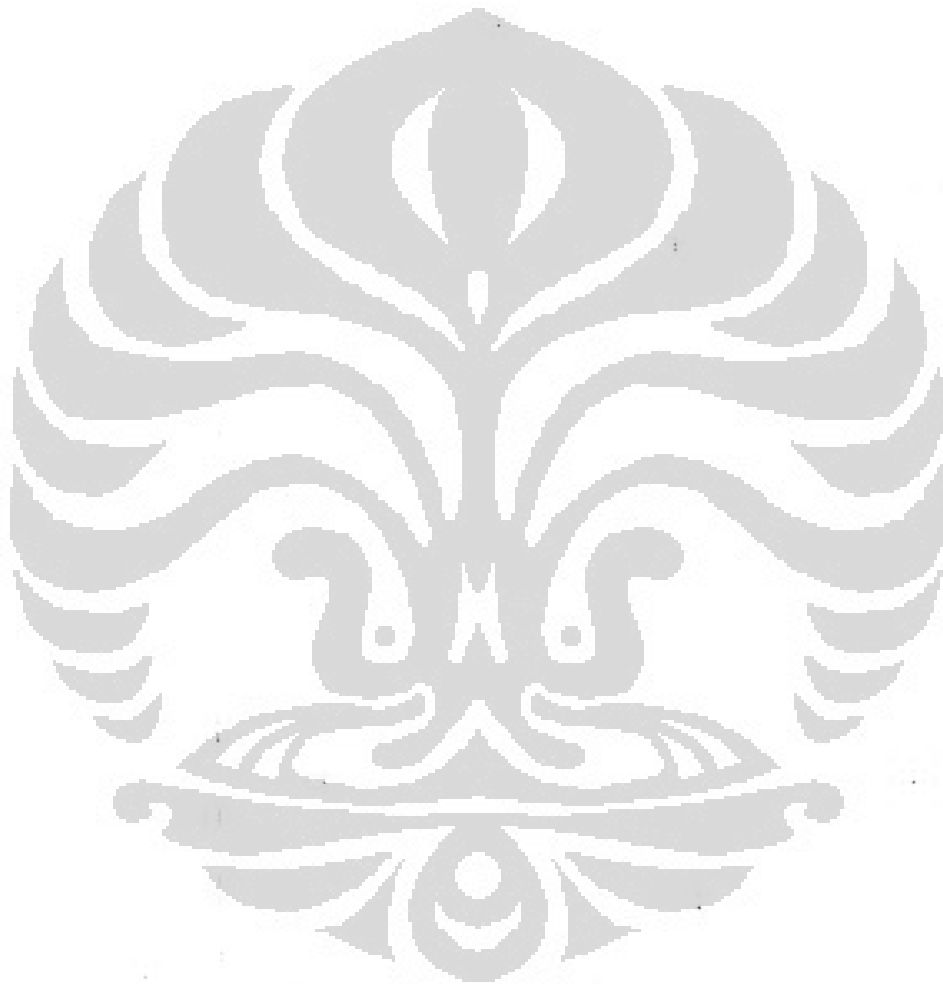
1. Karakteristik tugas dari segi kedatangan, batas waktu penyelesaian, dan perbedaan kepentingan diantara tugas
2. Karakteristik tugas dari segi banyaknya operasi, susunan mesin, waktu proses dan kendalanya.
3. Karakteristik mesin dari segi jumlah dan kapasitas mesin yang dipunyai, dan kemampuan serta kecocokan tiap mesin dengan tugas yang diberikan.

2.6.2. Penggiliran Kerja

Pergiliran kerja dilakukan agar jumlah tenaga kerja yang ada dapat dipergunakan dengan seefektif mungkin dan tidak adanya pekerja yang bekerja dengan *performance* yang dikeluarkan tidak penuh. Begitupun sebaliknya jumlah tenaga kerja yang ada tidak mencukupi untuk memenuhi permintaan dari konsumen, maka pihak manajemen perusahaan harus bisa sebaik mungkin untuk mengelola jumlah tenaga kerja yang dimiliki. Hal yang paling sering terjadi adalah bahwa sebuah pabrik harus bekerja penuh selama 7 hari dalam satu minggu, sementara seorang pekerja biasanya bekerja 5 hari dalam satu minggu, sehingga ada dua hari

Universitas Indonesia

libur yang diperoleh pada hari yang berturutan. Disamping itu kebutuhan tenaga kerja setiap harinya juga tidak sama, ada hari yang sibuk dan ada yang tidak. Oleh karena itu sistem pergiliran kerja ini akan sangat efektif untuk memenuhi permintaan yang cenderung berubah-ubah. Begitupun untuk jumlah permintaan yang tidak terlalu banyak berubah, dengan sistem pergiluran kerja ini, perusahaan juga dapat menghemat biaya tenaga kerja yang harus dikeluarkan, karena dapat menggunakan sejumlah tenaga kerja tertentu dengan tingkat utilisasi yang tinggi.



Universitas Indonesia

3. PENGUMPULAN DAN PENGOLAHAN DATA

Pada tahap pengumpulan dan pengolahan data, perlu dilakukan pemahaman terhadap konsep bisnis yang menjadi objek penelitian. Dengan demikian, sebelum melakukan pengumpulan dan pengolahan data diperlukan adanya suatu kajian lebih dalam mengenai kondisi riil bisnis manufaktur dari objek penelitian, baik dari segi jenis produksi yang digunakan, produk yang dihasilkan, biaya produksi yang dikeluarkan, perencanaan dan pengendalian produksi, serta jumlah *resources* yang digunakan dalam proses produksi tersebut. Pemahaman bisnis ini bertujuan untuk menangkap permasalahan yang ada di dalamnya. Permasalahan inilah yang akan dibawa pada suatu konsep tujuan. Berdasarkan pada konsep tujuan, maka dilakukan identifikasi data yang diperlukan untuk dapat mencapai tujuan yang diinginkan.

3.1. Pemahaman Bisnis PT.X

PT.X merupakan sebuah perusahaan yang bergerak di bidang produksi obat-obatan farmasi dengan produk utama adalah obat multivitamin. Ditinjau dari kedatangan konsumen dan jumlah permintaan produk per periode waktu, transformasi produksi yang digunakan oleh PT.X merupakan transformasi produksi *flow shop* tipe *flow line / batch*, dengan karakteristik produksi sebagai berikut:

1. Produksi dilakukan secara kontinu, baik ada pemesanan maupun tidak ada pemesanan yang tentunya dikendalikan oleh perencanaan produksi setiap periode waktu. Jumlah pesanan produk setiap bulan biasanya relatif besar (massal) dan jenis produksinya standar.
2. Volume produksi produk cukup besar bila dibandingkan dengan transformasi produksi tipe *job shop* ataupun *project*. Dengan volume produksi yang cukup besar ini, proses produksinya-pun termasuk dalam kategori proses produksi yang sederhana (proses yang berulang-ulang), sehingga karyawan yang dipekerjakan dalam transformasi produksi *flow shop* memiliki *skill* yang lebih rendah (*unskilled labour*) bila dibandingkan dengan transformasi produksi tipe *job shop* ataupun *project*. Karyawan

tersebut dialokasikan di setiap *work centre* yang terdiri dari beberapa jumlah mesin sesuai dengan proses produksinya.

Peralatan (mesin) yang digunakan dalam proses produksi *flow shop* ini lebih spesial sesuai dengan fungsinya masing-masing. Mesin-mesin disusun dalam satu lini produksi berdasarkan aliran produksi setiap proses. Sehingga tataletak yang diterapkan dalam pabrik adalah berdasarkan produk (*product layout*).

Tabel 3. 1 Produk-Produk PT.X

No.	Item	Product	Unit of Measure	Batch size
1	AAAA	Product A	Bottle	24.000
2	BBBB	Product B	Box/30	12.800
3	CCCC	Product C	Box/100	2.400
4	DDDD	Product D	Box/200	1.200
5	EEEE	Product E	Tin/1000	2.560
6	FFFF	Product F	Bottle	19.200

(Sumber: PT X)

Pada tabel diatas, dapat kita lihat bahwa PT X memproduksi enam jenis produk dengan *unit of measure* dan *batch size* yang masing-masing berbeda.

Dalam melakukan kegiatan produksi ada berbagai faktor yang harus dikelola oleh manajemen produksi perusahaan PT.X yang sering disebut sebagai faktor – faktor produksi yaitu:

1. Material atau Bahan

Manajemen produksi perusahaan mengelola ketersediaan material bahan baku yang diperlukan dalam proses produksi sehingga produksi berjalan normal.

2. Mesin atau peralatan

Manajemen produksi merencanakan kebutuhan mesin atau peralatan yang diperlukan dalam proses produksi, tataletaknya, serta biaya maintenance baik *preventive maintenance* maupun *scheduled maintenance*.

3. Manusia atau karyawan

Universitas Indonesia

Manajemen perusahaan mengelola serta merencanakan *capacity resources planning* yang dibutuhkan dalam produksi sehingga dapat meningkatkan *output* dan memperlancar proses produksi.

4. Modal

Manajemen perusahaan, terutama di bidang keuangan, mampu mengelola dan merencanakan keuangan perusahaan dengan meningkatkan keuntungan dan menjaga stabilitas keuangan perusahaan.

Sebagai perusahaan manufaktur yang memproduksi produk yang siap digunakan oleh konsumennya, maka manajemen operasi produksi perusahaan bertanggung jawab atas dihasilkannya *output* berupa produk obat-obatan yang sesuai dengan permintaan dan kebutuhan konsumen dengan kualitas yang baik dan harga yang terjangkau serta disampaikan tepat pada waktunya. Berdasarkan hal ini, maka terdapat beberapa hal yang bias menjadi *key performance indicator* bagi manajemen operasi produksi perusahaan, yaitu:

1. Biaya Produksi

Bila dikaitkan dengan tujuan suatu industri manufaktur pada umumnya, maka ukuran kinerja sering diukur dengan keuntungan yang dapat dicapai perusahaan. Namun sistem produksi hanyalah salah satu dari sub sistem yang ada dalam suatu sistem industri manufaktur, sehingga untuk mengukur seberapa besar kontribusi sistem operasi di dalam pencapaian keuntungan bukanlah hal yang mudah. Oleh karena itu, untuk mengukur kinerja sebuah sistem produksi diambil dengan ukuran waktu operasi tertentu (biasanya dalam waktu satu tahun).

Biaya produksi ini meliputi: semua biaya yang dikeluarkan untuk menghasilkan produk ketangan konsumen termasuk biaya pemeliharaan mesin dan peralatan pabrik. Dengan biaya produksi yang seminimal mungkin, diharapkan bahwa produk dapat dipasarkan dengan harga yang dapat dijangkau oleh konsumen.

2. Kualitas Produk

Realitas saat ini menunjukkan bahwa konsumen tidak hanya memilih produk yang harganya minimal namun juga produk yang berkualitas, oleh sebab itu baik buruknya suatu sistem produksi juga diukur dari kualitas produk yang dihasilkan. Ukuran kualitas produk yang dimaksudkan disini tentunya yang disesuaikan dengan kebutuhan konsumen bukan hanya dari ukuran kualitas secara teknologi.

3. Tingkat Pelayanan

Bagi konsumen untuk menilai baik buruknya suatu sistem operasi produksi lebih dinilai dari pelayanan yang dapat diberikan oleh sistem produksi kepada konsumen itu sendiri. Tingkat pelayanan (*service level*) ini pada umumnya merupakan ukuran yang lebih banyak dipengaruhi oleh faktor – faktor kualitatif, walaupun demikian beberapa ukuran obyektif yang sering digunakan antara lain:

- o Ketersediaan (*availability*) dan kemudahan untuk mendapatkan produk.
- o Kecepatan pelayanan baik yang berkaitan dengan waktu pengiriman (*delivery time*) maupun waktu pemrosesan (*processing time*).

Salah satu fungsi manajemen yang memiliki peranan penting dalam bisnis manufaktur adalah perencanaan produksi. Pihak manajemen harus mampu menyusun rencana penggunaan sumber daya perusahaan, serta ia harus mampu melaksanakan fungsi – fungsi manajemen lainnya (pengorganisasian, komunikasi, koordinasi, kepemimpinan serta motivasi), agar perencanaannya dapat direalisasikan dengan baik. Dalam hal ini terjadi siklus fungsi manajemen yang saling mempengaruhi.

Manajemen tidak akan efektif menjalankan kegiatannya, jika perencanaan yang disusunnya tidak baik. Sebaliknya walaupun rencana kerja sudah disusun baik, tidak akan berpengaruh banyak, jika kemampuan pelaksanaannya tidak baik. Manajemen pabrik sangat membutuhkan informasi – informasi tentang berapa besar kapasitas pabrik yang dipunyai, berapa besar kapasitas yang sudah digunakan dan berapa banyak kapasitas yang tersedia untuk menjalankan rencana

kerjanya. Perencanaan produksi aggregate, bertujuan untuk menguji apakah kapasitas pabrik yang dipunyai masih mampu mengerjakan sejumlah rencana produksi (*feasibility*). Jika perencanaan aggregate ini sudah *feasible*, maka langkah berikutnya adalah menyusun rencana produksi untuk setiap item / produk, sehingga menghasilkan jadwal induk produksi (JIP).

Penjadwalan produksi yang baik adalah penjadwalan yang dapat memenuhi fluktuasi permintaan dari berbagai jenis produk, dengan memperhatikan kapasitas (tenaga kerja, mesin dan sumber daya lain) yang ada. Makin terbatas fasilitas yang dipunyai, serta makin banyak jenis produk yang harus dibuat, makin kompleks persoalan perencanaan dan pengendalian produksinya. Untuk mengurangi kompleksitas masalah, perlu ada suatu metodologi yang membantu fungsi PPC khususnya dalam menetapkan jadwal produksi Induk (*Master Production Scheduling*). Pihak manajemen PT.X melakukan penjadwalan produksi tahunan berupa jumlah produksi agregat yang kemudian di disagregasi ke dalam kuantitas setiap model produk, serta dibagi dalam duabelas bulan produksi. Periode waktu penjadwalan produksi ini dipengaruhi oleh ketepatan dalam meramal keadaan pasar, kemampuan untuk melakukan penyesuaian terhadap perubahan pasar, *lead time* pengadaan fasilitas / sumber daya yang dibutuhkan, *lead time* produksi dan lain – lain.

Setelah Jadwal Induk Produksi tersebut disusun maka salah satu permasalahan yang perlu diperhatikan oleh manajemen perusahaan adalah mengatur dan merencanakan kapasitas sumber daya yang diperlukan dalam proses produksi baik tenaga kerja maupun jumlah mesin di masing-masing *work centre*. Masalah perencanaan kapasitas dan penjadwalan produksi muncul ketika sebuah perusahaan berkeinginan untuk mendirikan suatu unit usaha sampai dengan pengoperasian rutin unit usaha tersebut. Pertanyaan – pertanyaan yang sering muncul pada tahap persiapan: pendirian unit usaha diantaranya adalah berapa besar ukuran pabrik yang akan didirikan, apakah akan didirikan disatu lokasi atau beberapa lokasi, dan sebagainya.

Jumlah sumber daya/fasilitas mesin yang dimiliki oleh PT X dapat dilihat seperti yang ada pada tabel 3.2 dibawah ini

Universitas Indonesia

Tabel 3. 2 Jumlah Mesin di Masing-Masing *Work Centre*

Work Centre	Number of Machine (Unit)
Solid NBL	40
Sterile NBL	6
Liquid NBL	6
Betalactam	10
Cephalosphorin	5
Central Packaging	6
Total	73

(Sumber: PT X)

PT X ini terdapat enam *work centre* yang memiliki sumber daya mesin yang berbeda-beda di setiap *work centre* nya. Sehingga kapasitas di setiap *work centre* akan berbeda-beda pula.

Sedangkan masalah yang muncul setelah beroperasinya unit usaha tersebut diantaranya ialah bagaimana mengatur jumlah dan alokasi mesin dan peralatan serta karyawan yang akan menghadapi suatu pola permintaan dalam suatu kurun waktu tertentu, perlukah memberikan subkontrak kepada perusahaan lain, apakah suatu pesanan yang datang akan diterima atau ditolak, bagaimana mengatur mesin dan peralatan jika telah ditentukan suatu target produksi tertentu. Perencanaan kapasitas yang telah ditentukan memiliki *critical impact* yang cukup besar terhadap perusahaan, seperti struktur biaya produksi dan kebijakan inventori.

Jika keputusan akan perencanaan kapasitas telah dibuat, maka manajemen perusahaan akan merencanakan produksi harian untuk memenuhi jadwal induk produksi yang telah dikeluarkan. Dimana keputusan berapa jumlah produksi harian yang akan diproduksi juga mempertimbangkan total biaya listrik pabrik yang dikeluarkan, biaya pekerja normal dan lembur, dan pergiliran kerja (*shift*), sehingga dengan jumlah perintah produksi yang berbeda-beda untuk setiap *work centre* akan dihasilkan sebuah keputusan produksi harian yang efisien dengan total biaya yang seoptimal mungkin.

Untuk memproduksi enam jenis produk yang ada, masing-masing *works centre* memiliki kapasitas masing-masing, yang bergantung pada jumlah mesin, tenaga kerja dan proses dari masing-masing produk. Jumlah mesin yang digunakan pada masing-masing proses di enam *work centre* PT.X berbeda tergantung jenis proses di masing-masing *work centre* tersebut. Sehingga kondisi kapasitas masing-masing *work centre* dapat dilihat pada tabel 3.3 dibawah ini

Tabel 3. 3 Kondisi Kapasitas Masing-masing *Work Centre* PT X

Work Centre	Waktu Proses (Menit)	Jumlah Mesin (Unit)	Kapasitas Maksimu m/Shift
A	0,08	40	6000
B	0,05	6	9600
C	0,08	6	6000
D	0,08	10	6000
E	0,075	5	6400
F	0,1	6	4800
Total	0,465	73	38800

(Sumber: PT X)

Pada tabel 3.3 diatas dapat dilihat bahwa dengan jumlah mesin dan pekerja yang tersedia akan berpengaruh langsung terhadap waktu proses, sehingga kondisi kapasitas maksimum yang mampu dihasilkan oleh setiap *work centre* akan berbeda-beda pula, walaupun ada beberapa *work centre* yang memiliki jumlah kapasitas yang sama, namun hal ini juga bergantung pada proses dari masing-masing produk yang diproduksi di setiap *work centre*.

3.2. Konsep Tujuan

Untuk dapat melakukan pengumpulan data yang terfokus, maka perlu dibuat suatu konsep tujuan yang ingin diperoleh dari hasil pengumpulan data tersebut. Berdasarkan pemahaman bisnis, maka dibentuk suatu konsep tujuan sebagai berikut:

- Pemahaman mengenai waktu proses produksi dan kapasitas maksimum produksi PT.X dalam periode tertentu.
- Pemahaman mengenai kebutuhan listrik (mesin di setiap *work centre* dan penggunaan AC sentral) untuk operasional pabrik

Universitas Indonesia

- Pemahaman mengenai hubungan keterkaitan antara perencanaan jadwal induk produksi, rute produksi, perencanaan kapasitas sumber daya pekerja dan pengendalian rantai produksi dalam proses produksi, sehingga permintaan pasar terhadap produk dan biaya yang operasional pabrik dapat terpenuhi secara efisien dan efektif.

3.3. Identifikasi Data

Dalam tahap pengumpulan data, sebagaimana penjelasan terhadap jenis industri yang dijalankan oleh PT X ini, maka dalam perancangan sistem yang akan dibuat ini terdapat beberapa asumsi yang digunakan untuk proses selanjutnya, yaitu tahap pengolahan data.

1. Jumlah mesin dan pekerja adalah tetap. Saat ini PT.X memiliki 73 mesin yang dilalokasikan di masing-masing *work centre*. Dalam perencanaan penjadwalan produksi harian, PT.X tidak akan menambah jumlah mesin dan pekerjanya, jika ternyata kapasitas maksimum harian belum mencukupi permintaan maka disesuaikan dengan melakukan *overtime*
2. Upah pekerja pada waktu *overtime* (malam) adalah 150% gaji normal, dan 200% jika *overtime* dilakukan pada hari sabtu.
3. Biaya listrik untuk waktu normal, diluar beban puncak diberi koefisien 1, sedangkan untuk yang beban puncak di beri koefisien 1.5
4. Waktu proses merupakan waktu untuk menghasilkan 1 unit produk.
5. Jam kerja produksi adalah 1 bulan = 22 hari kerja, normal 2 shift perhari, dengan 8 jam per shift. Jam lembur maksimum untuk hari biasa adalah 4 jam, sedangkan lembur dihari sabtu maksimum 8 jam.

3.4. Pengumpulan dan Hasil Data

3.4.1. Klasifikasi *Work Centre* di Masing-Masing Lini Produksi

Berikut adalah tabel yang menunjukkan klasifikasi *work centre* di masing-masing lini produksi di pabrik PT.X.

Tabel 3. 4 Klasifikasi *Work Centre* di Masing-Masing Lini Produksi

No.	Work Centre
WRC01	Solid NBL
WRC02	Sterile NBL
WRC03	Liquid NBL
WRC04	Betalactam
WRC05	Cephalosphorin
WRC06	Central Packaging

(Sumber: PT X)

Dari tabel di atas, dapat dilihat bahwa lini produksi PT.X terbagi ke dalam enam *work centre*; *Solid NBL*, *Sterile NBL*, *Liquid NBL*, *Betalactam*, *Cephalosphorin*, dan *Central Packaging* yang masing-masingnya memiliki proses tertentu yang lebih spesifik.

3.4.2. Jumlah Mesin dan Pekerja di Setiap *Work Centre*

Jumlah mesin dan pekerja di setiap *work centre* dapat dilihat pada tabel 3.5 berikut.

Tabel 3. 5 Jumlah Mesin dan Pekerja di Setiap *Work Centre*

Work Centre	Number of Machine (Unit)	Man
Solid NBL	40	20
Sterile NBL	6	10
Liquid NBL	6	9
Betalactam	10	11
Cephalosphorin	5	6
Central Packaging	6	30
TOTAL	73	86

(Sumber: PT X)

Berdasarkan tabel 3.5 diatas, diperoleh data bahwa jumlah mesin dan pekerja untuk setiap *work centre* berbeda tergantung kebutuhan dari masing-masing *work centre*. Secara keseluruhan, jumlah mesin yang dibutuhkan oleh PT.X untuk memproduksi seluruh produknya adalah sebesar 73 mesin dengan jumlah mesin terbanyak terdapat pada *work centre Solid NBL (work centre 1)* yaitu sebanyak 40 mesin, sedangkan total jumlah pekerja yang dibutuhkan dalam proses produksi

Universitas Indonesia

PT.X adalah sebanyak 86 orang dengan *work centre Central Packaging (work centre 6)* membutuhkan jumlah pekerja terbanyak yaitu 30 orang.

3.4.3. Jam Kerja Produksi

Berikut adalah tabel yang memuat data waktu operasi produksi pekerja setiap hari di seluruh *work centre*.

Tabel 3. 6 Operasi Produksi Pekerja di Seluruh *Work Centre*

Operation Time	Working Time (Minutes)
23,00 - 03,00 dan 04,00 - 08,00	480
08,00 - 12,00 dan 13,00 - 17,00	480
17,00 - 21,00	240

Berdasarkan tabel di atas, dapat dilihat bahwa jam operasi kerja produksi pabrik adalah 16 jam/hari yang dibagi dalam 2 *shift* kerja dan jam lembur maksimum adalah 4 jam/hari. Operasi produksi dimulai pada pukul 23:00 WIB-17:00 WIB, sedangkan jam lembur dimulai pada pukul 17:00 WIB-21:00 WIB. Dan perhitungan 1 bulan produksi adalah 22 hari kerja.

3.4.4. Waktu Proses, Kapasitas, dan Biaya

Tabel 3. 7 Waktu Proses, Kapasitas, dan Biaya

Work Centre	Waktu Proses (Menit)	Jumlah Mesin (Unit)	Jumlah pekerja (orang)	Kapasitas Maksimum/Shift (Unit)	Gaji Pekerja/jam	Kebutuhan Listrik / Unit	Maksimum Lembur (Unit)
A	0,08	40	20	6000	Rp300.000	Rp6,94	3.000
B	0,05	6	10	9600	Rp150.000	Rp4,34	4.800
C	0,08	6	9	6000	Rp135.000	Rp6,94	3.000
D	0,08	10	11	6000	Rp165.000	Rp6,94	3.000
E	0,075	5	6	6400	Rp90.000	Rp6,51	3.200
F	0,1	6	30	4800	Rp450.000	Rp8,68	2.400
Total	0,465	73	86	38800	Rp1.290.000	40	19.400

Waktu proses merupakan waktu yang dibutuhkan oleh satu *work centre* untuk menghasilkan satu unit produk. Kapasitas maksimum per shift merupakan jumlah unit yang mampu diproduksi oleh sebuah *work centre* pada jam kerja normal shift

tersebut. Gaji pekerja per jam, merupakan gaji per jam yang harus dibayarkan oleh perusahaan untuk sejumlah pekerja dalam work centre tersebut. Khusus untuk jam kerja lembur, jam kerja untuk memproduksi sejumlah unit produk yang dihasilkan akan dihitung dengan hitungan jam, bukan menit, jadi jika hanya bekerja selama 1 jam 31 menit, maka gaji yang dibayarkan adalah selama 2 jam. Kebutuhan listrik / unit merupakan biaya listrik yang harus dikeluarkan untuk memproduksi satu unit produk, biaya ini didapat dari biaya listrik yang dikeluarkan saat berproduksi secara penuh, dibagi maksimum kapasitas masing-masing *work centre*. Sedangkan maksimum lembur, merupakan jumlah unit maksimum yang dapat dihasilkan oleh setiap *work centre* saat lembur.

3.4.5. Jadwal induk Produksi (*Mater Production Scheduling*)

Tabel di bawah ini merupakan data historis dari jadwal induk produksi PT X pada bulan November 2007 untuk setiap *work centre*

Tabel 3. 8 Jadwal induk Produksi (*Mater Production Scheduling*)

to Feb 08								Nop-07			
Item#	Product	uom	Batch size	Buffer stock (month)	Stock 30 Oct 07	WIP 30 Oct 07	Month coverage	Forecast	Delivery Plan to WH	Est end stock	Months coverage
AAAA	Product A	Bottle	24.000	1	15.000	-	0,9	16.000	24.000	23.000	1,5
BBBB	Product B	Box/30	12.800	1,5	7.500	3.000	0,9	11.500	12.800	11.800	2,0
CCCC	Product C	Box/100	2.400	2,0	8.500	-	3,4	2.500	-	6.000	2,4
DDDD	Product D	Box/200	1.200	1,0	4.500	5.000	6,3	1.500	-	8.000	1,2
EEEE	Product E	Tin/1000	2.560	3,0	250	500	0,3	3.000	5.120	2.870	6,4
FFFF	Product F	Bottle	19.200	1	7.500	6.000	0,8	17.000	19.200	15.700	1,6

(Sumber: PT X)

Pada tabel diatas dapat kita lihat bahwa setiap *work centre* memiliki kapasitas dan *demand/ forecasting* yang berbeda-beda. Untuk *work centre* 1, 2 dan 6 diharapkan mampu menghasilkan produk diatas 10.000 Batch, masing-masing 16.000, 11.500 dan 17.000, sedangkan *work centre* 3,4 dan 5 hanya diperlukan untuk memproduksi sebesar 2.500, 1.500 dan 3.000 Batch. Data ini diolah untuk mengetahui apakah semua *work centre* yang ada akan bekerja dan selesai dalam

Universitas Indonesia

waktu yang hampir sama, dimana diharapkan nantinya kejadian bahwa salah satu *work centre* telah mampu menghasilkan sejumlah produk yang diminta, namun ada beberapa *work centre* yang masih bekerja. Hal ini akan menyebabkan kurang efektifnya biaya listrik pabrik yang dikeluarkan. Oleh karena itu diperlukan sebuah sistem yang menyediakan perencanaan produksi harian setiap *work center* yang seoptimal mungkin agar biaya listrik pabrik yang dikeluarkan dapat dipergunakan dengan seefisien mungkin.

3.4.6. Penjadwalan Produksi Harian

Jadwal produksi harian yang dikeluarkan oleh perusahaan biasanya disajikan dalam sebuah tabel yang memperlihatkan rencana jadwal produksi selama satu periode produksi telah diolah sebelumnya dari perintah produksi yang dikeluarkan dari penjadwalan induk produksi. Tabel 3.9 dibawah merupakan salah satu contoh rencana jadwal produksi harian dan aktual produksi yang dihasilkan pada salah satu *work centre* di PT X.

Dimana pada tabel 3.9 tersebut dapat dilihat bahwa PT X untuk periode produksi bulan tersebut perintah produksi yang dikeluarkan oleh jadwal induk produksi yaitu sebesar 251.350 unit dapat dipenuhi oleh *work centre* dalam waktu yang tersedia yaitu 21 hari dengan menggunakan *shift* kerja normal. Dimana kapasitas produksi maksimum per *shift* pada *work centre* ini adalah sebanyak 6000 unit, dan maksimum produksi lembur per hari nya adalah sebesar 3000 unit. Sedangkan actual produksi merupakan jumlah unit produksi yang mampu dihasilkan oleh *work centre* tersebut dalam satu hari kerja, yaitu penjumlahan dari *shift* 1, *shift* 2 dan *overtime* (kalau ada).

Tabel 3. 9 Jadwal harian produksi *work centre A*

Perintah Produksi :

251350 Unit

Hari	Shift 1 (23.00 08.00)	Shift 2 (08.00 17.00)	Overtime (17.00-21.00)	Total Aktual Produksi
1	6000	6000	-	12000
2	6000	6000	-	12000
3	6000	6000	-	12000
4	6000	6000	-	12000
5	6000	6000	-	12000
6	6000	6000	-	12000
7	5800	5900	-	11700
8	6000	6000	-	12000
9	6000	6000	-	12000
10	6000	6000	-	12000
11	6000	6000	-	12000
12	6000	6000	-	12000
13	6000	6000	-	12000
14	6000	6000	-	12000
15	6000	6000	-	12000
16	6000	6000	-	12000
17	6000	6000	-	12000
18	6000	6000	-	12000
19	5900	5900	-	11800
20	5950	5900	-	11850
21	6000	6000	-	12000

(Sumber:PT X)

4. PENGOLAHAN DATA DAN ANALISIS

4.1. Penyusunan Algoritma

Proses pengolahan data pada perancangan sistem ini dilakukan secara komputerisasi dengan menggunakan bahasa pemrograman *Microsoft Visual Basic* dan *Microsoft Access*. Penggunaan bahasa pemrograman *Microsoft Visual Basic* ini didasarkan pada kemampuan *Microsoft Visual Basic* untuk digunakan dalam penulisan program berbasis *Microsoft Windows*. Bahasa pemrograman *Microsoft Visual Basic* menyediakan semua sarana yang dibutuhkan oleh para *programmer* untuk membangun sebuah program yang kompleks dengan prosedur yang sederhana serta dapat diaplikasikan secara efisien dan efektif. Program yang dihasilkan dari bahasa pemrograman *Microsoft Visual Basic* tersebut akan diintegrasikan dengan *Microsoft Access* yang dikenal sebagai program aplikasi *database*. *Database* merupakan sekumpulan informasi yang terorganisir dan disimpan secara elektronis dalam sebuah *file*. *Database* pada *Microsoft Access* merupakan *database* relasional karena bisa menyimpan informasi dalam banyak tabel yang saling berhubungan dan informasi yang sama tidak perlu dimasukkan lebih dari satu kali¹³. *Microsoft Access* berfungsi sebagai penyimpan (*record*) data historis perusahaan yang jumlah dan jenisnya sangat beraneka ragam yang kemudian informasi tersebut akan dimanfaatkan oleh *user* melalui penggunaan program yang dibentuk dengan bahasa pemrograman *Microsoft Visual Basic*. *User* dapat menemukan dan menggunakan informasi sesuai dengan kebutuhan mereka secara efisien dan efektif dengan adanya integrasi antara *Microsoft Visual Basic* dan *Microsoft Access* ini.

Dalam perancangan sistem ini, ketiga jenis perencanaan dalam manajemen produksi yaitu; perencanaan sumber daya produksi, pengendalian rantai produksi, dan penjadwalan induk produksi menjadi bagian yang saling berkaitan dan terintegrasi satu sama lainnya. Informasi yang diberikan oleh penjadwalan induk produksi akan *direcord* dan digunakan oleh *user* perencanaan sumber daya produksi dan pengendalian rantai produksi yang selanjutnya diolah untuk

¹³ Online Training Solutions, Inc., *Step by Step Microsoft Access*.2001.hal 3

menghasilkan informasi berupa *ouput* sesuai dengan tujuan masing-masing perencanaan tersebut. Informasi tersebut akan selalu di-*record* dan di-*update* dalam periode tertentu sehingga *user* dapat mengambil keputusan secara efisien dan efektif sesuai dengan tujuan perancangan sistem ini. Untuk perencanaan sumber daya produksi, sistem yang di-*develop* dengan *Microsoft Visual Basic* akan melakukan perhitungan jumlah pekerja dan jam lembur optimal yang diperlukan untuk memproduksi jumlah produk sesuai dengan perintah produksi yang terdapat dalam penjadwalan induk produksi. Dan *output* yang dihasilkan akan disimpan dalam *database* yang terintegrasi dengan *Microsoft Access* dalam jangka waktu tertentu, sehingga data historis tersebut dapat digunakan oleh *user* untuk melakukan perhitungan perencanaan untuk periode berikutnya.

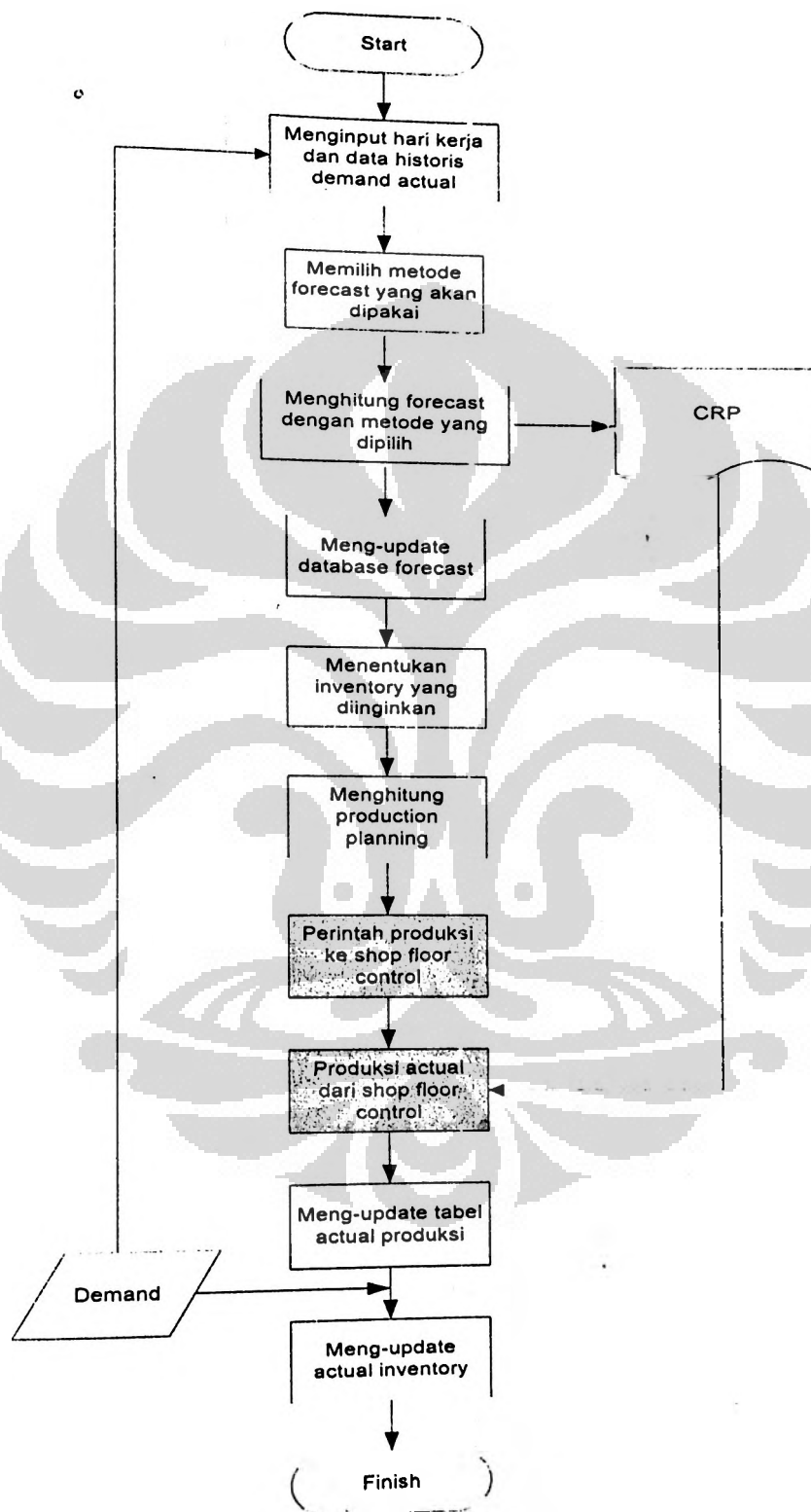
Untuk mempermudah perancangan sistem perencanaan kapasitas sumber daya produksi ini, maka terlebih dahulu dilihat hubungan saling keterkaitan antara perencanaan kapasitas sumber daya produksi dengan pengendalian rantai produksi dan penjadwalan induk produksi. Gambar 4.1 berikut adalah *flowchart* yang menunjukkan hubungan saling keterkaitan dari ketiga sistem tersebut.

Pada *flowchart* tersebut dapat dilihat bahwa jadwal induk produksi akan memberikan perintah produksi untuk satu periode produksi ke pengendalian rantai produksi untuk diolah selanjutnya dengan perubahan yang terjadi pada perencanaan kapasitas, menjadi penjadwalan harian di setiap *work centre* yang ada. Kemudian sistem akan memberikan *feedback* berupa produksi actual dan permintaan actual ke jadwal induk produksi.

Sedangkan aktivitas utama pada sistem pengendalian rantai produksi ini dimulai dengan melakukan *input* data variable, yang kemudian diolah menjadi variable biaya dan kapasitas yang akan berpengaruh terhadap *output* yang akan dihasilkan. Sebelum dilakukan perhitungan penjadwalan harian yang efisien, sistem akan menunggu kebijakan yang akan diambil oleh perencana kapasitas apakah akan menambah jumlah pekerja ataupun jumlah mesin yang akan beroperasi, dikonfirmasi oleh perencana kapasitas, sistem akan melaksanakan langkah terakhir dalam pengolahan data yang di-*input* yaitu menghitung jumlah produksi harian yang harus dipenuhi oleh setiap *work centre*. Secara umum aktivitas utama

Universitas Indonesia

dalam sistem pengendalian rantai produksi ini dapat dilihat pada gambar 4.2 berikut:



Gambar 4. 1 Flowchart Hubungan dan interaksi antara ketiga sistem

4.1.1. Langkah – Langkah Penyusunan Algoritma Sistem

Berikut adalah beberapa langkah yang digunakan dalam perancangan sistem perencanaan kapasitas sumber daya produksi ini. Secara umum langkah - langkah tersebut dimulai dari pengidentifikasian data *input*, pembuatan algoritma *visual basic*-nya sebagai langkah awal dalam perancangan sistem hingga verifikasi dan validasi dari sistem tersebut sebagai cara pengujian *reliability* sistem apakah algoritma sistem telah sesuai perhitungannya.

4.1.1.1. Pembuatan Draft Sistem

Pada tahap pembuatan *draft* sistem ini, terdapat beberapa hal yang penting dilakukan terutama identifikasi terhadap data yang berkaitan dengan *input*, proses, dan *output* dari sistem yang dirancang.

1. Identifikasi Data yang Akan di-Input ke Dalam Sistem

Dalam perancangan sistem pengendalian lintai produksi ini, data awal yang di-*input* ke dalam sistem adalah data perintah produksi yang berasal dari MPS, jumlah mesin per *work centre*, jumlah pekerja per *work centre*, upah pekerja per jam, total biaya listrik per *work centre*, dan waktu proses untuk menghasilkan 1 unit produk per *work centre*.

Data-data ini termasuk data basic dan historis dari perusahaan sehingga *user* dapat memasukkan data tersebut ke dalam sistem sesuai dengan *field*-nya. Dimana setiap data tersebut hanya di-*Input* sekali saja, kecuali data perintah produksi yang dikeluarkan MPS, selama tidak terjadi perubahan-perubahan yang signifikan, seperti jumlah mesin dan jumlah pekerja, yang akan menyebabkan perubahan waktu proses secara keseluruhan, untuk data total biaya listrik setiap *work centre* data yang digunakan adalah data saat *work centre* tersebut berproduksi dalam kapasitas maksimum, data ini dapat dilihat pada data historis perusahaan, karena data ini dijadikan sebagai patokan untuk pengolahan data selanjutnya, sedangkan data perintah produksi di-*Input* sekali setiap periode, karena *demand* yang akan keluar belum tentu sama untuk setiap periode, artinya jika satu periode produksi adalah satu bulan, maka data ini di-*Input* setiap awal bulan setiap periode, Data-data ini selanjutnya akan diolah dalam sistem menjadi informasi yang diperlukan

untuk memperoleh *output*. Dimana Jenis-jenis data tersebut dapat dilihat pada table 4.1 dibawah ini.

Tabel 4.1 Jenis Data Input dan Sumber nya

No	Jenis Data Input	Sumber	Satuan
1	Perintah Produksi	MPS	Unit/bulan
2	Jumlah Mesin	<i>Work Centre</i>	Unit
3	Jumlah Pekerja	<i>Work Centre</i>	Unit
4	Upah Pekerja	Data Umum	Rupiah / Jam
5	Biaya Listrik	<i>Work Centre</i>	Rupiah
6	Waktu Proses	<i>Work Centre</i>	Menit

2. Identifikasi Data yang Menjadi Informasi Untuk Selanjutnya Diolah Menjadi *Output*

Adapun data-data yang akan menjadi informasi tambahan dalam penyusunan sistem ini didapat dari perhitungan data-data input utama diatas yang selanjutnya berguna untuk diolah selanjutnya menjadi *output* utama. Adapun data-data ini antara lain kapasitas produksi per jam setiap *work centre*, kapasitas maksimum setiap *work centre* per shift, kapasitas maksimum produksi saat lembur setiap *work centre*, biaya pekerja per jam setiap *work centre* dan biaya listrik per unit produk yang dihasilkan di setiap *work centre*. Data-data inilah yang selanjutnya akan diolah oleh sistem untuk mendapatkan output yang diinginkan *user*. Dimana jenis-jenis data tersebut beserta data-data yang mempengaruhi data tersebut dapat dilihat pada table 4.2 dibawah ini

Tabel 4.2 Jenis Data Informasi

No	Jenis Data Informasi	Sumber
1	Upah Pekerja / Jam / <i>Work Centre</i>	Jumlah pekerja dan Upah Pekerja / Jam
2	Kapasitas Produksi Maksimum / <i>Work Centre</i>	Jam kerja / <i>Shift</i> dan waktu proses Setiap <i>Work Centre</i>
3	Biaya Listrik / Unit Produk / <i>Work Centre</i>	Biaya listrik / <i>Work Centre</i> dan Kapasitas Produksi Maksimum

3. Mengidentifikasi jenis data *output* yang akan dihasilkan

Sebagaimana tujuan penelitian ini diatas, dimana *output* yang diharapkan adalah sebuah penjadwalan produksi harian yang efektif dan efisien dengan biaya yang paling minimal, untuk memenuhi perintah produksi yang dikeluarkan oleh MPS. Sistem yang dirancang ini diharapkan mampu melakukan perhitungan untuk

menghindari terjadinya ketidakseimbangan antar *work centre* yang ada, karena perintah produksi yang dikeluarkan untuk masing-masing *work centre* tidak selalu sesuai dengan kapasitas *work centre* itu sendiri. Sehingga biaya listrik pabrik yang dikeluarkan adalah seoptimal mungkin dengan kondisi yang seperti tersebut. Selain memperhitungkan total biaya listrik yang dikeluarkan, sistem ini juga akan memberikan informasi mengenai total biaya pekerja yang akan dikeluarkan untuk memproduksi sejumlah produk yang dihasilkan.

4.1.2 Pengelompokkan Jenis Data

Setelah pengidentifikasian *draft* sistem yang berhubungan dengan data *input*, proses, dan *output* sistem dilakukan, maka langkah selanjutnya adalah melakukan pengelompokkan jenis data. Pengelompokkan jenis data ini perlu dilakukan karena data-data yang digunakan di setiap *work centre* berbeda satu dengan lainnya, terutama pada data-data *input* awal seperti jumlah pekerja di setiap *work centre*, kapasitas maksimum produksi setiap *work centre*, jumlah mesin setiap *work centre*, dan waktu proses produksi setiap *work centre*. Oleh karena itu, pada sistem yang akan dirancang nantinya perlu menampilkan *field* data yang berbeda untuk setiap *work centre*. Sehingga nantinya *user* yang akan menggunakan sistem ini akan mudah dalam melakukan *input* data-data yang diperlukan. Tabel 4.3 berikut ini merupakan hasil dari pengelompokkan data yang dilakukan berdasarkan jenis *work centre* yang ada pada PT X ini.

Tabel 4.3 Pengelompokkan Jenis Data

	Work Centre					
Work Centre	A	B	C	D	E	F
Jumlah Pekerja (Orang)						
Jumlah Mesin (Unit)						
Waktu proses (Menit)						
Maksimum Produksi / shift						

4.1.3 Pengidentifikasian dan Pemvisualisasian *Interface* Sistem, Termasuk *Field-Field* yang Akan Ditampilkan

Dalam pengidentifikasian dan pemvisualisasian *interface* sistem yang akan dirancang, hal yang perlu diperhatikan adalah sistem dapat menampilkan *field-field* yang menyediakan informasi penting bagi *user*. Baik itu informasi untuk

Universitas Indonesia

input data, proses pengolahan data, maupun *output* yang menjadi rekomendasi bagi *user* dalam mengambil keputusan. Dalam hal ini, *interface* sistem ditampilkan dalam bentuk *tab* sehingga lebih menghemat *work space* dan mempermudah proses *input* data. Berikut adalah tampilan dari *interface* sistem yang akan dirancang:

	Work Center A	Work Center B	Work Center C	Work Center D	Work Center E	Work Center F
Process time (min)						
Electrical cost/day (Rp)						
Number of employee						

Number of Days:

Wages /person/hr:

Save

Gambar 4.3 Interface untuk Input General Data

Gambar 4.1 di atas merupakan *interface* untuk *input general data*. Dimana *general data* ini diletakkan pada form MPS, hal ini karena dalam rangkaian sistem ini, MPS merupakan tahap awal dalam pemrosesan penjadwalan produksi pabrik, sehingga, sebelum menghitung berapa jumlah produksi yang harus diproduksi, maka *user* di MPS harus menginput *general data* ini terlebih dahulu. *User* dapat memasukkan data sesuai dengan *work centre* masing-masing pada kolom yang telah tersedia. Pada sistem disediakan kolom data hanya untuk enam *work centre* dan setiap memasukkan data, *user* harus melakukan penyimpanan (*save*), sehingga data dapat disimpan dalam *database* sebagai data historis perusahaan.

Selain itu sebuah data *input* lagi yaitu perintah produksi dari MPS, dimana perintah produksi ini muncul setelah MPS melakukan perhitungan terhadap *forecast* permintaan untuk periode selanjutnya. Gambar 4.2 berikut merupakan tampilan dari perintah produksi

General Data Moving Average Exponential Smoothing Winter's Method MPS

MPS A MPS B MPS C MPS D MPS E MPS F

FORECAST A

Desired inv. 30000

Months	Forecast	Production	Demand	Actual Production	Desired Inventory	Actual Inventory
1	232733	275000	220000	220000	29000	19200
2	220067				30000	
3	220300					
4	195333					
5	220633					
6	222567					
7	239567					
8	234567					
9	224700					
10	219733					
11	214200					
12	201033					

Calculate Production Order

Gambar 4. 4 Interface Input Data Perintah Produksi

Kemudian dari *general data* diatas maka dibutuhkan sebuah *field* untuk melihat parameter atau batasan yang akan digunakan untuk memperoleh hasil yang diinginkan. Parameter ini akan menjadi informasi bagi *user* dalam menentukan kebijakan produksi yang diambil. Gambar 4.3 Berikut ini merupakan tampilan dari parameter tersebut

General Informations Daily Production Plan

Parameters

Work Center	Production Max/ shift	Overtime Max	Elect. Cost/ unit	Wages/ W.C/ hr	Production Max/ hr
A					
B					
C					
D					
E					
F					

Gambar 4. 5 Interface untuk General Information dalam Sistem

Setelah semua jenis data yang dibutuhkan telah diinput maka *user* dapat langsung masuk ke *tab Daily Production* untuk melakukan proses perhitungan jadwal produksi harian yang diinginkan. Gambar 4.4 berikut ini merupakan tampilan dari proses perhitungan jadwal produksi harian di setiap *work centre*

General Informations

Daily Production Plan

Input

Today: 0

	Work Center A	Work Center B	Work Center C	Work Center D	Work Center E	Work Center F
Actual Production						
Actual demand						

Schedule the Production

Save Actual Production

Results

Product A | Product B | Product C | Product D | Product E | Product F

Product A

Gambar 4. 6 Interface Pengolahan Data Penjadwalan Produksi Harian

4.1.4. Memformulasikan Rumus yang Akan Dikonversi ke Dalam Program yang Akan Dirancang

Dalam sistem pengendalian rantai produksi ini, fungsi tujuan yang ingin dicapai adalah memperoleh penjadwalan produksi harian dengan penggunaan biaya listrik pabrik yang efektif dan efisien di setiap *work centre* yang ada. Untuk memperoleh tujuan tersebut, maka dalam pengolahan data-data *input* yang ada, terdapat beberapa langkah perhitungan yang dibuat sebelumnya pada sebuah *work sheet* yang nantinya akan dikonversikan menjadi sebuah formula logika bagi sistem yang akan dirancang. Adapun beberapa langkah perhitungan tersebut antara lain dapat dijabarkan sebagai berikut:

1. Meng-*input* data awal di setiap *work centre* berupa data-data umum dalam sebuah proses produksi di sebuah pabrik; antara lain; waktu proses untuk menghasilkan satu unit produk, jumlah mesin, jumlah pekerja, upah pekerja per jam kerja, jumlah jam lembur maksimal per hari, jumlah jam kerja per *shift*, jumlah hari produksi dalam satu periode produksi, dan perintah produksi yang akan dikeluarkan oleh jadwal induk produksi.

Universitas Indonesia

Data-data tersebut merupakan data standar yang bisa langsung *diinput* berdasarkan kondisi yang ada pada masing-masing *work centre* di pabrik tersebut. Dimana jenis data ini dapat dilihat pada tabel 4.1 diatas yaitu mengenai jenis data input dan sumber nya.

2. Selanjutnya data-data tersebut akan disimpan dalam sebuah *database* produksi yang nantinya akan langsung diolah untuk memperoleh *output* yang diinginkan. Namun sebelum *output* utama dihasilkan diperlukan beberapa perhitungan terlebih dahulu dimana hasil dari perhitungan ini akan menjadi beberapa data, antara lain; kapasitas maksimum produksi setiap *work centre* per jam, kapasitas maksimum produksi setiap *work centre* per *shift*, kapasitas maksimum produksi saat lembur, biaya listrik per unit produk dan sebuah variabel penalti yang nantinya akan menjadi batasan dalam menghitung berapa produksi harian yang efisien. Berikut ini merupakan perhitungan dalam menghasilkan jenis data ini:

$$\text{Kapasitas maksimum produksi per work centre per jam} = \frac{\text{Qty Maksimal}}{\text{Waktu proses}}$$

$$\begin{aligned} \text{Kapasitas maksimum produksi work centre per shift} \\ = \text{Kapasitas maksimum produksi per work centre per jam} \times \text{Jumlah kerja per shift} \end{aligned}$$

$$\begin{aligned} \text{Kapasitas maksimum produksi per hari} \\ = \text{Kapasitas maksimum per shift} \times \text{Jumlah shift per hari} \end{aligned}$$

$$\text{Biaya listrik per unit produk} = \frac{\text{Biaya listrik per WC}}{\text{Kapasitas maksimum produksi per hari}}$$

$$\text{Biaya listrik per unit produk} = \frac{\text{Biaya listrik per WC}}{\text{Kapasitas maksimum produksi per hari}}$$

Sedangkan untuk perhitungan variabel penalti dapat dilihat pada tabel 4.4 berikut

Tabel 4.4 Perhitungan Penalty Produksi

Penalti		
Batas Atas	Batas Bawah	Biaya Listrik
100% A	90% A	0% B
90% A	80% A	10% B
80% A	70% A	20% B
70% A	60% A	30% B
60% A	50% A	40% B
50% A	40% A	50% B
40% A	30% A	60% B
30% A	20% A	70% B
20% A	10% A	80% B
10% A	0% A	90% B

Adapun penggunaan penalti pada sistem ini bertujuan untuk memaksimalkan penggunaan biaya listrik yang digunakan di setiap *work centre*, dengan kata lain, semakin besar tingkat produksinya (mendekati kapasitas maksimum produksi), maka semakin kecil pula penalti yang akan didapat. Hal ini bertujuan untuk memaksimalkan penggunaan biaya listrik yang dikeluarkan oleh pabrik oleh ke enam *work centre* yang ada. Logika perhitungan untuk penalti produksi seperti tabel 4.4 diatas adalah jika *work centre* tersebut berproduksi antara persentase batas atas dan batas bawah kapasitas produksi maksimum per *shift*, maka yang penalti yang dikenakan adalah sebesar selisih antara produksi yang direncanakan dengan batas bawah nya kemudian dikali dengan biaya listrik per unit produk nya.

Sebagai contoh, jika sebuah *work centre* berproduksi sebesar 87% dari kapasitas maksimum produksinya, berarti berada pada batas atas 90% dan batas bawah 80%, sehingga biaya listrik penaltinya adalah sebesar 20%, 100% dikurangi batas bawah kapasitas maksimum produksi, maka biaya penaltinya adalah sebesar 87% dikurang 80% dikali biaya listrik penaltinya.

3. Setelah semua data diatas disimpan dalam database variabel yang ada, maka akan dilakukan random data untuk masing-masing *shift* disetiap

work centre yang ada, dengan menggunakan batasan data yang ada pada langkah ke dua diatas, yaitu maksimum kapasitas produksi per *shift* dan maksimum kapasitas produksi saat lembur.

4.1.5. Penyusunan Sistem

Penyusunan sistem dilakukan dengan menggunakan bahasa pemrograman *Microsoft Visual Basic* yang dihubungkan ke *Microsoft Access* menggunakan bahasa SQL (*Structured Query Language*). Dalam penulisan kode program ini juga konversi rumus dari draft *Excel* sebelumnya ke dalam bahasa *Visual Basic*. Setiap saat, penulisan kode ini selalu diikuti oleh *debuging* yaitu pengecekan ada tidaknya kesalahan (*errors*) yang dilakukan secara kontinu. Jadi dalam penyusunan sistem tersebut menggunakan beberapa *code* pemrograman sehingga menghasilkan suatu sistem yang dapat melakukan perhitungan jadwal produksi harian pada sistem pengendalian rantai produksi dan terintegrasi dengan MPS dan CRP. Berikut adalah beberapa tahapan dalam penyusunan *coding* nya:

1 *Input Data*

Input data ini terbagi menjadi dua jenis, yaitu langsung diinput oleh *user*, dan perintah produksi yang harus dihitung terlebih dahulu.

2 *General Information dan Parameter yang digunakan*

Berisi mengenai proses untuk menampilkan parameter dan *general information* yang dibutuhkan dari database variabel dalam pengolahan penjadwalan produksi harian.

3 *Pengolahan Data Penjadwalan Produksi Harian*

Merupakan *coding* untuk melakukan proses data untuk diolah menjadi penjadwalan produksi harian kemudian menyimpannya ke dalam database yang telah disiapkan, yang kemudian akan langsung memberikan informasi berupa jumlah aktual produksi dan permintaan aktual pada periode produksi tersebut

4.2. Validasi dan Verifikasi Sistem

Sebelum dilakukan penyelesaian terhadap perancangan sistem pengendalian rantai produksi pada PT.X ini, maka terlebih dahulu dilakukan verifikasi dan validasi program. Tujuan utama dari proses verifikasi dan validasi program ini adalah untuk melihat sejauh mana keakuratan sistem dalam menghasilkan tujuan yang ingin dicapai dan menemukan kekurangan dalam sebuah sistem yang telah dibuat sehingga dapat dilakukan perbaikan dan memperkirakan apakah sistem berguna sesuai spesifikasi dan dapat digunakan dalam situasi operasional.

Tahap verifikasi (*are we building the product right*) merupakan tahap melihat kesesuaian antara model program yang didapat dengan konseptual model yang telah dibuat atau dengan kata lain apakah sistem yang telah dibuat sesuai dengan spesifikasinya. Parameter model program dikatakan telah terverifikasi apabila telah berjalan sesuai konseptual model, dimana proses-proses dan data-data yang ada dan dihasilkan pada tahap penyusunan draft sistem telah terpenuhi pada sistem ini. Parameter yang harus ada antara lain sebagai berikut:

- Adanya informasi mengenai data-data input yang ada
- Adanya informasi mengenai jumlah perintah produksi yang diberikan oleh jadwal induk produksi
- Adanya informasi perubahan data sumber daya atau kapasitas yang diberikan oleh perencanaan kapasitas.

Sebuah program telah terverifikasi jika pada saat program dijalankan kemudian dilakukan perubahan terhadap parameter-parameter tersebut, maka *output* yang dihasilkan juga berubah sesuai dengan hasil yang seharusnya. Pada sistem pengendalian rantai produksi yang telah dirancang dengan menggunakan *Microsoft Visual Basic* ini, secara umum sistem telah terverifikasi. Sistem yang telah dirancang ini telah sesuai dengan spesifikasi tujuan penggunaannya, yaitu mampu melakukan perhitungan penjadwalan produksi harian di setiap *work centre* dengan penggunaan biaya yang paling efisien atau dengan kata lain perhitungan yang menghasilkan jumlah produksi harian dengan penalti yang terkecil dalam satu periode produksinya.

Sistem mulai bekerja setelah semua *input* data yang diperlukan telah disimpan dalam *database* dan perencana kapasitas telah memutuskan untuk melakukan perubahan terhadap jumlah pekerja untuk memenuhi perintah produksi dari jadwal induk produksi. Kemudian sistem pengendalian rantai produksi ini akan mengolah data-data tersebut menjadi jadwal produksi harian di setiap *work centre*.

Setelah mendukung verifikasi terhadap program, maka tahap pengujian sistem selanjutnya adalah melakukan validasi program. Validasi (*Are we building the right product*) terhadap program yang telah dibuat dilakukan dengan memasukkan data *dummy*. Tujuannya dari validasi ini adalah bahwa program telah berjalan sesuai dengan fungsinya seperti yang disyaratkan oleh *user* dan menghasilkan *output* yang sesuai dengan yang seharusnya dan penyimpangan yang mungkin terjadi masih dalam batasan yang seharusnya. Adapun langkah-langkah validasi program adalah membandingkan perhitungan manual data *dummy* dengan yang dihasilkan oleh sistem.

Berikut ini merupakan proses dari verifikasi dan validasi program, dengan data *dummy* yang digunakan adalah sebagai berikut:

1. Penentuan asumsi-asumsi awal

Berikut ini merupakan asumsi-asumsi awal yang digunakan:

- 1 Jam = 60 menit
- 1 *shift* = 8 jam
- Jam lembur maksimum = 4 jam
- Jumlah *shift* per hari = 2 *shift*
- Jumlah hari operasi produksi = 21 hari

2. *Input* data umum setiap *work centre*, dalam hal ini hanya dilakukan untuk satu *work centre*. Pada perintah produksi akan digunakan dua jenis data, yaitu untuk jumlah perintah produksi yang melebihi kapasitas produksi *shift* normal, atau dengan kata lain harus menggunakan jam lembur dan perintah produksi yang bisa dipenuhi dengan menggunakan *shift* normal.

Tabel 4.5 berikut ini merupakan data umum dari salah satu *work centre* yang terdiri dari data waktu proses, biaya listrik setiap *work centre* ditambah biaya AC

Universitas Indonesia

pusat yang digunakan, jumlah mesin, jumlah pekerja, upah pekerja per jam kerja dan perintah produksi yang dikeluarkan oleh jadwal induk produksi.

Tabel 4.5 Data Umum *Work Centre* (Lembur)

Data	Nilai	Satuan
Waktu proses	0,08	Menit
Biaya listrik	800.000	Rupiah
Jumlah mesin	40	Unit
Jumlah pekerja	20	Orang
Upah pekerja	15.000	Rupiah/Orang/Jam
Perintah prod	275.000	Unit

Tabel 4.6 Data Umum *Work Centre* (Normal)

Data	Nilai	Satuan
Waktu proses	0,08	Menit
Biaya listrik	800.000	Rupiah
Jumlah mesin	40	Unit
Jumlah pekerja	20	Orang
Upah pekerja	15.000	Rupiah/Orang/Jam
Perintah prod	230.000	Unit

3. Dari data input diatas didapat beberapa data tambahan yang bersifat informasi yang berguna untuk pengolahan data selanjutnya. Data-data dengan perhitungan manual, hasilnya dapat dilihat pada tabel 4.6 berikut

Tabel 4.7 Data Informasi Tambahan (Lembur)

Data	Nilai	Satuan
kapasitas produksi per jam	750	Unit
kapasitas maksimum / shift	6.000	Unit
Kapasitas lembur	3.000	Unit
biaya pekerja per jam /WC	300.000	Rupiah
biaya listrik per unit	81	Rupiah

Semua data diatas merupakan data yang diasumsikan sebagai data akhir yang telah ditetapkan oleh perencana kapasitas sebelumnya.

4. Dari data informasi tambahan diatas, maka dibuatlah batasan variabel penalti yang akan digunakan dalam menentukan jumlah produyksi harian yang akan digunakan. Data variabel penalti ini dapat dilihat pada tabel 4.7 berikut.

Tabel 4.8 Data Variabel Penalty

Penalti		
Batas Atas	Batas Bawah	Biaya Listrik
5400	4800	80,56
4800	4200	161,11
4200	3600	241,67
3600	3000	322,22
3000	2400	402,78
2400	1800	483,33
1800	1200	563,89
1200	600	644,44
600	0	725,00

5. Perhitungan penjadwalan produksi harian

Dalam menghitung rencana produksi harian ini, langsung menggunakan sistem yang telah dirancang, dimana dalam perhitungan ini jumlah produksi harian nya adalah berdasarkan hasil random data di sistem ini. Adapun jumlah random yang dipakai adalah sebanyak 400 kali, 450 kali, 500 kali, dan 550 kali random.

Berikut ini akan diperlihatkan hasil dari masing-masing randoman data, dimana setiap random data dilakukan sebanyak dua kali.

Hasil perhitungan perintah produksi sebesar 275.000 unit, dengan 400 kali random dapat dilihat pada tabel 4.9 berikut:

Tabel 4.9 Hasil penjadwalan dengan 400 kali random

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	6000	6000	3000	16111,11111
2	6000	6000	3000	
3	6000	6000	3000	
4	6000	6000	3000	
5	6000	6000	3000	
6	6000	6000	3000	
7	6000	6000	3000	
8	6000	6000	3000	
9	6000	6000	2000	
10	6000	6000		
11	6000	6000		
12	6000	6000		
13	6000	6000		
14	6000	6000		
15	6000	6000		
16	6000	6000		
17	6000	6000		
18	6000	6000		
19	6000	6000		
20	6000	6000		
21	6000	6000		

Hasil perhitungan dengan 400 kali random yang dilakukan sebanyak dua kali, menghasilkan besar penalti yang sama yaitu sebesar Rp. 16.111,111.

Sedangkan dengan menggunakan 550 kali random, hasil yang didapat adalah seperti pada tabel 4.9 berikut:

Tabel 4.10 Hasil penjadwalan dengan 550 kali random

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	6000	6000	3000	16111,11111
2	6000	6000	3000	
3	6000	6000	3000	
4	6000	6000	3000	
5	6000	6000	3000	
6	6000	6000	3000	
7	6000	6000	3000	
8	6000	6000	2000	
9	6000	6000		
10	6000	6000		
11	6000	6000		
12	6000	6000		
13	6000	6000		
14	6000	6000		
15	6000	6000		
16	6000	6000		
17	6000	6000		
18	6000	6000		
19	6000	6000		
20	6000	6000		
21	6000	6000		

Ternyata hasil yang didapat sama persis dengan yang dihasilkan dengan menggunakan 400 kali random. Berarti Untuk perintah produksi yang melebihi kapasitas maksimum produksi normal, sistem ini sudah sesuai dengan prinsip kerja atau proses yang seharusnya, seperti yang akan dijelaskan pada cara kerja sistem, yaitu memenuhi jam kerja normal terlebih dahulu dan sisanya baru diperhitungkan pada saat lembur, dimana pada saat lembur pun juga menggunakan aturan yang sama, yaitu memenuhi kapasitas lembur per hari terlebih dahulu, dan sisanya dipenuhi pada hari selanjutnya.

Untuk perintah produksi yang bisa dipenuhi dengan *shift* normal yaitu sebesar 230.000 unit dengan 400 kali, 450 kali, 500 kali, dan 550 kali random adalah sebagai berikut :

Perintah produksi sebesar 230.000 unit dengan 400 kali random

Tabel 4.11 Hasil Penjadwalan dengan 400 kali random (1)

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5195	5828	-	320852,7778
2	5999	5862	-	
3	5683	5485	-	
4	5785	5426	-	
5	3749	5427	-	
6	5570	5886	-	
7	5170	5999	-	
8	5872	5458	-	
9	5917	5819	-	
10	5983	5613	-	
11	5331	3979	-	
12	4049	3819	-	
13	4673	5988	-	
14	3861	5544	-	
15	5973	5887	-	
16	5928	5957	-	
17	5971	5672	-	
18	5843	5502	-	
19	5904	3974	-	
20	5074	5887	-	
21	5597	5861	-	

Tabel 4.12 Hasil Penjadwalan dengan 400 kali random (2)

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5955	5907	-	313844,4445
2	5094	5924	-	
3	5367	5511	-	
4	5580	5598	-	
5	4939	5425	-	
6	5438	5855	-	
7	5997	4656	-	
8	5911	5224	-	
9	5762	4673	-	
10	4843	5537	-	
11	5025	5067	-	
12	5995	5990	-	
13	4801	5185	-	
14	5153	5513	-	
15	5746	5682	-	
16	5954	5911	-	
17	4630	5540	-	
18	5666	5240	-	
19	5722	5663	-	
20	5732	4870	-	
21	5724	5995	-	

Dari kedua hasil perhitungan diatas, ternyata penalti yang paling kecil dihasilkan saat *run* yang kedua yaitu sebesar Rp. 313.844,4445. Sedangkan pada saat *run* yang pertama hasil yang didapat adalah sebesar Rp. 320852,777788842. Namun selisih antara keduanya masih sangat kecil, yaitu sebesar Rp. 7008,33333357499 atau sekitar 2.23%

Sedangkan dengan 450 kali random hasil yang didapat adalah sebagai berikut

Tabel 4.13 Hasil Penjadwalan dengan 450 kali random (1)

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5210	5250	-	286697,2222
2	5184	4951	-	
3	5907	5849	-	
4	4855	5964	-	
5	5230	5513	-	
6	4668	4882	-	
7	5635	4994	-	
8	5990	5720	-	
9	5309	5865	-	
10	5905	5947	-	
11	5566	5991	-	
12	5881	5549	-	
13	5539	5273	-	
14	5240	4540	-	
15	5998	5999	-	
16	5678	5582	-	
17	5983	4770	-	
18	5753	5486	-	
19	5224	5565	-	
20	5187	5608	-	
21	4797	5963	-	

Tabel 4.14 Hasil Penjadwalan dengan 450 kali random (2)

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5737	5913	-	292630,5556
2	5249	5338	-	
3	5046	4303	-	
4	4576	4928	-	
5	5886	4116	-	
6	5756	5985	-	
7	5900	5758	-	
8	5828	5801	-	
9	5595	5976	-	
10	5926	5936	-	
11	5749	5329	-	
12	4132	5800	-	
13	5441	5728	-	
14	5516	5595	-	
15	5680	5726	-	
16	5792	4719	-	
17	5826	5818	-	
18	5805	5899	-	
19	4362	5829	-	
20	5141	5413	-	
21	5622	5525	-	

Pada tabel 4.12 dan 4.13 ternyata hasil yang paling kecil didapat pada saat *run* yang pertama dengan nilai penalti sebesar Rp. 286697,222232109 sedangkan untuk *run* yang kedua menghasilkan penalti sebesar Rp. 292680,555564959. Namun selisih antara keduanya masih kecil yaitu sebesar 2.08%

Perhitungan dengan 500 kali random adalah sebagai berikut

Tabel 4.15 Hasil Penjadwalan dengan 500 kali random (1)

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5999	5418	-	259791,6667
2	5709	5904	-	
3	5511	4762	-	
4	5034	5386	-	
5	5984	4575	-	
6	5754	5632	-	
7	5976	5997	-	
8	5333	5306	-	
9	5979	5943	-	
10	5973	5563	-	
11	5856	5787	-	
12	4591	5248	-	
13	5899	5803	-	
14	5974	4137	-	
15	5787	5738	-	
16	5899	5077	-	
17	4463	5856	-	
18	5824	5986	-	
19	4821	5434	-	
20	5707	5989	-	
21	4292	4034	-	

Tabel 4.16 Hasil Penjadwalan dengan 500 kali random (2)

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5737	5913	-	272680,5556
2	5249	5338	-	
3	5046	4303	-	
4	4576	4928	-	
5	5986	4116	-	
6	5756	5985	-	
7	5900	5758	-	
8	5828	5801	-	
9	5595	5976	-	
10	5926	5936	-	
11	5749	5329	-	
12	4132	4790	-	
13	5441	5345	-	
14	5516	5768	-	
15	5680	5740	-	
16	5792	4619	-	
17	5972	5858	-	
18	5825	5999	-	
19	4362	5929	-	
20	5141	5713	-	
21	5922	5725	-	

Pada tabel 4.14 dan 4.15 diatas, dapat dilihat bahwa hasil *run* yang pertama menghasilkan nilai yang paling kecil yaitu sebesar Rp. 259.791,666675625, sedangkan pada *run* yang kedua, penalti yang dihasilkan adalah sebesar Rp. 272.680,555564959. Namun selisih antara keduanya tidak terlalu signifikan, yaitu sebesar 4.96%

Perhitungan dengan 550 kali random adalah sebagai berikut

Tabel 4.17 Hasil Penjadwalan dengan 550 kali random (1)

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5191	5667	-	272236,1111
2	5614	4582	-	
3	4273	4898	-	
4	5631	5391	-	
5	5908	4914	-	
6	5085	5779	-	
7	5266	5309	-	
8	5348	5985	-	
9	5042	5918	-	
10	5394	5992	-	
11	5891	5923	-	
12	5759	5743	-	
13	4871	5706	-	
14	5907	5407	-	
15	4924	5733	-	
16	5802	5111	-	
17	5966	5678	-	
18	5615	5829	-	
19	5927	5450	-	
20	5752	5997	-	
21	4326	5496	-	

Tabel 4.18 Hasil Penjadwalan dengan 550 kali random (2)

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5955	5907	-	286844,4445
2	5094	5924	-	
3	5367	5511	-	
4	5580	5598	-	
5	4939	5425	-	
6	5438	5855	-	
7	5997	4656	-	
8	5911	5224	-	
9	5762	4673	-	
10	4843	5537	-	
11	5025	5067	-	
12	5995	5990	-	
13	4801	5185	-	
14	5153	5513	-	
15	5746	5682	-	
16	5954	5911	-	
17	4630	5540	-	
18	5666	5240	-	
19	5722	5663	-	
20	5732	4870	-	
21	5724	5995	-	

Pada kedua tabel diatas, ternyata yang memberikan hasil yang paling kecil adalah pada saat *run* yang pertama yaitu sebesar Rp. 272.236.1111, sedangkan saat *run* yang kedua menghasilkan nilai penalti sebesar Rp. 286.844.4445. Namun selisih antara kedua nya tidak terlalu signifikan yaitu sebesar 6.03%.

Dari semua perhitungan diatas, didapatkan hasil penjadwalan produksi harian yang menghasilkan nilai penalti paling kecil adalah dengan 500 kali random yaitu sebesar Rp. 259.791,666675625. Selisih antara 500 dan 450 kali random adalah sebesar 10.35%, sedangkan antara 500 dan 400 kali adalah sebesar 20.80%, dan antara 500 dengan 550 kali adalah sebesar 4.79%

Maka penjadwalan harian yang optimal untuk perintah produksi sebesar 230.000 unit adalah seperti pada tabel berikut ini

Tabel 4.19 Penjadwalan Produksi Harian yang Optimal

Days	AShift1	AShift2	AShift3	ATotalPenalti
1	5999	5418	-	259791,6667
2	5709	5904	-	
3	5511	4762	-	
4	5034	5386	-	
5	5984	4575	-	
6	5754	5632	-	
7	5976	5997	-	
8	5333	5306	-	
9	5979	5943	-	
10	5973	5563	-	
11	5856	5787	-	
12	4591	5248	-	
13	5899	5803	-	
14	5974	4137	-	
15	5787	5738	-	
16	5899	5077	-	
17	4463	5856	-	
18	5824	5986	-	
19	4821	5434	-	
20	5707	5989	-	
21	4292	4094	-	

4.3 Cara Kerja Sistem

Sistem pengendalian rantai produksi untuk menghitung penjadwalan harian yang telah dirancang ini cukup *user friendly* dan tidak terlalu rumit dalam penggunaannya. Pada langkah awal, *user* perlu meng-input data awal seperti

waktu proses setiap *work centre*, biaya listrik per hari, jumlah pekerja normal di setiap *work centre*, serta upah pekerja normal per hari.

Gambar 4.7 berikut ini merupakan tampilan dari *input data* yang telah dilakukan

The screenshot displays the MPS software interface. The top menu bar includes 'Data' and 'Window'. Below the menu bar, there are tabs for 'General Data', 'Moving Average', 'Exponential Smoothing', 'Winter's Method', and 'MPS'. The 'General Data' tab is active, showing a form with the following data:

	Work Center A	Work Center B	Work Center C	Work Center D	Work Center E	Work Center F
Process time (min)	0.08	0.05	0.08	0.08	0.075	0.1
Electrical cost/day (Rp)	800000	500000	500000	650000	750000	350000
Number of employees	20	10	5	10	6	30

Below the table, there are fields for 'Number of Days' (21) and 'Wages /person/hr' (15000). A 'Save' button is visible on the right.

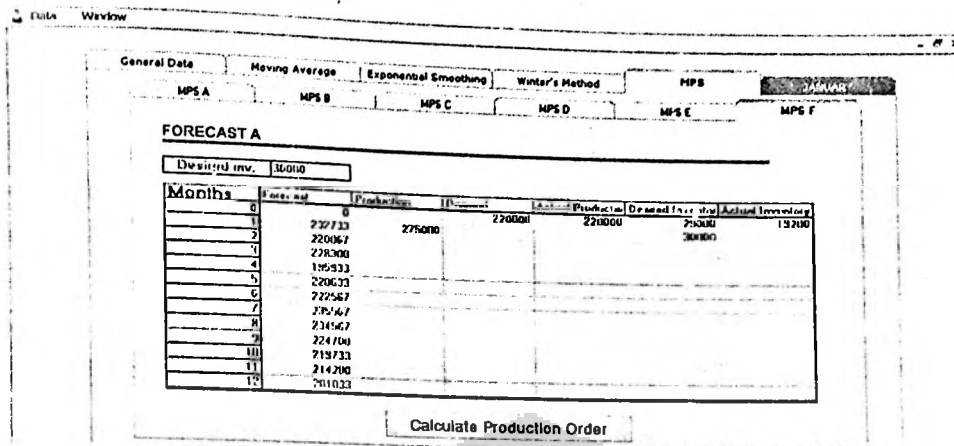
The 'Historical Data (Live Data)' section shows a table with 14 months of data for six work centers (A-F). The data is as follows:

Month	A	B	C	D	E	F
1	200000	470000	290000	230000	300000	220000
2	270000	470000	290000	230000	300000	220000
3	229500	390800	269300	300000	270700	250200
4	240000	474300	273800	287100	260100	182000
5	210700	472900	230900	228400	270700	244700
6	202500	410700	290200	235900	232700	161100
7	261900	382000	228300	281000	308600	256100
8	266000	318500	253100	218000	223100	250500
9	203000	627000	260100	297900	339000	200100
10	293000	364900	261200	241600	232500	253100
11	330000	362800	233600	206400	341700	220900
12	224000	413200	225500	228900	276400	105200
13	205900	381000	222500	250700	254900	230500
14	301000	422700	205700	221600	231700	181400

Gambar 4. 7 Hasil setelah Input General Data

Kemudian setelah MPS melakukan perhitungan terhadap *forecasting* terhadap permintaan dan didapatkan jumlah *production order* yang akan diberikan ke rantai produksi untuk diproses selanjutnya menjadi penjadwalan produksi harian di setiap *work centre*.

Gambar 4.8 berikut merupakan tampilan dari *production order* yang akan diberikan ke rantai produksi



Gambar 4. 8 Production Order dari MPS

Kemudian setelah *production order* dikeluarkan oleh MPS, maka sistem langsung menyimpan *production order* ini ke database, yang nantinya akan digunakan kembali saat memulai perhitungan pada penjadwalan produksi harian.

Selanjutnya *user* langsung masuk pada *window Shop Floor Control (SFC)*, dimana pada sistem ini terdapat dua tab, yaitu *tab general information* yang berisi informasi mengenai parameter perhitungan dan informasi mengenai kapasitas produksi setiap *work centre*, kemudian tab yang kedua yaitu *daily production*, yang berisi hasil dari penjadwalan produksi harian dan aktual produksi dan permintaan aktual. Pada saat masuk ke proses untuk pada sistem pengendalian rantai produksi, *user* dapat langsung menghitung penjadwalan produksi harian setiap *work centre* dengan mengklik tombol *schedule the production*. Kemudian sistem akan mulai menghitung jadwal produksi tersebut untuk beberapa saat.

Gambar 4.7 berikut merupakan proses yang terjadi pada saat perhitungan jadwal produksi harian tersebut. Dimana waktu dari perhitungan jadwal produksi harian ini sampai menampilkan hasil akhir untuk setiap *work centre* adalah sekitar 15 sampai 20 detik.

Data Window

General Data Moving Average Exponential Smoothing Winter's Method MPS

MPS A MPS B MPS C MPS D MPS E MPS F

FORECAST A

Desired inv. 30000

Months	Forecast	Production	Demand	Actual Production	Desired Inv. stock	Actual Inventory
0	0					
1	232733		270000		25000	19200
2	220067	275000			30000	
3	228300					
4	195933					
5	220633					
6	222587					
7	235467					
8	234567					
9	224700					
10	219733					
11	214200					
12	201033					

Calculate Production Order

Gambar 4. 8 Production Order dari MPS

Kemudian setelah *production order* dikeluarkan oleh MPS, maka sistem langsung menyimpan *production order* ini ke database, yang nantinya akan digunakan kembali saat memulai perhitungan pada penjadwalan produksi harian.

Selanjutnya *user* langsung masuk pada window *Shop Floor Control (SFC)*, dimana pada sistem ini terdapat dua tab, yaitu *tab general information* yang berisi informasi mengenai parameter perhitungan dan informasi mengenai kapasitas produksi setiap *work centre*, kemudian tab yang kedua yaitu *daily production*, yang berisi hasil dari penjadwalan produksi harian dan aktual produksi dan permintaan aktual. Pada saat masuk ke proses untuk pada sistem pengendalian rantai produksi, *user* dapat langsung menghitung penjadwalan produksi harian setiap *work centre* dengan mengklik tombol *schedule the production*. Kemudian sistem akan mulai menghitung jadwal produksi tersebut untuk beberapa saat.

Gambar 4.7 berikut merupakan proses yang terjadi pada saat perhitungan jadwal produksi harian tersebut. Dimana waktu dari perhitungan jadwal produksi harian ini sampai menampilkan hasil akhir untuk setiap *work centre* adalah sekitar 15 sampai 20 detik.

The screenshot displays a web-based interface for a 'Daily Production Plan'. At the top, there are tabs for 'General Information' and 'Daily Production Plan'. The 'Input' section includes a 'Today' date field and a table for 'Actual Production' and 'Actual demand' across six work centers (A-F). Below this, the 'Results' section shows tabs for Product A through Product F. A large modal window titled 'Processing Data' is active, displaying the message: 'The data is being processed. It may take a few minutes. Please wait...'. Below this, another modal window titled 'Saving data' is visible, displaying the message: 'The results is being saved. Please wait...'. The background of the interface features a large, faint watermark of the Universitas Indonesia logo.

Gambar 4. 9 Pemrosesan Jadwal Produksi Harian

Pada pemrosesan jadwal produksi harian ini, terdapat beberapa langkah sebelum melakukan random data, yaitu sebagai berikut:

- Jika *production order* yang dikeluarkan lebih kecil atau sama dengan kapasitas maksimum produksi perbulan maka sistem akan melakukan random data sebanyak jumlah hari kerja yang telah di *input* sampai jumlahnya telah memenuhi *production order* yang diberikan.
- Jika *production order* yang dikeluarkan melebihi kapasitas maksimum produksi perbulan maka, untuk *shift* 1 dan 2 langsung diisi sesuai dengan

kapasitas maksimum produksi per *shift*, kemudian sisanya akan diproses dengan aturan yang sama untuk *shift* lembur, dimana diisi sesuai dengan kapasitas maksimum lembur pada hari sebelumnya, kemudian sisanya langsung dipenuhi untuk hari selanjutnya.

Kemudian hasil dari perhitungan ini akan langsung ditampilkan pada sistem dan disimpan kedalam database produksi pada bulan tersebut.

The screenshot displays a 'Daily Production Plan' window. It includes an 'Input' section with a 'Today' field set to '1' and a table for 'Actual Production' and 'Actual demand' across six work centers (A-F). A 'Save Actual Production' button is also present. The 'Results' section features tabs for Product A through F. A detailed table shows production data for 17 days across three shifts (1, 2, 3) and a 'Production Cost' column.

Days	Shift 1	Shift 2	Shift 3	Production Cost
1	6000	6000	3000	2000
2	6000	6000	3000	2000
3	6000	6000	3000	2000
4	6000	6000	3000	2000
5	6000	6000	1033	2000
6	6000	6000	3000	2000
7	6000	6000	3000	2000
8	6000	6000	3000	2000
9	6000	6000	3000	2000
10	6000	6000	3000	2000
11	6000	6000	3000	2000
12	6000	6000	1000	2000
13	6000	6000	6000	2000
14	6000	6000	6000	2000
15	6000	6000	6000	2000
16	6000	6000	6000	2000
17	6000	6000	6000	2000

Gambar 4. 10 Hasil Akhir Penjadwalan Produksi Harian

Dengan sistem ini, *user* dapat melakukan tiga tugas dalam perencanaan dan pengendalian produksi dalam satu waktu serta hanya dilakukan dalam satu area kerja (*window*). Selain itu hasil yang didapatkan dari sistem *te-record* dengan baik dan perhitungannya juga cepat dan akurat. Sehingga *user* dapat melakukan perencanaan secara efisien dan efektif sesuai dengan tujuan awal perancangan sistem ini.

4.4 Analisis

4.4.1 Analisis Sistem Pengendalian Lantai Produksi

Lantai produksi merupakan sebuah elemen penting dalam kelancaran siklus produksi dalam sebuah pabrik. Salah satu aktivitas yang sangat penting dalam sistem pengendalian lantai produksi ini adalah penjadwalan produksi.

Penjadwalan produksi yang baik adalah penjadwalan yang dapat memenuhi fluktuasi permintaan dari berbagai jenis produk, dengan memperhatikan kapasitas (tenaga kerja, mesin dan sumber daya lain) yang ada. Makin terbatas fasilitas yang tersedia dan semakin banyak jenis produk yang harus dibuat akan menyebabkan semakin kompleks persoalan perencanaan dan pengendalian produksinya. Untuk mengurangi kompleksitas masalah, perlu ada suatu metodologi/sistem yang akan membantu fungsi dari *Production Planning Control* khususnya dalam menetapkan penjadwalan rantai produksi yang akan menyesuaikan dengan jadwal produksi Induk (*Master Production Scheduling*) yang diberikan.

Selain untuk memenuhi permintaan sejumlah produk dari jadwal induk produksi yang diberikan, sistem pengendalian rantai produksi ini juga akan memperhitungkan kondisi dari setiap *work centre* dan lini produksi yang tersedia, antara lain kapasitas, biaya dan kebutuhan dari setiap *work centre* dan lini produksi ini. Sehingga dengan adanya sistem ini diharapkan akan mampu membantu mengurangi kompleksitas masalah yang harus diselesaikan oleh *Production Planning Control* dengan dihasilkannya sebuah penjadwalan harian dari setiap *work centre* dan lini produksi dengan penggunaan kapasitas dan biaya yang seefisien mungkin.

4.4.2. Analisis Rancangan Sistem Pengendalian Rantai Produksi

Dalam melakukan perancangan sistem pengendalian rantai produksi ini, kita harus memahami konsep dan tujuan yang diharapkan dari sistem pengendalian rantai produksi sesuai dengan kondisi yang diinginkan oleh *user* dalam hal ini *Production Planning Control*, yaitu menghasilkan sebuah rancangan penjadwalan produksi harian dengan memperhitungkan kapasitas dan biaya yang akan dikeluarkan oleh setiap *work centre* sehingga aktivitas pabrik akan terlaksana dengan seoptimal dan seefisien mungkin.

Rancangan sistem yang dihasilkan ini merupakan bagian dari sebuah sistem yang terintegrasi dengan sistem jadwal induk produksi dan perencanaan kapasitas, dimana sistem pengendalian produksi merupakan pelaksana dari kedua sistem diatas. *Output* dari jadwal induk produksi, yaitu berupa perintah produksi bulanan

yang harus dihasilkan oleh setiap *work centre* akan menjadi *input* pada sistem pengendalian rantai produksi yang nantinya akan memberikan *feedback* ke jadwal induk produksi berupa produksi aktual yang mampu dihasilkan oleh setiap *work centre* dalam jangka waktu periode yang diberikan dan permintaan aktual dari pasar pada periode tersebut. Sedangkan *output* dari perencanaan kapasitas yaitu berupa jumlah tenaga kerja dan jumlah mesin yang akan beroperasi dalam jangka waktu periode produksi yang diberikan, sedangkan *feedback* yang akan diberikan adalah berupa total biaya yang dikeluarkan selama beroperasi dengan menggunakan jumlah tenaga kerja dan mesin yang direncanakan, dimana *feedback* ini hanya bersifat sebagai informasi bagi perencanaan kapasitas.

4.4.3. Analisis Penyusunan Sistem

Pada penyusunan sistem pengendalian rantai produksi ini digunakan bahasa pemrograman *Microsoft Visual Basic* dan sebuah program aplikasi *database Microsoft Acces*. Dimana integrasi keduanya dimulai saat *user* mulai menginput data pada sistem, dimana nantinya akan disimpan kedalam *database* yang telah disediakan sesuai dengan jenis datanya masing-masing.

Setelah semua data yang diperlukan telah tersedia, maka dilakukan penyusunan sistem itu sendiri, yaitu memasukkan logika sistem ke dalam bahasa pemrograman yang digunakan. Tujuan akhir dari perancangan sistem ini adalah untuk menentukan penjadwalan produksi harian setiap *work centre* yang ada, dengan penggunaan biaya yang paling efisien, maka dalam penyusunan bahasa pemrogramannya dilakukan pengacakan angka-angka dengan batasan tertentu yang nantinya akan menghasilkan jumlah yang sesuai dengan jumlah perintah yang dikeluarkan oleh jadwal induk produksi. Salah satu hal yang mendasar dalam penyusunan sebuah sistem adalah tentang keakuratan dari sistem itu sendiri.

Oleh karena itu sebelum menyelesaikan rancangan dilakukan validasi dan verifikasi terlebih dahulu untuk memastikan bahwa rancangan sistem yang dibuat dapat langsung diaplikasikan sesuai dengan kebutuhan dari *user*. Dari hasil validasi dan verifikasi yang dilakukan maka dihasilkanlah bahwa pengacakan yang dilakukan

untuk menghasilkan *output* sesuai yang diinginkan adalah sebanyak 500 kali pengacakan, dengan menghasilkan biaya penalti yang paling minimal.

4.4.4. Analisis Hasil

Hasil penjadwalan produksi harian yang didapat dengan menggunakan sistem ini adalah sebuah penjadwalan harian di setiap *shift* untuk masing-masing *work centre* dengan penalti yang dihasilkan seminimal mungkin. Sehingga sesuai dengan validasi dan verifikasi yang dilaksanakan diatas, keefisienan penggunaan biaya listrik secara keseluruhan dapat tercapai dengan mengefisienkan penjadwalan produksi di setiap *work centre* yang ada.

4.5. Kelebihan dan *Improvement* dari Sistem Pengendalian Lantai Produksi

Dalam perancangan sistem ini pada pelaksanaan dan aplikasinya memiliki beberapa kelebihan jika dibandingkan dengan penggunaan perhitungan manual yang digunakan sebelumnya, selain itu sistem ini juga dapat dilakukan beberapa *improvement* guna kesempurnaan sistem ini terhadap tingkat kompleksitas masalah yang semakin tinggi di masa depan.

4.5.1 Kelebihan Sistem

Berikut ini akan dijelaskan beberapa kelebihan yang dimiliki oleh sistem ini jika dibandingkan dengan perhitungan manual. Beberapa kelebihan tersebut dapat dilihat dari segi kegunaan dan fungsi sistem, rancangan sistem, dan aplikasi sistem dalam praktek di lapangan.

4.5.1.1 Kegunaan dan Fungsi Sistem

Sistem ini dirancang untuk mempermudah dan mempercepat *user* dalam mengambil keputusan dan memperhitungkan penjadwalan produksi yang akan dilaksanakan oleh setiap *work centre* jika dibandingkan dengan melakukan perhitungan manual dengan menggunakan *work sheet* biasa. Sistem ini juga akan menghasilkan sebuah penjadwalan produksi harian dengan memperhitungkan aspek kapasitas dan biaya dari setiap *work centre*, sehingga hasil yang didapatkan memberikan pengaruh terhadap keseluruhan aktivitas pabrik.

Selain itu, sistem pengendalian rantai produksi ini tidak hanya berfungsi sebagai pelaksana saja, tetapi juga akan memberikan informasi berupa *feedback* ke jadwal induk produksi dan perencanaan kapasitas. Sehingga akan terjadi sebuah interaksi yang melibatkan secara aktif tiga system yang sangat berpengaruh terhadap kelancaran produksi sebuah pabrik.

4.5.1.2 Rancangan Sistem

Rancangan dari sistem ini juga sangat mudah dipahami dan digunakan oleh *user* yang akan menggunakan. Dimana *user* hanya perlu melakukan beberapa proses dalam memasukkan data yang diperlukan dalam perhitungan penjadwalan produksi. Kemudian sesuai dengan beberapa langkah-langkah dari cara kerja yang dijelaskan diatas, maka *user* akan mendapatkan hasil yang diharapkan.

4.5.1.3 Aplikasi Sistem

Sistem ini dirancang untuk di-*update* setiap hari sehingga perubahan-perubahan yang terjadi dapat langsung disesuaikan dengan kondisi *work centre*. Misalnya salah satu *work centre* bermasalah sehingga tidak mampu memenuhi produksi harian yang dibebankan, maka *user* dapat langsung meng-*input* produksi actual yang dihasilkan dan jika hal tersebut masih tetap berlangsung selama beberapa hari, maka *user* melakukan evaluasi terhadap *work centre* tersebut. Sehingga untuk periode produksi kedepan diharapkan masalah tersebut telah diselesaikan.

4.5.2 Improvement

Beberapa *improvement* yang dapat dilakukan terhadap sistem pengendalian rantai produksi ini antara lain dalam hal tingkat ke detail'an dari sistem. Dimana dalam sistem yang dirancang belum terdapatnya *field* kalender produksi yang otomatis dalam meng-*update* hari kerja aktual yang ada pada bulan yang sedang berjalan.

5. KESIMPULAN

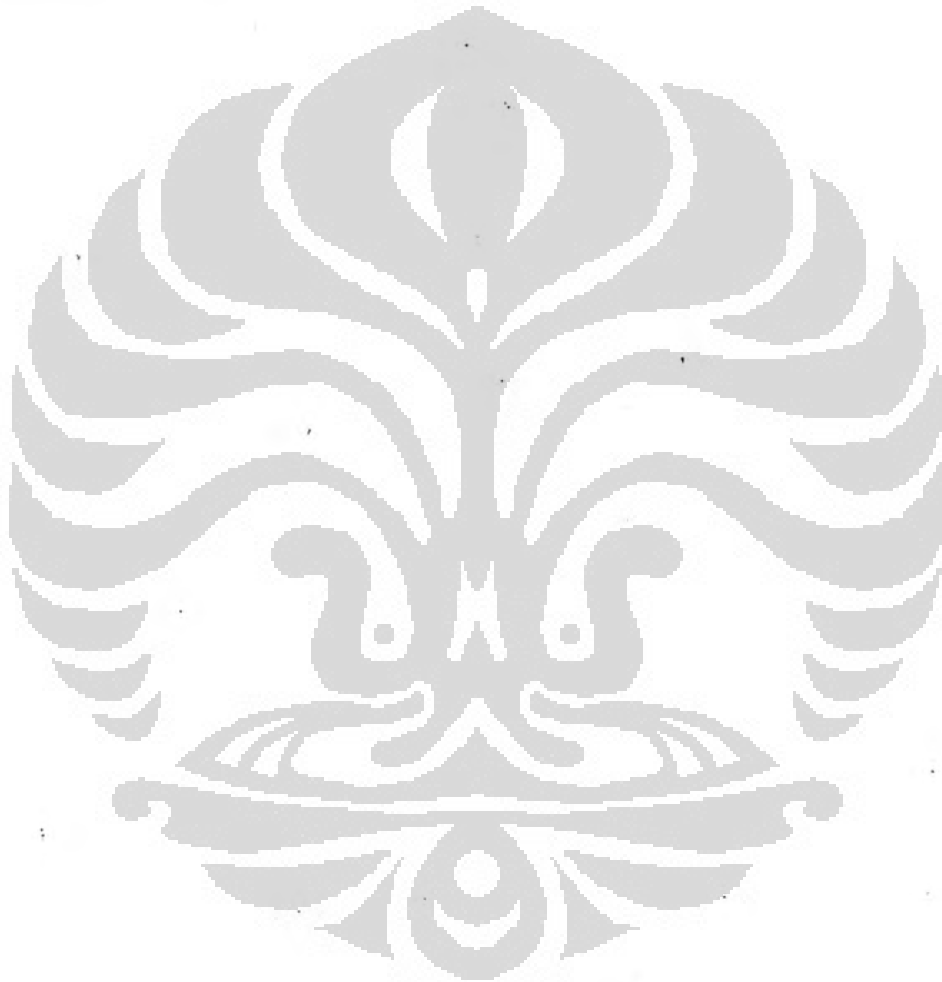
Berdasarkan hasil analisis dari pengumpulan dan pengolahan data yang dilakukan dalam perancangan sistem dengan menggunakan *Microsoft Visual Basic* dan *Microsoft Access*, dapat dihasilkan suatu sistem pengendalian rantai produksi yang terkomputerisasi dan terintegrasi, sehingga *user* dapat melakukan perhitungan penjadwalan produksi harian secara efisien dan efektif.

Sesuai dengan tujuan penelitian ini, maka dapat ditarik suatu kesimpulan sebagai berikut:

1. Pengendalian rantai produksi merupakan proses penentuan penjadwalan produksi harian untuk setiap *work centre* dan lini produksi yang ada, dengan memperhitungkan penggunaan biaya listrik pabrik secara keseluruhan.
2. Hasil identifikasi data menunjukkan, bahwa data yang diperlukan adalah data waktu proses setiap *work centre*, jumlah pekerja normal setiap *work centre*, biaya pekerja setiap *work centre*, perintah produksi, dan biaya listrik pabrik per hari.
3. Sistem yang dirancang digunakan untuk mempermudah dan mempercepat *user* dalam melakukan perhitungan penjadwalan produksi harian di setiap *work centre* jika dibandingkan dengan melakukan perhitungan manual. Sistem yang dihasilkan ini juga mengintegrasikan perencanaan kapasitas (*Capacity Resource Planning*), sistem jadwal induk produksi (*Master Production Scheduling*) dan pengendalian rantai produksi (*Shop Floor Control*) sebagai bagian dari sebuah sistem perencanaan dan pengendalian produksi.
4. Sistem akan menghasilkan sebuah *output* berupa rencana penjadwalan produksi harian di setiap *work centre* pada periode produksi tertentu dan memberikan informasi kepada *user* tentang jumlah produksi aktual yang

mampu dihasilkan oleh setiap *work centre*, dan permintaan aktual dari konsumen.

Sistem yang dirancang ini merupakan kerjasama dengan salah satu perusahaan, maka aplikasi dari sistem ini hanya dapat digunakan untuk perusahaan ini saja, atau perusahaan yang memiliki jenis dan jumlah data yang sama atau kurang, dimana sistem hanya menyediakan perhitungan untuk maksimal enam *work centre*. Jadi, perlu dilakukan beberapa *improvement* agar sistem dapat diaplikasikan di seluruh perusahaan.



Universitas Indonesia

DAFTAR REFERENSI

- Amstead, B.H., Ostwald, Philip F., Begemen & Myrm, L. (1987) *Manufacturing Processes*. John Wiley and Sons
- Arian Zwegers. (1998.) *On Systems Architecting : A Study in Shop Floor Control to Determine architecting concepts and principles* Eindhoven : University Press Facilities
- Bedworth D.D & Bailey J.E,(1987). *Integrated Production Control System*, John Wiley & Sons,
- Everet dan Robert, (1999). *Production and Operation Management*, New Jersey: Prentice Hall
- J.R. Tony Arnold, dan Stephen N. Chapman, (2004). *Introduction to Materials Management*, Pearson Education, Inc
- Microsoft Business Solutions. Shop Floor Control on Axapta.
www.microsoft.com/BusinessSolutions.
- Microsoft Corporation, Microsoft Dynamics AX Shop Floor.
www.microsoft.com/dynamics/ax
- M. P. Groover. (1996) *Fundamentals of Modern Manufacturing*, New Jersey: Prentice-Hall
- Patrik Jonsson; Stig-Arne Mattsson(2003). The implications of fit between planning environments and manufacturing planning and control methods. *International Journal of Operations & Production Management*, 23, 872-900
- Pinedo, M dan Chao, (1999), *Operation Schedulling with Application in Manufacturing and Services*, New York: McGraw-Hill
- www.ikaitsttt.org/pages/download/MANproduksi.pdf
- Thomas M. Carol and Robert D. Dean (1996) *A Bayesian Approach to Plant – Location Decision*. Sciences

LAMPIRAN 4.1

Kode dari Program Sistem Pengendalian Lantai Produksi

```
Option Explicit
Public WithEvents adoPrimaryRS As Recordset
Private Sub setMonth(fgh)
    Select Case fgh
        Case 1
            xtbSFC.TabCaption(2) = "JANUARY"
        Case 2
            xtbSFC.TabCaption(2) = "FEBRUARY"
        Case 3
            xtbSFC.TabCaption(2) = "MARCH"
        Case 4
            xtbSFC.TabCaption(2) = "APRIL"
        Case 5
            xtbSFC.TabCaption(2) = "MAY"
        Case 6
            xtbSFC.TabCaption(2) = "JUNE"
        Case 7
            xtbSFC.TabCaption(2) = "JULY"
        Case 8
            xtbSFC.TabCaption(2) = "AUGUST"
        Case 9
            xtbSFC.TabCaption(2) = "SEPTEMBER"
        Case 10
            xtbSFC.TabCaption(2) = "OKTOBER"
        Case 11
            xtbSFC.TabCaption(2) = "NOVEMBER"
        Case 12
            xtbSFC.TabCaption(2) = "DECEMBER"
    End Select
End Sub
Private Sub loadGrid(gridIndex, jmlRow, jmlCol, _
    initRow, initCol, selRow, selCol)

    adoPrimaryRS.MoveFirst

    With fgdSFC(gridIndex)
        .Rows = jmlRow
        .Cols = jmlCol
        .Row = initRow
        .Col = initCol
        .RowSel = selRow
        .ColSel = selCol
        .Clip = adoPrimaryRS.GetString(adClipString, -1, Chr(9),
            Chr(13), vbNullString)

        .Row = 1
        .Visible = True
        DoEvents
    End With
End Sub
```

```

Public Sub savePenalti()
    Dim d As Integer
    dblJmlRange = adoPrimaryRS.RecordCount
    For d = 1 To dblJmlRange
        dblBatasBawah_A(d) = adoPrimaryRS![BatasBawah A]
        dblBatasAtas_A(d) = adoPrimaryRS![BatasAtas A]
        dblBatasBawah_B(d) = adoPrimaryRS![BatasBawah B]
        dblBatasAtas_B(d) = adoPrimaryRS![BatasAtas B]
        dblBatasBawah_C(d) = adoPrimaryRS![BatasBawah C]
        dblBatasAtas_C(d) = adoPrimaryRS![BatasAtas C]
        dblBatasBawah_D(d) = adoPrimaryRS![BatasBawah D]
        dblBatasAtas_D(d) = adoPrimaryRS![BatasAtas D]
        dblBatasBawah_E(d) = adoPrimaryRS![BatasBawah E]
        dblBatasAtas_E(d) = adoPrimaryRS![BatasAtas E]
        dblBatasBawah_F(d) = adoPrimaryRS![BatasBawah F]
        dblBatasAtas_F(d) = adoPrimaryRS![BatasAtas F]
        dblPenalti_A(d) = adoPrimaryRS![Penalti A]
        dblPenalti_B(d) = adoPrimaryRS![Penalti B]
        dblPenalti_C(d) = adoPrimaryRS![Penalti C]
        dblPenalti_D(d) = adoPrimaryRS![Penalti D]
        dblPenalti_E(d) = adoPrimaryRS![Penalti E]
        dblPenalti_F(d) = adoPrimaryRS![Penalti F]
        adoPrimaryRS.MoveNext
    Next d
End Sub

Public Sub saveMaksProd()
    Dim d As Integer
    For d = 1 To 6
        arrMax(d) = adoPrimaryRS![MaksProduksiperShift]
        adoPrimaryRS.MoveNext
    Next d
End Sub

Public Sub saveCopyProd()
    Dim d As Integer

    Set adoPrimaryRS = New Recordset
    For d = 1 To 6
        adoPrimaryRS.Open _
            "UPDATE Produksi_ " & strMonth & " SET " & Chr(64 + d) & _
            "Shift4 = " & Str(Val(txtActProd(d))) & " WHERE Days = " & _
            Str(intToday), CON, adOpenDynamic, adLockOptimistic
        DoEvents
        'adoPrimaryRS.Close
    Next d
End Sub

Private Sub cmdSaveActual_Click()
    Dim bbb As Integer

```

```

intToday = Val(txtToday)
If intToday = 0 Then
    Load frmMsg
    frmMsg.lblHeader.Caption = "Input error"
    frmMsg.lblMsg.Caption = "Today field is empty. Please fill it first!"
    frmMsg.Show
    Exit Sub
End If

Call convBulan
Call saveCopyProd

If intToday = intJmlHari Then
    'SAVE DATA ACTUAL DEMAND BULAN INI DISINI
    Set adoPrimaryRS = New Recordset
    'masukkan demand bulan lalu menjadi data historis
    For bbb = 1 To 6
        adoPrimaryRS.Open _
            "UPDATE DataHistoris SET " & Chr(64 + bbb) & " = " & _
                & Str(Val(txtDemandBlnLalu(bbb))) & _
                " WHERE Bulan = " & Str(intBlnIni), CON,
                adOpenDynamic, adLockOptimistic

        DoEvents
    Next bbb

    For bbb = 1 To 6
        totalProd(bbb) = 0
    Next bbb

    For bbb = 1 To intJmlHari
        adoPrimaryRS.Open _
            "SELECT AShift4,BShift4,CShift4,DShift4,EShift4, _
                FShift4 FROM Produksi_" & strMonth & _
            " WHERE Days = " & Str(bbb), CON,
            adOpenDynamic, adLockOptimistic

        totalProd(1) = totalProd(1) + adoPrimaryRS![AShift4]
        totalProd(2) = totalProd(2) + adoPrimaryRS![BShift4]
        totalProd(3) = totalProd(3) + adoPrimaryRS![CShift4]
        totalProd(4) = totalProd(4) + adoPrimaryRS![DShift4]
        totalProd(5) = totalProd(5) + adoPrimaryRS![EShift4]
        totalProd(6) = totalProd(6) + adoPrimaryRS![FShift4]

        DoEvents
        adoPrimaryRS.Close
    Next bbb

```

```

'input actual inventory ke variabel
For bbb = 1 To 6
    adoPrimaryRS.Open "SELECT * from [MPS " & Chr(64 + bbb)
        & "]" WHERE Bulan = " & Str(intBlnIni - 1), CON,
        adOpenDynamic, adLockOptimistic
    actualBefore(bbb) = adoPrimaryRS![ActualInv]
    DoEvents
    adoPrimaryRS.Close
Next bbb

'SAVE DI DATABASE
For bbb = 1 To 6
    adoPrimaryRS.Open _
        "UPDATE [MPS " & Chr(64 + bbb) & "]" SET DemandActual =
            " & Str(Val(txtDemandBlnLalu(bbb))) & _
            ", ActualProd=" & Str(totalProd(bbb)) & _
            ", ActualInv=" & Str(totalProd(bbb) +
            actualBefore(bbb) - Val(txtDemandBlnLalu(bbb))) & _
            " WHERE Bulan = " & Str(intBlnIni), CON,
            adOpenDynamic, adLockOptimistic
    DoEvents
Next bbb

'NORMALKAN JML PEKERJA
For bbb = 1 To 6
    adoPrimaryRS.Open _
        "UPDATE Variabel SET JumlahPekerja =
            " & Str(defPekerja(bbb)) & _
            " WHERE ID = " & Str(bbb), CON,
            adOpenDynamic, adLockOptimistic
    DoEvents
Next bbb

End If
End Sub

Private Sub Form_Load()
    Randomize
    barWidth = 12900 'GANTI
    randomLimit = 500 'GANTI
    intToMonth = intBlnIni
    Call setMonth(intBlnIni)
    defPekerja(1) = 20
    defPekerja(2) = 10
    defPekerja(3) = 9
    defPekerja(4) = 11
    defPekerja(5) = 6
    defPekerja(6) = 30

    UnlockTheFormKoneksi
    xtbSFC_TabSwitch 1
End Sub

```

```

Private Sub cmdSaveSFC_Click()
Dim r, s As Integer

Set adoPrimaryRS = New Recordset

For s = 1 To 6
    If blnLembur(s) = False Then
        For r = 1 To intJmlHari
            adoPrimaryRS.Open _
            "UPDATE Produksi_" & strMonth & " SET " & _
            Chr(64 + s) & "Shift1 = " & arrRandom(s, r) & ", " & _
            Chr(64 + s) & "Shift2 = " & arrRandom(s, r + intJmlHari) & _
            " WHERE Days = " & Str(r), CON, adOpenDynamic, adLockOptimistic
            DoEvents

            adoPrimaryRS.Open _
            "UPDATE Produksi_" & strMonth & " SET " & _
            Chr(64 + s) & "TotalPenalti = " & dblMinNow(s) & _
            " WHERE Days = 1", CON, adOpenDynamic, adLockOptimistic
        Next r
    Else
        For r = 1 To intJmlHari
            adoPrimaryRS.Open _
            "UPDATE Produksi_" & strMonth & " SET " & Chr(64 + s) & _
            "Shift1 = " & arrMax(s) * 2 & ", " & Chr(64 + s) & _
            "Shift2 = " & arrMax(s) * 2 & " WHERE Days = " & _
            Str(r), CON, adOpenDynamic, adLockOptimistic
            DoEvents

            adoPrimaryRS.Open _
            "UPDATE Produksi_" & strMonth & " SET " & _
            Chr(64 + s) & "TotalPenalti = " & dblMinNow(s) & _
            " WHERE Days = 1", CON, adOpenDynamic, adLockOptimistic
        Next r
        For r = 1 To (pengali(s) + 1)
            adoPrimaryRS.Open _
            "UPDATE Produksi_" & strMonth & " SET " & Chr(64 + s) & _
            "Shift3 = " & arrRandom(s, r) & " WHERE Days = " & _
            Str(r), CON, adOpenDynamic, adLockOptimistic
            DoEvents
        Next r
    End If
Next s

```

```

End Sub
Private Sub cmdStart_Click()
    Dim g As Integer
    intToday = Val(txtToday)
    If intToday = 0 Then
        Load frmMsg
        frmMsg.lblHeader.Caption = "Input error"
        frmMsg.lblMsg.Caption = vbCrLf + "Today field is empty."
        Please fill it first!
        frmMsg.Show
        Exit Sub
    End If
    Call convBulan

    For g = 1 To 6
        blnAkhRnd(g) = False 'netralkan flag pengakhir random
    Next g

    For g = 1 To 6
        adoPrimaryRS.Open "SELECT ProductionOrder from [MPS " & Chr(64 + g)
        & "]" WHERE Bulan = " & Str(intBlnIni),
        CON, adOpenDynamic, adLockOptimistic
        arrDemand(g) = adoPrimaryRS![ProductionOrder]

        dblBuff(g) = 0 'jumlah total demand hasil random
        arrRndCount(g) = 0 'berapa kali random
        If arrDemand(g) > (arrMax(g) * intJmlHari * 2) Then
            blnLembur(g) = True
            arrDemand(g) = arrDemand(g) - (arrMax(g) * intJmlHari * 2)
            arrMax(g) = arrMax(g) / 2
        End If
        DoEvents
        adoPrimaryRS.Close
    Next g

    Load frmMsg
    frmMsg.cmdOK.Visible = False
    frmMsg.lblHeader.Caption = "Processing Data"
    frmMsg.lblMsg.Caption = vbCrLf + "The data is being processed."
    It may take a few minutes." + vbCrLf + "Please wait..."
    frmMsg.Show

    cmdStart.Enabled = False
    tmrRandomA.Enabled = True
    tmrRandomB.Enabled = True
    tmrRandomC.Enabled = True
    tmrRandomD.Enabled = True
    tmrRandomE.Enabled = True
    tmrRandomF.Enabled = True

End Sub

```

```

Private Sub tmrLimit_Timer()
frmMsg.lblHeader.Caption = "Saving data"
frmMsg.lblMsg.Caption = vbCrLf + "The results is being saved."
    + vbCrLf + "Please wait..."
frmMsg.Show

cmdSaveSFC_Click
tmrLimit.Enabled = False

frmMsg.lblHeader.Caption = "Printing data"
frmMsg.lblMsg.Caption = vbCrLf + "The results is being displayed."
    + vbCrLf + "Please wait..."
frmMsg.Show
loadDaily
End Sub

Private Sub tmrRandomA_Timer()
    randomWCA 1
End Sub

Private Sub tmrRandomB_Timer()
    randomWCB 2
End Sub

Private Sub tmrRandomC_Timer()
    randomWCC 3
End Sub

Private Sub tmrRandomD_Timer()
    randomWCD 4
End Sub

Private Sub tmrRandomE_Timer()
    randomWCE 5
End Sub

Private Sub tmrRandomF_Timer()
    randomWCF 6
End Sub

Private Sub loadDaily()
    Dim fk, lk As Integer

    For fk = 1 To 6
        XTab1.ActiveTab = fk - 1
        Set adoPrimaryRS = New Recordset
        adoPrimaryRS.Open "SELECT Days," & _
            Chr(64 + fk) & "Shift1," & _
            Chr(64 + fk) & "Shift2," & _
            Chr(64 + fk) & "Shift3," & _
            Chr(64 + fk) & "TotalPenalti FROM Produksi_" & strMonth,
            CON, adOpenDynamic, adLockOptimistic

        '-----
        Call loadGrid((2 + fk), _
            (adoPrimaryRS.RecordCount + 1), _
            adoPrimaryRS.Fields.Count, 1, 0, _
            (adoPrimaryRS.RecordCount), _
            (adoPrimaryRS.Fields.Count - 1))
    
```

```

        .Clip = "Shift 2"
    Case 3
        .Clip = "Shift 3"
    Case 4
        .Clip = "Penalty Cost"
    End Select
    .ColWidth(1k) = 2100
Next 1k
End With
adoPrimaryRS.Close
'=====
Next fk
Unload frmMsg
End Sub
Private Sub xtbSFC_TabSwitch(ByVal iLastActiveTab As Integer)
Dim 1k, intBufs, intMods As Integer
Set adoPrimaryRS = New Recordset
adoPrimaryRS.Open "SELECT WC,MaksProduksiperShift,MaksLembur,
ListrikPerProduk,BiayaPekerjaPerJam,MaksProdPerJam FROM Variabel",
CON, adOpenDynamic, adLockOptimistic
Call saveMaksProd
Call loadGrid(1, 7, 6, 1, 0, 6, 5)
'Set the header of the columns
With fgdSFC(1)
    For 1k = 0 To 5
        .Col = 1k
        .Row = 0
        Select Case 1k
            Case 0
                .Clip = "Work Center"
            Case 1
                .Clip = "Production Max/ shift"
            Case 2
                .Clip = "Overtime Max"
            Case 3
                .Clip = "Elect. Cost/ unit"
            Case 4
                .Clip = "Wages/ WC/ hr"
            Case 5
                .Clip = "Production Max/ hr"
        End Select
        fgdSFC(1).ColWidth(1k) = 2065
    Next 1k
End With
adoPrimaryRS.Close

```

```

adoPrimaryRS.Open "select * from Penalti",
CON, adOpenDynamic, adLockOptimistic
Call savePenalti
Call loadGrid(2, _
    (adoPrimaryRS.RecordCount + 1), _
    adoPrimaryRS.Fields.Count, 1, 0, _
    (adoPrimaryRS.RecordCount), _
    (adoPrimaryRS.Fields.Count - 1))

'Set the header of the columns
With fgdSFC(2)
    For lk = 0 To 18
        .Col = lk
        .Row = 0

        intMods = lk Mod 3
        If intMods <> 0 Then
            intBufs = Int(lk / 3) + 1
        Else
            intBufs = Int(lk / 3)
        End If

        Select Case intMods
            Case 1
                .Clip = "BatasBawah " & Chr(64 + intBufs)
            Case 2
                .Clip = "BatasAtas " & Chr(64 + intBufs)
            Case 0
                .Clip = "Penalti " & Chr(64 + intBufs)
        End Select
        fgdSFC(2).ColWidth(lk) = 1600

        If lk = 0 Then
            .ColWidth(lk) = 700
            .Clip = "No."
        End If
    Next lk
End With
adoPrimaryRS.Close
End Sub

```

LAMPIRAN 4.2

Module dari Program

```

Option Explicit
Option Base 1
Public intJmlHari, intCounter(6), barWidth, totalProd(6) As Double
Public arrMax(6), arrDemand(6), dblRangeBawah(6),
dblRangeAtas(6), dblBuff(6) As Double
Public dblBalancer(6), dblJmlRange, dblSisaBalancer(6) As Double
Public dblBatasBawah_A(10), dblBatasAtas_A(10), dblPenalti_A(10) As Double
Public dblBatasBawah_B(10), dblBatasAtas_B(10), dblPenalti_B(10) As Double
Public dblBatasBawah_C(10), dblBatasAtas_C(10), dblPenalti_C(10) As Double
Public dblBatasBawah_D(10), dblBatasAtas_D(10), dblPenalti_D(10) As Double
Public dblBatasBawah_E(10), dblBatasAtas_E(10), dblPenalti_E(10) As Double
Public dblBatasBawah_F(10), dblBatasAtas_F(10), dblPenalti_F(10) As Double
Public arrPenalti(6), subPenalti(6, 44), arrRndCount(6),
dblActualProdTotal(6) As Double
Public arrRandom(6, 44), dblMinNow(6), dblMinCurr(6) As Double
Public intToday, intToMonth, randomLimit, pengali(6), defPekerja(6) As Integer
Public blnKetemu(6) As Boolean
Public blnAkhirRnd(6) As Boolean
Public blnLembur(6) As Boolean
Public strMonth As String
Public Sub lookUpA(RBawahA, RAtasA, RPenaltiA, indxA, wctype2A, dblInputCompareA)
    If dblInputCompareA >= RBawahA And dblInputCompareA < RAtasA Then
        arrPenalti(wctype2A) = RPenaltiA 'BIAYA PENALTI DLM RANGE YBS
        subPenalti(wctype2A, indxA) = (dblInputCompareA - RBawahA) *
arrPenalti(wctype2A) 'SELISIH PRODUKSI DIKALI BIAYA PENALTI DLM RANGE YBS
        dblMinCurr(wctype2A) = dblMinCurr(wctype2A) + subPenalti(wctype2A,
indxA) 'TOTAL PENALTI UTK DIPERBANDINGKAN
        blnKetemu(wctype2A) = True
        'KODE DI BAWAH INI UTK NGEPRINT NILAI HASIL RANDOM
        'BUAT PENGECEKAN DOANG, (DIKOPI KE EXCEL)
        'frmSFC.Text1.Text = frmSFC.Text1.Text & Str(dblInput)
        & Chr(9) & Str(subPenalti(indxA)) + vbCrLf
    Else
    End If
End Sub
Public Sub lookUpB(RBawahB, RAtasB, RPenaltiB, indxB, wctype2B,
dblInputCompareB)
    If dblInputCompareB >= RBawahB And dblInputCompareB < RAtasB Then
        arrPenalti(wctype2B) = RPenaltiB
        subPenalti(wctype2B, indxB) = (dblInputCompareB - RBawahB)
        * arrPenalti(wctype2B)
        dblMinCurr(wctype2B) = dblMinCurr(wctype2B) + subPenalti(wctype2B, indxB)
        blnKetemu(wctype2B) = True
        'KODE DI BAWAH INI UTK NGEPRINT NILAI HASIL RANDOM
        'BUAT PENGECEKAN DOANG, (DIKOPI KE EXCEL)
        'frmSFC.Text1.Text = frmSFC.Text1.Text & Str(dblInput)
        & Chr(9) & Str(subPenalti(indxB)) + vbCrLf
    Else

```

```

End If
End Sub

Public Sub lookUpC(RBawahC, RAtasC, RPenaltiC, indxC, wctype2C, dblInputCompareC)
If dblInputCompareC >= RBawahC And dblInputCompareC < RAtasC Then
    arrPenalti(wctype2C) = RPenaltiC
    subPenalti(wctype2C, indxC) = (dblInputCompareC - RBawahC)
    * arrPenalti(wctype2C)
    dblMinCurr(wctype2C) = dblMinCurr(wctype2C) + subPenalti(wctype2C, indxC)
    blnKetemu(wctype2C) = True
    'KODE DI BAWAH INI UTK NGEPRINT NILAI HASIL RANDOM
    'BUAT PENGECEKAN DOANG, (DIKOPI KE EXCEL)
    'frmSFC.Text1.Text = frmSFC.Text1.Text & Str(dblInput) & Chr(9)
    & Str(subPenalti(indxC)) + vbCrLf
Else
End If
End Sub

Public Sub lookUpD(RBawahD, RAtasD, RPenaltiD, indxD, wctype2D, dblInputCompareD)
If dblInputCompareD >= RBawahD And dblInputCompareD < RAtasD Then
    arrPenalti(wctype2D) = RPenaltiD
    subPenalti(wctype2D, indxD) = (dblInputCompareD - RBawahD)
    * arrPenalti(wctype2D)
    dblMinCurr(wctype2D) = dblMinCurr(wctype2D) + subPenalti(wctype2D, indxD)
    blnKetemu(wctype2D) = True
    'KODE DI BAWAH INI UTK NGEPRINT NILAI HASIL RANDOM
    'BUAT PENGECEKAN DOANG, (DIKOPI KE EXCEL)
    'frmSFC.Text1.Text = frmSFC.Text1.Text & Str(dblInput)
    & Chr(9) & Str(subPenalti(indxD)) + vbCrLf
Else
End If
End Sub

Public Sub lookUpE(RBawahE, RAtasE, RPenaltiE, indxE, wctype2E, dblInputCompareE)
If dblInputCompareE >= RBawahE And dblInputCompareE < RAtasE Then
    arrPenalti(wctype2E) = RPenaltiE
    subPenalti(wctype2E, indxE) = (dblInputCompareE - RBawahE)
    * arrPenalti(wctype2E)
    dblMinCurr(wctype2E) = dblMinCurr(wctype2E) + subPenalti(wctype2E, indxE)
    blnKetemu(wctype2E) = True
    'KODE DI BAWAH INI UTK NGEPRINT NILAI HASIL RANDOM
    'BUAT PENGECEKAN DOANG, (DIKOPI KE EXCEL)
    'frmSFC.Text1.Text = frmSFC.Text1.Text & Str(dblInput) & Chr(9)
    & Str(subPenalti(indxE)) + vbCrLf
Else
End If
End Sub

```

```

Public Sub lookUpF(RBawahF, RAtasF, RPenaltiF, indxF, wctype2F, dblInputCompareF)
    If dblInputCompareF >= RBawahF And dblInputCompareF < RAtasF Then
        arrPenalti(wctype2F) = RPenaltiF
        subPenalti(wctype2F, indxF) = (dblInputCompareF - RBawahF)
            * arrPenalti(wctype2F)
        dblMinCurr(wctype2F) = dblMinCurr(wctype2F) + subPenalti(wctype2F, indxF)
        blnKetemu(wctype2F) = True
        'KODE DI BAWAH INI UTK NGEPRINT NILAI HASIL RANDOM
        'BUAT PENGECEKAN DOANG, (DIKOPI KE EXCEL)
        'frmSFC.Text1.Text = frmSFC.Text1.Text & Str(dblInput) & Chr(9)
            & Str(subPenalti(indxF)) + vbCrLf
    Else
    End If
End Sub

Public Sub randomWCA(wctypeA)
    Dim a1, b1 As Integer

    For a1 = 1 To 6
        dblMinCurr(a1) = 0
    Next a1

```

```

If blnLembur(wctypeA) = False Then
'CUT HERE!! =====
dblRangeBawah(wctypeA) = arrDemand(wctypeA) Mod arrMax(wctypeA) 'UPDATE:
dblRangeAtas(wctypeA) = arrMax(wctypeA) - dblRangeBawah(wctypeA) 'CABANG

'BLOK ACRAK DONG ADA DI SINI =====
For al = 1 To intJmlHari * 2 'UPDATE: angka satu diganti menjadi variabel hari
Randomize
arrRandom(wctypeA, al) = Round(dblRangeBawah(wctypeA) + Rnd * (dblRangeAtas(wctypeA)))
dblBuff(wctypeA) = dblBuff(wctypeA) + arrRandom(wctypeA, al)
Next al

If dblBuff(wctypeA) > arrDemand(wctypeA) Then 'CABANG
dblBalancer(wctypeA) = (dblBuff(wctypeA) - arrDemand(wctypeA)) / (intJmlHari * 2)
dblBuff(wctypeA) = 0
For al = 1 To intJmlHari * 2
arrRandom(wctypeA, al) = arrRandom(wctypeA, al) - dblBalancer(wctypeA)
Next al
Else
atasLagi:
dblBalancer(wctypeA) = (arrDemand(wctypeA) - dblBuff(wctypeA)) / (intJmlHari * 2)
dblSisaBalancer(wctypeA) = 0
dblBuff(wctypeA) = 0
For al = 1 To intJmlHari * 2
If arrMax(wctypeA) - arrRandom(wctypeA, al) > dblBalancer(wctypeA) Then
arrRandom(wctypeA, al) = arrRandom(wctypeA, al) + dblBalancer(wctypeA)
Else
dblSisaBalancer(wctypeA) = dblSisaBalancer(wctypeA) + dblBalancer(wctypeA)
End If
Next al

If dblSisaBalancer(wctypeA) > 0 Then
For al = 1 To intJmlHari * 2
dblBuff(wctypeA) = dblBuff(wctypeA) + arrRandom(wctypeA, al)
Next al
dblBalancer(wctypeA) = dblSisaBalancer(wctypeA)
GoTo atasLagi
Else
'do nothing
End If
End If

```

```

For al = 1 To intJmlHari * 2
    arrRandom(wctypeA, al) = Int(arrRandom(wctypeA, al))
    dblBuff(wctypeA) = dblBuff(wctypeA) + arrRandom(wctypeA, al)
Next al

If arrDemand(wctypeA) > dblBuff(wctypeA) Then
    intCounter(wctypeA) = 0
    For al = 1 To intJmlHari * 2
        If arrMax(wctypeA) - arrRandom(wctypeA, al) > ((arrDemand(wctypeA) - dblBuff(wctypeA)) / 2) Then
            intCounter(wctypeA) = intCounter(wctypeA) + 1
            Select Case intCounter(wctypeA)
                Case 1
                    arrRandom(wctypeA, al) = arrRandom(wctypeA, al) + Int(((arrDemand(wctypeA) - dblBuff(wctypeA)) / 2))
                Case 2
                    arrRandom(wctypeA, al) = arrRandom(wctypeA, al) + (arrDemand(wctypeA) - dblBuff(wctypeA)) -
                        Int(((arrDemand(wctypeA) - dblBuff(wctypeA)) / 2))
            End Select
        End If
    Next al
End If
dblBuff(wctypeA) = 0
'END ACAK DONG=====

For al = 1 To intJmlHari * 2
    binKetemu(wctypeA) = False
    For bl = 1 To dblJmlRange
        lookUpA dblBatasBawah_A(bl), dblBatasAtas_A(bl), dblPenalti_A(bl), al, wctypeA, arrRandom(wctypeA, al)
        If binKetemu(wctypeA) = True Then Exit For
    Next bl
Next al

'cek apakah nilai yg baru lebih kecil,
'klo iya jadikan nilai terkecil yg baru
If dblMinNow(wctypeA) = 0 Or dblMinCurr(wctypeA) < dblMinNow(wctypeA) Then
    dblMinNow(wctypeA) = dblMinCurr(wctypeA)
End If
END CUT!! =====

frmSFC.txtPrinter(wctypeA).Text = frmSFC.txtPrinter(wctypeA).Text + Str(dblMinNow(wctypeA)) + vbCrLf
rrPndCount(wctypeA) = arrRndCount(wctypeA) + 1 'berapa kali random

If arrRndCount(wctypeA) > randomLimit Then
    frmSFC.tmrRandomA.Enabled = False
    frmSFC.txtPrinter(wctypeA).Text = frmSFC.txtPrinter(wctypeA).Text + "Selesai dg "
    arrRndCount(wctypeA) & " kali random"
    rrRndCount(wctypeA) = 0
End If

```

```

Else
'EDIT TERBARU --- MAKSIMALKAN LEMBUR -----
pengali(wctypeA) = Int(arrDemand(wctypeA) / arrMax(wctypeA))
For a1 = 1 To pengali(wctypeA)
    arrRandom(wctypeA, a1) = arrMax(wctypeA)
Next a1
arrRandom(wctypeA, pengali(wctypeA) + 1) = arrDemand(wctypeA)
    - (arrMax(wctypeA) * pengali(wctypeA))
'END CUT!! -----

For a1 = 1 To intJmlHari
    blnKetemu(wctypeA) = False
    For b1 = 6 To dblJmlRange
        lookUpA dblBatasBawah_A(b1), dblBarasAtas_A(b1), dblPenalti_A(b1 - 5),
            a1, wctypeA, arrRandom(wctypeA, a1)
        If blnKetemu(wctypeA) = True Then Exit For
    Next b1
Next a1

'cek apakah nilai yg baru lebih kecil,
'klo iya jadikan nilai terkecil yg baru
If dblMinNow(wctypeA) = 0 Or dblMinCurr(wctypeA) < dblMinNow(wctypeA) Then
    dblMinNow(wctypeA) = dblMinCurr(wctypeA)
End If
'END CUT!! -----

frmSFC.txtPrinter(wctypeA).Text = frmSFC.txtPrinter(wctypeA).Text
    + Str(dblMinNow(wctypeA)) + vbCrLf
arrRndCount(wctypeA) = arrRndCount(wctypeA) + 1 'berapa kali random
If arrRndCount(wctypeA) > randomLimit Then
    frmSFC.tmrRandomA.Enabled = False
    frmSFC.txtPrinter(wctypeA).Text = frmSFC.txtPrinter(wctypeA).Text +
        "Selesai dg " & arrRndCount(wctypeA) & " kali random"
    arrRndCount(wctypeA) = 0
End If
End If
End Sub

Public Sub randomWCB(wctypeB)
    Dim a2, b2 As Integer

    For a2 = 1 To 6
        dblMinCurr(a2) = 0
    Next a2
    If blnLembur(wctypeB) = False Then
        'CUT HERE!! -----
        dblRangeBawah(wctypeB) = arrDemand(wctypeB) Mod arrMax(wctypeB)
        dblRangeAtas(wctypeB) = arrMax(wctypeB) - dblRangeBawah(wctypeB)
        'BLOK ACAK DONG ADA DI SINI -----
        For a2 = 1 To intJmlHari * 2
            Randomize
            arrRandom(wctypeB, a2) = Round(dblRangeBawah(wctypeB) + Rnd * (dblRangeAtas(wctypeB)))
            dblBuff(wctypeB) = dblBuff(wctypeB) + arrRandom(wctypeB, a2)
        Next a2
        If dblBuff(wctypeB) > arrDemand(wctypeB) Then
            dblBalancer(wctypeB) = (dblBuff(wctypeB) - arrDemand(wctypeB)) / (intJmlHari * 2)
            dblBuff(wctypeB) = 0
            For a2 = 1 To intJmlHari * 2
                arrRandom(wctypeB, a2) = arrRandom(wctypeB, a2) - dblBalancer(wctypeB)
            Next a2
        Else

```

```

atasLagi:
dblBalancer(wctypeB) = (arrDemand(wctypeB) - dblBuff(wctypeB)) / (intJmlHari * 2)
dblSisaBalancer(wctypeB) = 0
dblBuff(wctypeB) = 0
For a2 = 1 To intJmlHari * 2
    If arrMax(wctypeB) - arrRandom(wctypeB, a2) > dblBalancer(wctypeB) Then
        arrRandom(wctypeB, a2) = arrRandom(wctypeB, a2) + dblBalancer(wctypeB)
    Else
        dblSisaBalancer(wctypeB) = dblSisaBalancer(wctypeB) + dblBalancer(wctypeB)
    End If
Next a2

If dblSisaBalancer(wctypeB) > 0 Then
    For a2 = 1 To intJmlHari * 2
        dblBuff(wctypeB) = dblBuff(wctypeB) + arrRandom(wctypeB, a2)
    Next a2
    dblBalancer(wctypeB) = dblSisaBalancer(wctypeB)
    GoTo atasLagi
Else
    'do nothing
End If
End If

For a2 = 1 To intJmlHari * 2
    arrRandom(wctypeB, a2) = Int(arrRandom(wctypeB, a2))
    dblBuff(wctypeB) = dblBuff(wctypeB) + arrRandom(wctypeB, a2)
Next a2

If arrDemand(wctypeB) > dblBuff(wctypeB) Then
    intCounter(wctypeB) = 0
    For a2 = 1 To intJmlHari * 2
        If arrMax(wctypeB) - arrRandom(wctypeB, a2) > ((arrDemand(wctypeB) -
        - dblBuff(wctypeB)) / 2) Then
            intCounter(wctypeB) = intCounter(wctypeB) + 1
            Select Case intCounter(wctypeB)
            Case 1
                arrRandom(wctypeB, a2) = arrRandom(wctypeB, a2) + Int(((arrDemand(wctypeB)
                - dblBuff(wctypeB)) / 2))
            Case 2
                arrRandom(wctypeB, a2) = arrRandom(wctypeB, a2) + (arrDemand(wctypeB) -
                dblBuff(wctypeB)) - Int(((arrDemand(wctypeB) - dblBuff(wctypeB)) / 2))
            End Select
        End If
    Next a2
End If

dblBuff(wctypeB) = 0
'END ACAK DONG=====

```

```

for a2 = 1 To intJmlHari * 2
    blnKetemu(wctypeB) = False
    For b2 = 1 To dblJmlRange
        lookUpB dblBatasBawah_B(b2),
        dblBatasAtas_B(b2), dblPenalti_B(b2), a2, wctypeB, arrRandom(wctypeB,
        a2)
    For
        If blnKetemu(wctypeB) = True Then Exit
    Next b2
    Next a2
    'cek apakah nilai yg baru lebih kecil,
    'klo iya jadikan nilai terkecil yg baru
    If dblMinNow(wctypeB) = 0 Or dblMinCurr(wctypeB) <
    dblMinNow(wctypeB) Then
        dblMinNow(wctypeB) = dblMinCurr(wctypeB)
    End If
    'END CUT!!
    =====
    frmSFC.txtPrinter(wctypeB).Text =
    frmSFC.txtPrinter(wctypeB).Text + Str(dblMinNow(wctypeB)) + vbCrLf
    arrRndCount(wctypeB) = arrRndCount(wctypeB) + 1
    'berapa kali random
    If arrRndCount(wctypeB) > randomLimit Then
        frmSFC.tmrRandomB.Enabled = False
        frmSFC.txtPrinter(wctypeB).Text =
        frmSFC.txtPrinter(wctypeB).Text + "Selesai dg " & arrRndCount(wctypeB) &
        " kali random"
        arrRndCount(wctypeB) = 0
    End If
Else
    'EDIT TERBARU --- MAKSIMALKAN LEMBUR
    =====
    pengali(wctypeB) = Int(arrDemand(wctypeB) / arrMax(wctypeB))
    For a2 = 1 To pengali(wctypeB)
        arrRandom(wctypeB, a2) = arrMax(wctypeB)
    Next a2
    arrRandom(wctypeB, pengali(wctypeB) + 1) = arrDemand(wctypeB) -
    (arrMax(wctypeB) * pengali(wctypeB))
    'END CUT!!
    =====

    For a2 = 1 To intJmlHari
        blnKetemu(wctypeB) = False
        For b2 = 6 To dblJmlRange
            lookUpB dblBatasBawah_B(b2), dblBatasAtas_B(b2),
            dblPenalti_B(b2 - 5), a2, wctypeB, arrRandom(wctypeB, a2)
            If blnKetemu(wctypeB) = True Then Exit For
        Next b2
    Next a2

    'cek apakah nilai yg baru lebih kecil,
    'klo iya jadikan nilai terkecil yg baru
    If dblMinNow(wctypeB) = 0 Or dblMinCurr(wctypeB) <
    dblMinNow(wctypeB) Then
        dblMinNow(wctypeB) = dblMinCurr(wctypeB)
    End If
    'END CUT!! =====
    frmSFC.txtPrinter(wctypeB).Text = frmSFC.txtPrinter(wctypeB).Text +
    Str(dblMinNow(wctypeB)) + vbCrLf
    arrRndCount(wctypeB) = arrRndCount(wctypeB) + 1 'berapa kali
    random

    If arrRndCount(wctypeB) > randomLimit Then

```

```

        frmSFC.tmrRandomB.Enabled = False
        frmSFC.txtPrinter(wctypeB).Text =
        frmSFC.txtPrinter(wctypeB).Text + "Selesai dg " & arrRndCount(wctypeB)
        & " kali random"
        arrRndCount(wctypeB) = 0
    End If

End If

End Sub

Public Sub randomWCC(wctypeC)
    Dim a3, b3 As Integer

    For a3 = 1 To 6
        dblMinCurr(a3) = 0
    Next a3

    If blnLembur(wctypeC) = False Then
        'CUT HERE!!
        =====
        arrMax(wctypeC) = dblRangeBawah(wctypeC) = arrDemand(wctypeC) Mod
        dblRangeAtas(wctypeC) = arrMax(wctypeC) -
        dblRangeBawah(wctypeC)
        'BLOK ACAK DONG ADA DI SINI
        =====
        For a3 = 1 To intJmlHari * 2
            Randomize
            arrRandom(wctypeC, a3) =
            Round(dblRangeBawah(wctypeC) + Rnd * (dblRangeAtas(wctypeC)))
            dblBuff(wctypeC) = dblBuff(wctypeC) +
            arrRandom(wctypeC, a3)
        Next a3

        If dblBuff(wctypeC) > arrDemand(wctypeC)
        Then
            dblBalancer(wctypeC) =
            (dblBuff(wctypeC) - arrDemand(wctypeC)) / (intJmlHari * 2)
            dblBuff(wctypeC) = 0
            For a3 = 1 To intJmlHari * 2
                arrRandom(wctypeC, a3) = arrRandom(wctypeC, a3) - dblBalancer(wctypeC)
            Next a3
        Else
            atasLagi:
            dblBalancer(wctypeC) =
            (arrDemand(wctypeC) - dblBuff(wctypeC)) / (intJmlHari * 2)
            dblSisaBalancer(wctypeC) = 0
            dblBuff(wctypeC) = 0
            For a3 = 1 To intJmlHari * 2
                If arrMax(wctypeC) -
                arrRandom(wctypeC, a3) > dblBalancer(wctypeC) Then
                    arrRandom(wctypeC, a3) =
                    arrRandom(wctypeC, a3) + dblBalancer(wctypeC)
                Else
                    dblSisaBalancer(wctypeC) =
                    dblSisaBalancer(wctypeC) + dblBalancer(wctypeC)
                End If
            Next a3

            If dblSisaBalancer(wctypeC) > 0 Then
                For a3 = 1 To intJmlHari * 2
                    dblBuff(wctypeC) =
                    dblBuff(wctypeC) + arrRandom(wctypeC, a3)
                Next a3
                dblBalancer(wctypeC) =

```

```

                                Next a3
                                dblBalancer(wctypeC) =
                                GoTo atasLagi
                                Else
                                'do nothing
                                End If
                                End If

                                For a3 = 1 To intJmlHari * 2
                                arrRandom(wctypeC, a3) =
                                dblBuff(wctypeC) = dblBuff(wctypeC) +
                                arrRandom(wctypeC, a3)

                                Next a3
                                If arrDemand(wctypeC) > dblBuff(wctypeC) Then
                                intCounter(wctypeC) = 0
                                For a3 = 1 To intJmlHari * 2
                                If arrMax(wctypeC) -
                                arrRandom(wctypeC, a3) > ((arrDemand(wctypeC) - dblBuff(wctypeC)) / 2)
                                Then
                                intCounter(wctypeC) + 1
                                intCounter(wctypeC) =
                                Select Case
                                Case 1
                                arrRandom(wctypeC, a3) =
                                arrRandom(wctypeC, a3) + Int(((arrDemand(wctypeC) - dblBuff(wctypeC)) -
                                / 2))
                                Case 2
                                arrRandom(wctypeC, a3) =
                                arrRandom(wctypeC, a3) + (arrDemand(wctypeC) - dblBuff(wctypeC)) -
                                Int(((arrDemand(wctypeC) - dblBuff(wctypeC)) / 2))
                                Exit For
                                End Select
                                End If
                                Next a3
                                End If
                                dblBuff(wctypeC) = 0
                                'END ACAK

                                DONG=====

                                For a3 = 1 To intJmlHari * 2
                                blnKetemu(wctypeC) = False
                                For b3 = 1 To dblJmlRange
                                lookUpC dblBatasBawah_C(b3),
                                dblBatasAtas_C(b3), dblPenalti_C(b3), a3, wctypeC, arrRandom(wctypeC,
                                a3)
                                If blnKetemu(wctypeC) = True Then Exit
                                For
                                Next b3
                                Next a3

                                'cek apakah nilai yg baru lebih kecil,
                                'klo iya jadikan nilai terkecil yg baru
                                If dblMinNow(wctypeC) = 0 Or dblMinCurr(wctypeC) <
                                dblMinNow(wctypeC) Then
                                dblMinNow(wctypeC) = dblMinCurr(wctypeC)
                                End If

                                'END CUT!! =====

                                frmSFC.txtPrinter(wctypeC).Text =
                                frmSFC.txtPrinter(wctypeC).Text + Str(dblMinNow(wctypeC)) + vbCrLf
                                arrRndCount(wctypeC) = arrEndCount(wctypeC) + 1
                                'berapa kali random

```

```

If arrRndCount(wctypeC) > randomLimit Then
    frmSFC.tmrRandomC.Enabled = False
    frmSFC.txtPrinter(wctypeC).Text =
    & " kali random"
    arrRndCount(wctypeC) = 0
End If
Else
    'EDIT TERBARU --- MAKSIMALKAN LEMBUR
    =====
    pengali(wctypeC) = Int(arrDemand(wctypeC) / arrMax(wctypeC))
    For a3 = 1 To pengali(wctypeC)
        arrRandom(wctypeC, a3) = arrMax(wctypeC)
    Next a3
    arrRandom(wctypeC, pengali(wctypeC) + 1) = arrDemand(wctypeC) -
    (arrMax(wctypeC) * pengali(wctypeC))
    'END CUT!!
    =====

    For a3 = 1 To intJmlHari
        blnKetemu(wctypeC) = False
        For b3 = 6 To dblJmlRange
            lookUpC dblBatasBawah_C(b3), dblBatasAtas_C(b3),
            dblPenalti_C(b3 - 5), a3, wctypeC, arrRandom(wctypeC, a3)
            If blnKetemu(wctypeC) = True Then Exit For
        Next b3
    Next a3

    'cek apakah nilai yg baru lebih kecil,
    'klo iya jadikan nilai terkecil yg baru
    If dblMinNow(wctypeC) = 0 Or dblMinCurr(wctypeC) <
    dblMinNow(wctypeC) Then
        dblMinNow(wctypeC) = dblMinCurr(wctypeC)
    End If
    'END CUT!! =====

    frmSFC.txtPrinter(wctypeC).Text = frmSFC.txtPrinter(wctypeC).Text +
    Str(dblMinNow(wctypeC)) + vbCrLf
    arrRndCount(wctypeC) = arrRndCount(wctypeC) + 1 'berapa kali
    random

    'If blnAkhirRnd(wctypeC) = True Then
    '    frmSFC.tmrRandomC.Enabled = False
    '    frmSFC.txtPrinter(wctypeC).Text =
    frmSFC.txtPrinter(wctypeC).Text + "Selesai dg " & arrRndCount(wctypeC)
    & " kali random"
    '    arrRndCount(wctypeC) = 0
    'End If

    If arrRndCount(wctypeC) > randomLimit Then
        frmSFC.tmrRandomC.Enabled = False
        frmSFC.txtPrinter(wctypeC).Text =
        frmSFC.txtPrinter(wctypeC).Text + "Selesai dg " & arrRndCount(wctypeC)
        & " kali random"
        arrRndCount(wctypeC) = 0
    End If
End If
End Sub
Public Sub randomWCD(wctypeD)
    Dim a4, b4 As Integer

    For a4 = 1 To 6
        dblMinCurr(a4) = 0
    Next a4
    If blnLembur(wctypeD) = False Then

```

```

'CUT HERE!! =====
arrMax(wctypeD)          dblRangeBawah(wctypeD) = arrDenand(wctypeD) Mod
dblRangeBawah(wctypeD)   dblRangeAtas(wctypeD) = arrMax(wctypeD) -

=====
'BLOK ACAK DONG ADA DI SINI
=====
For a4 = 1 To intJmlHari * 2
    Randomize
    arrRandom(wctypeD, a4) =
Round(dblRangeBawah(wctypeD) + Rnd * (dblRangeAtas(wctypeD)))
+ arrRandom(wctypeD, a4)    dblBuff(wctypeD) = dblBuff(wctypeD)
Next a4

Then                        If dblBuff(wctypeD) > arrDemand(wctypeD)
                            dblBalancer(wctypeD) =
(dblBuff(wctypeD) - arrDemand(wctypeD)) / (intJmlHari * 2)
                            dblBuff(wctypeD) = 0
                            For a4 = 1 To intJmlHari * 2
                                arrRandom(wctypeD, a4) =
arrRandom(wctypeD, a4) - dblBalancer(wctypeD)
                            Next a4
                            Else
atasLagi:                  dblBalancer(wctypeD) =
(arrDemand(wctypeD) - dblBuff(wctypeD)) / (intJmlHari * 2)
                            dblSisaBalancer(wctypeD) = 0
                            dblBuff(wctypeD) = 0
                            For a4 = 1 To intJmlHari * 2
                                If arrMax(wctypeD) -
arrRandom(wctypeD, a4) > dblBalancer(wctypeD) Then
                                    arrRandom(wctypeD, a4) =
arrRandom(wctypeD, a4) + dblBalancer(wctypeD)
                                Else
                                    dblSisaBalancer(wctypeD) =
dblSisaBalancer(wctypeD) + dblBalancer(wctypeD)
                                End If
                            Next a4
                            If dblSisaBalancer(wctypeD) > 0 Then
                                For a4 = 1 To intJmlHari * 2
                                    dblBuff(wctypeD) =
dblBuff(wctypeD) + arrRandom(wctypeD, a4)
                                Next a4
                                dblBalancer(wctypeD) =
dblSisaBalancer(wctypeD)
                                GoTo atasLagi
                            Else
                                'do nothing
                            End If
                            End If

                            For a4 = 1 To intJmlHari * 2
                                arrRandom(wctypeD, a4) =
Int(arrRandom(wctypeD, a4))
                                dblBuff(wctypeD) = dblBuff(wctypeD)
                                + arrRandom(wctypeD, a4)
                            Next a4
                            If arrDemand(wctypeD) > dblBuff(wctypeD)
Then
                                intCounter(wctypeD) = 0

```

```

Then
    intCounter(wctypeD) = 0
    For a4 = 1 To intJmlHari * 2
        If arrMax(wctypeD) -
            Then
                arrRandom(wctypeD, a4) > ((arrDemand(wctypeD) - dblBuff(wctypeD)) / 2)
                intCounter(wctypeD) + 1
                intCounter(wctypeD) =
                Select Case
                    Case 1
                        arrRandom(wctypeD, a4) =
                        2))
                    Case 2
                        arrRandom(wctypeD, a4) =
                        Int((arrDemand(wctypeD) - dblBuff(wctypeD)) -
                        Int((arrDemand(wctypeD) - dblBuff(wctypeD)) / 2))
                        Exit For
                    End Select
                End If
            Next a4
        End If
        dblBuff(wctypeD) = 0
    'END ACAK
DONG=====
For a4 = 1 To intJmlHari * 2
    blnKetemu(wctypeD) = False
    For b4 = 1 To dblJmlRange
        lookUpD dblBatasBawah_D(b4),
        dblBatasAtas_D(b4), dblPenalti_D(b4), a4, wctypeD, arrRandom(wctypeD,
        a4)
        If blnKetemu(wctypeD) = True Then Exit
    For
        Next b4
    Next a4

    'cek apakah nilai yg baru lebih kecil,
    'klo iya jadikan nilai terkecil yg baru
    If dblMinNow(wctypeD) = 0 Or dblMinCurr(wctypeD)
    < dblMinNow(wctypeD) Then
        dblMinNow(wctypeD) = dblMinCurr(wctypeD)
    End If
    'END CUT!!
=====

    frmSFC.txtPrinter(wctypeD).Text =
    frmSFC.txtPrinter(wctypeD).Text + Str(dblMinNow(wctypeD)) + vbCrLf
    arrRndCount(wctypeD) = arrRndCount(wctypeD) + 1
    'berapa kali random

    If arrRndCount(wctypeD) > randomLimit Then
        frmSFC.tmrRandomD.Enabled = False
        frmSFC.txtPrinter(wctypeD).Text =
        frmSFC.txtPrinter(wctypeD).Text + "Selesai dg " & arrRndCount(wctypeD) &
        " kali random"
        arrRndCount(wctypeD) = 0
    End If

Else
    'EDIT TERBARU --- MAKSIMALKAN LEMBUR
    =====

```

```

'EDIT TERBARU --- MAKSIMALKAN LEMBUR
=====
    pengali(wctypeD) = Int(arrDemand(wctypeD) / arrMax(wctypeD))
    For a4 = 1 To pengali(wctypeD)
        arrRandom(wctypeD, a4) = arrMax(wctypeD)
    Next a4
    arrRandom(wctypeD, pengali(wctypeD) + 1) = arrDemand(wctypeD) -
(arrMax(wctypeD) * pengali(wctypeD))
'END CUT!!
=====
    For a4 = 1 To intJmlHari
        blnKetemu(wctypeD) = False
        For b4 = 6 To dblJmlRange
            lookUpD dblBatasBawah_D(b4), dblBatasAtas_D(b4),
            dblPenalti_D(b4 - 5), a4, wctypeD, arrRandom(wctypeD, a4)
            If blnKetemu(wctypeD) = True Then Exit For
        Next b4
    Next a4

    'cek apakah nilai yg baru lebih kecil,
    'klo iya jadikan nilai terkecil yg baru
    If dblMinNow(wctypeD) = 0 Or dblMinCurr(wctypeD) <
    dblMinNow(wctypeD) Then
        dblMinNow(wctypeD) = dblMinCurr(wctypeD)
    End If
'END CUT!! =====

    frmSFC.txtPrinter(wctypeD).Text = frmSFC.txtPrinter(wctypeD).Text +
    Str(dblMinNow(wctypeD)) + vbCrLf
    arrRndCount(wctypeD) = arrRndCount(wctypeD) + 1 'berapa kali
    random

    'If blnAkhirRnd(wctypeD) = True Then
    '    frmSFC.tmrRandomD.Enabled = False
    '    frmSFC.txtPrinter(wctypeD).Text =
    frmSFC.txtPrinter(wctypeD).Text + "Selesai dg " & arrRndCount(wctypeD) &
    " kali random"
    '    arrRndCount(wctypeD) = 0
    'End If

    If arrRndCount(wctypeD) > randomLimit Then
        frmSFC.tmrRandomD.Enabled = False
        frmSFC.txtPrinter(wctypeD).Text =
        frmSFC.txtPrinter(wctypeD).Text + "Selesai dg " & arrRndCount(wctypeD) &
        " kali random"
        arrRndCount(wctypeD) = 0
    End If
End If
End Sub

Public Sub randomWCE(wctypeE)
    Dim a5, b5 As Integer
    For a5 = 1 To 6
        dblMinCurr(a5) = 0
    Next a5
    If blnLembur(wctypeE) = False Then

```

```

'CUT HERE!! =====
arrMax(wctypeE)          dblRangeBawah(wctypeE) = arrDemand(wctypeE) Mod
dblRangeBawah(wctypeE)   dblRangeAtas(wctypeE) = arrMax(wctypeE) -

=====
'BLOK ACAK DONG ADA DI SINI
=====
For a5 = 1 To intJmlHari * 2
    Randomize
    arrRandom(wctypeE, a5) =
Round(dblRangeBawah(wctypeE) + Rnd * (dblRangeAtas(wctypeE)))
    dblBuff(wctypeE) = dblBuff(wctypeE)
+ arrRandom(wctypeE, a5)
Next a5

Then
    If dblBuff(wctypeE) > arrDemand(wctypeE)
        dblBalancer(wctypeE) =
        (dblBuff(wctypeE) - arrDemand(wctypeE)) / (intJmlHari * 2)
        dblBuff(wctypeE) = 0
        For a5 = 1 To intJmlHari * 2
            arrRandom(wctypeE, a5) =
arrRandom(wctypeE, a5) - dblBalancer(wctypeE)
        Next a5
    Else
        ataslagi:
            dblBalancer(wctypeE) =
            (arrDemand(wctypeE) - dblBuff(wctypeE)) / (intJmlHari * 2)
            dblSisaBalancer(wctypeE) = 0
            dblBuff(wctypeE) = 0
            For a5 = 1 To intJmlHari * 2
                If arrMax(wctypeE) -
arrRandom(wctypeE, a5) > dblBalancer(wctypeE) Then
                    arrRandom(wctypeE, a5) =
arrRandom(wctypeE, a5) + dblBalancer(wctypeE)
                Else
                    dblSisaBalancer(wctypeE) =
dblSisaBalancer(wctypeE) + dblBalancer(wctypeE)
                End If
            Next a5
            If dblSisaBalancer(wctypeE) > 0 Then
                For a5 = 1 To intJmlHari * 2
                    dblBuff(wctypeE) =
dblBuff(wctypeE) + arrRandom(wctypeE, a5)
                Next a5
                dblBalancer(wctypeE) =
dblSisaBalancer(wctypeE)
                GoTo ataslagi
            Else
                'do nothing
            End If
        End If

        For a5 = 1 To intJmlHari * 2
            arrRandom(wctypeE, a5) =
Int(arrRandom(wctypeE, a5))
            dblBuff(wctypeE) = dblBuff(wctypeE)
+ arrRandom(wctypeE, a5)
        Next a5

        If arrDemand(wctypeE) > dblBuff(wctypeE)
            Then
                intCounter(wctypeE) = 0

```

```

arrRandom(wctypeE, a5) > ((arrDemand(wctypeE) - dblBuff(wctypeE)) / 2)
Then
    intCounter(wctypeE) + 1
    intCounter(wctypeE)
    arrRandom(wctypeE, a5) + Int(((arrDemand(wctypeE) - dblBuff(wctypeE)) / 2))
    arrRandom(wctypeE, a5) + (arrDemand(wctypeE) - dblBuff(wctypeE)) -
    Int(((arrDemand(wctypeE) - dblBuff(wctypeE)) / 2))
    Exit For
End Select
End If
Next a5
End If
dblBuff(wctypeE) = 0
'END ACAK
DONG=====

For a5 = 1 To intJmlHari * 2
    blnKetemu(wctypeE) = False
    For b5 = 1 To dblJmlRange
        lookUpE dblBatasBawah E(b5),
        dblBatasAtas_E(b5), dblPenalti_E(b5), a5, wctypeE, arrRandom(wctypeE,
        a5)
        If blnKetemu(wctypeE) = True Then Exit
    Next b5
Next a5

'cek apakah nilai yg baru lebih kecil,
'klo iya jadikan nilai terkecil yg baru
If dblMinNow(wctypeE) = 0 Or dblMinCurr(wctypeE)
< dblMinNow(wctypeE) Then
    dblMinNow(wctypeE) = dblMinCurr(wctypeE)
End If
'END CUT!!
=====

frmSFC.txtPrinter(wctypeE).Text =
frmSFC.txtPrinter(wctypeE).Text + Str(dblMinNow(wctypeE)) + vbCrLf
arrRndCount(wctypeE) = arrRndCount(wctypeE) + 1
'berapa kali random

If blnAkhrrnd(wctypeE) = True Then
    frmSFC.tmrRandomE.Enabled = False
    frmSFC.txtPrinter(wctypeE).Text =
    frmSFC.txtPrinter(wctypeE).Text + "Selesai dg " & arrRndCount(wctypeE) &
    " kali random"
    arrRndCount(wctypeE) = 0
End If

If arrRndCount(wctypeE) > randomLimit Then
    frmSFC.tmrRandomE.Enabled = False
    frmSFC.txtPrinter(wctypeE).Text =
    frmSFC.txtPrinter(wctypeE).Text + "Selesai dg " & arrRndCount(wctypeE) &
    " kali random"
    arrRndCount(wctypeE) = 0
End If

```

```

Else
    'EDIT TERBARU --- MAKSIMALKAN LEMBUR
    =====
    pengali(wctypeE) = Int(arrDemand(wctypeE) / arrMax(wctypeE))
    For a5 = 1 To pengali(wctypeE)
        arrRandom(wctypeE, a5) = arrMax(wctypeE)
    Next a5
    arrRandom(wctypeE, pengali(wctypeE) + 1) = arrDemand(wctypeE) -
    (arrMax(wctypeE) * pengali(wctypeE))
    'END CUT!!
    =====

    For a5 = 1 To intJmlHari
        blnKetemu(wctypeE) = False
        For b5 = 6 To dblJmlRange
            lookUpE dblBatasBawah_E(b5), dblBatasAtas_E(b5),
            dblPenalti_E(b5 - 5), a5, wctypeE, arrRandom(wctypeE, a5)
            If blnKetemu(wctypeE) = True Then Exit For
        Next b5
    Next a5

    'cek apakah nilai yg baru lebih kecil,
    'klo iya jadikan nilai terkecil yg baru
    If dblMinNow(wctypeE) = 0 Or dblMinCurr(wctypeE) <
    dblMinNow(wctypeE) Then
        dblMinNow(wctypeE) = dblMinCurr(wctypeE)
    End If
    'END CUT!! =====

    frmSFC.txtPrinter(wctypeE).Text = frmSFC.txtPrinter(wctypeE).Text +
    Str(dblMinNow(wctypeE)) + vbCrLf
    arrRndCount(wctypeE) = arrRndCount(wctypeE) + 1 'berapa kali
    random

    'If blnAkhrrRnd(wctypeE) = True Then
    '    frmSFC.tmrRandomE.Enabled = False
    '    frmSFC.txtPrinter(wctypeE).Text =
    frmSFC.txtPrinter(wctypeE).Text + "Selesai dg " & arrRndCount(wctypeE) &
    " kali random"
    '    arrRndCount(wctypeE) = 0
    'End If

    If arrRndCount(wctypeE) > randomLimit Then
        frmSFC.tmrRandomE.Enabled = False
        frmSFC.txtPrinter(wctypeE).Text =
        frmSFC.txtPrinter(wctypeE).Text + "Selesai dg " & arrRndCount(wctypeE) &
        " kali random"
        arrRndCount(wctypeE) = 0
    End If
End If

End Sub

Public Sub randomWCF(wctypeF)
    Dim a6, b6 As Integer

    For a6 = 1 To 6
        dblMinCurr(a6) = 0
    Next a6

    If blnLembur(wctypeF) = False Then
        'CUT HERE!! =====

```

```

arrMax(wctypeF)      dblRangeBawah(wctypeF) = arrDemand(wctypeF) Mod
                      dblRangeAtas(wctypeF) = arrMax(wctypeF) -
                      dblRangeBawah(wctypeF)

                      'BLOK ACAK DONG ADA DI SINI
                      =====
                      For a6 = 1 To intJmlHari * 2
                        Randomize
                        arrRandom(wctypeF, a6) =
Round(dblRangeBawah(wctypeF) + Rnd * (dblRangeAtas(wctypeF)))
                        dblBuff(wctypeF) = dblBuff(wctypeF) +
arrRandom(wctypeF, a6)
                      Next a6

Then
                      If dblBuff(wctypeF) > arrDemand(wctypeF)
                        dblBalancer(wctypeF) = (dblBuff(wctypeF)
- arrDemand(wctypeF)) / (intJmlHari * 2)
                        dblBuff(wctypeF) = 0
                        For a6 = 1 To intJmlHari * 2
                          arrRandom(wctypeF, a6) =
arrRandom(wctypeF, a6) - dblBalancer(wctypeF)
                        Next a6
                        Else
                          atasLagi:
                            dblBalancer(wctypeF) =
(arrDemand(wctypeF) - dblBuff(wctypeF)) / (intJmlHari * 2)
                            dblSisaBalancer(wctypeF) = 0
                            dblBuff(wctypeF) = 0
                            For a6 = 1 To intJmlHari * 2
                              If arrMax(wctypeF) -
arrRandom(wctypeF, a6) > dblBalancer(wctypeF) Then
                                arrRandom(wctypeF, a6) =
arrRandom(wctypeF, a6) + dblBalancer(wctypeF)
                              Else
                                dblSisaBalancer(wctypeF) =
dblSisaBalancer(wctypeF) + dblBalancer(wctypeF)
                              End If
                            Next a6
                            If dblSisaBalancer(wctypeF) > 0 Then
                              For a6 = 1 To intJmlHari * 2
                                dblBuff(wctypeF) =
dblBuff(wctypeF) + arrRandom(wctypeF, a6)
                              Next a6
                              dblBalancer(wctypeF) =
dblSisaBalancer(wctypeF)
                              GoTo atasLagi
                            Else
                              'do nothing
                            End If
                          End If
                        For a6 = 1 To intJmlHari * 2
                          arrRandom(wctypeF, a6) =
Int(arrRandom(wctypeF, a6))
                          dblBuff(wctypeF) = dblBuff(wctypeF) +
arrRandom(wctypeF, a6)
                        Next a6
                        If arrDemand(wctypeF) > dblBuff(wctypeF)
Then
                          intCounter(wctypeF) = 0
                          For a6 = 1 To intJmlHari * 2

```

```

arrRandom(wctypeF, a6) > ((arrDemand(wctypeF) -
Then                                     If arrMax(wctypeF) -
                                         dblBuff(wctypeF)) / 2)

intCounter(wctypeF) + 1                 intCounter(wctypeF) =

                                         Select Case intCounter(wctypeF)
                                         Case 1
arrRandom(wctypeF, a6) + Int(((arrDemand(wctypeF) - dblBuff(wctypeF)) /
2))                                     arrRandom(wctypeF, a6) =

                                         Case 2
arrRandom(wctypeF, a6) + (arrDemand(wctypeF) - dblBuff(wctypeF)) -
Int(((arrDemand(wctypeF) - dblBuff(wctypeF)) / 2))   arrRandom(wctypeF, a6) =
                                         Exit For
                                         End Select
                                         End If
                                         Next a6
                                         End If
                                         dblBuff(wctypeF) = 0

'END ACAK DONG=====

For a6 = 1 To intJmlHari * 2
    blnKetemu(wctypeF) = False
    For b6 = 1 To dblJmlRange
        lookUpF dblBatasBawah F(b6),
        dblBatasAtas_F(b6), dblPenalti_F(b6), a6, wctypeF, arrRandom(wctypeF,
        a6)
        If blnKetemu(wctypeF) = True Then Exit For
    Next b6
Next a6

'cek apakah nilai yg baru lebih kecil,
'klo iya jadikan nilai terkecil yg baru
If dblMinNow(wctypeF) = 0 Or dblMinCurr(wctypeF) <
dblMinNow(wctypeF) Then
    dblMinNow(wctypeF) = dblMinCurr(wctypeF)
End If
'END CUT!!
=====

frmSFC.txtPrinter(wctypeF).Text =
frmSFC.txtPrinter(wctypeF).Text + Str(dblMinNow(wctypeF)) + vbCrLf
arrRndCount(wctypeF) = arrRndCount(wctypeF) + 1
'berapa kali random

'If blnAkhirRnd(wctypeF) = True Then
'    frmSFC.tmrRandomF.Enabled = False
'    frmSFC.txtPrinter(wctypeF).Text =
frmSFC.txtPrinter(wctypeF).Text + "Selesai dg " & arrRndCount(wctypeF) &
" kali random"
'    arrRndCount(wctypeF) = 0
'End If

frmSFC.lblProgress.Width = arrRndCount(wctypeF) *
(barWidth / randomLimit)
If arrRndCount(wctypeF) > randomLimit Then
    frmSFC.tmrRandomF.Enabled = False
    frmSFC.txtPrinter(wctypeF).Text =
frmSFC.txtPrinter(wctypeF).Text + "Selesai dg " & arrRndCount(wctypeF) &
" kali random"
    arrRndCount(wctypeF) = 0
    frmSFC.tmrLimit.Enabled = True
End If

```

```

'EDIT TERBARU --- MAKSIMALKAN LEMBUR
=====
pengali(wctypeF) = Int(arrDemand(wctypeF) / arrMax(wctypeF))
For a6 = 1 To pengali(wctypeF)
    arrRandom(wctypeF, a6) = arrMax(wctypeF)
Next a6
arrRandom(wctypeF, pengali(wctypeF) + 1) = arrDemand(wctypeF) -
(arrMax(wctypeF) * pengali(wctypeF))
'END CUT!!
=====
For a6 = 1 To intJmlHari
    blnKetemu(wctypeF) = False
    For b6 = 6 To dblJmlRange
        lookUpF dblBatasBawah_F(b6), dblBatasAtas_F(b6),
        dblPenalti_F(b6 - 5), a6, wctypeF, arrRandom(wctypeF, a6)
        If blnKetemu(wctypeF) = True Then Exit For
    Next b6
Next a6

'cek apakah nilai yg baru lebih kecil,
'klo iya jadikan nilai terkecil yg baru
If dblMinNow(wctypeF) = 0 Or dblMinCurr(wctypeF) <
dblMinNow(wctypeF) Then
    dblMinNow(wctypeF) = dblMinCurr(wctypeF)
End If
'END CUT!! =====

frmSFC.txtPrinter(wctypeF).Text = frmSFC.txtPrinter(wctypeF).Text +
Str(dblMinNow(wctypeF)) + vbCrLf
arrRndCount(wctypeF) = arrRndCount(wctypeF) + 1 'berapa kali
random

'If blnAkhirRnd(wctypeF) = True Then
'    frmSFC.tmrRandomF.Enabled = False
'    frmSFC.txtPrinter(wctypeF).Text =
frmSFC.txtPrinter(wctypeF).Text + "Selesai dg " & arrRndCount(wctypeF) &
" kali random"
'    arrRndCount(wctypeF) = 0
'End If

frmSFC.lblProgress.Width = arrRndCount(wctypeF) * (barWidth /
randomLimit)
If arrRndCount(wctypeF) >= randomLimit Then
    frmSFC.tmrRandomF.Enabled = False
    frmSFC.txtPrinter(wctypeF).Text =
frmSFC.txtPrinter(wctypeF).Text + "Selesai dg " & arrRndCount(wctypeF) &
" kali random"
    arrRndCount(wctypeF) = 0
    frmSFC.tmrLimit.Enabled = True
End If
End If

End Sub
Public Sub convBulan()
    Select Case intToMonth
        Case 1 strMonth = "Jan"
        Case 2 strMonth = "Feb"
        Case 3 strMonth = "Mar"
        Case 4 strMonth = "Apr"
        Case 5 strMonth = "May"
        Case 6 strMonth = "Jun"
        Case 7 strMonth = "Jul"
        Case 9 strMonth = "Aug"
        Case 8 strMonth = "Sep"
        Case 10 strMonth = "Oct"
        Case 11 strMonth = "Nov"
        Case 12 strMonth = "Dec"
    End Select
End Sub

```