บรรณานุกรม

เจริญ สุนทรวาณิชย์. **การวางแผนการผลิตและพัสดุคงคลังสำหรับโรงงานกระดาษเหนียว.** วิทยานิพนธ์มหาบัณฑิต ภาควิชาวิศวกรรมอุตสาหการ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย,2929.

ดุษฎี ตัณฑโกไศย และคณะ. **การสร้างระบบผู้เชี่ยวชาญอย่างง่าย.** วิศวกรรมสาร เล่มที่ 3 ประจำปี 2531.หน้า 59-62

ดำริห์ พงษ์ปาลิต **อุตสาหกรรมอาหารสัตว์.** วารสารเศรษฐกิจ ธนาคารกรุงเทพ จำกัด ปีที่ 22 มิถุนายน 2533. หน้า 284-293

ศจีรัตน์ เลิศเบญจรงค์. **ปัญหาการผลิตของอุตสาหกรรมอาหารสัตว์ในประเทศไทย.** วิทยานิพนธ์มหาบัณฑิต ภาควิชาการบัญชี บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2522.

ศิริจันทร์ ทองประเสริฐ. **การจำลองแบบปัญหา.** ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย.

สุโกวิท โชติวัฒนกุล. **ประสิทธิภาพการผลิตของอุตสาหกรรมอาหารสัตว์ในประเทศไทย** วิทยานิพนธ์มหาบัณฑิต คณะเศรษฐศาสตร์ บัณฑิตวิทยาลัย มหาวิทยาลัยธรรมศาสตร์, 2530.

อุทัย คันโธ. **อาหารและการผลิตอาหารเลี้ยงสุกรและสัตว์ปีก.** มหาวิทยาลัยเกษตรศาสตร์ นครปฐม พิมพ์ครั้งที่ 1, 2529.

Adam Wolsz. Production rate optimization in a two-stage system with finite intermediate storage. **European Journal of Operational Research 18.** 369-376, 1984.

Andrew Kusiak, and Mingyuan Chen. Expert System for Planning and Scheduling Manufacturing Systems. **European Journal of Operation Research 34.** 113-130, 1988.

Baker, K, R. **Introduction to Sequencing and Scheduling.** John Wiley. New York, 1974.

Cihan H. Dagli, and Tatchapol Poshyanonda. **Heuristic Generation in Knowledge-Based Schdeuling Through Simulation.** Department of Engineering Management. University of Missouri-Rolla, 1988.

Cihan H. Dagli, and Tatchapol Poshyanonda. **Knowledge-Based Scheduling and Control.** Department of Engineering Management. University of Missouri-Rolla, 1989.

David A. White. **PCModel.** Personal Compuler Character Graphic Modelling and Simubtion System User's Guide. Version 8, Simulation Softeare Systems, Inc.,1988

Geoffrey Gordon. **System Simulation.** 2nd. Ed., Prentice-Hall. Inc., 1987.

Janes E. Day, and Michael P. Hottenstcin. **Review of Sequencing Research.**

Jatinder N.D. Gupta. Heruristic Algorithms for Multistage Flowship Scheduling Problem. **AIIE Transactions.** 11-18, March 1972

John D. Yanney, and Way Kuo. A Practical Approach to Scheduling a Multistage Multiprocessor Flow-shop Problem . **International Journal of Production Research.**1989. vol.27, No.10, 1733-1742.

Joseph J. Moder, and Salah E. Elmaghraby. **Handbook of Operations Research Model and Applications.** Van Nostrand Reinhold Company, 1978.

Richard W. Conway, William L. M., and Louis W. M. **Theory of Scheduling.** Addison Wesley Publishing Company, 1967.

Robert M., O'Keefe, Osman Balci, and Eric P. Smith. Validating Expert System Performance. **IEEE .** winter, 1987.

Simon French. **Sequencing and Scheduling : An introduction to the Mathematics of the Job-Shop.** Ellis Horwood Ltd, 1982.

Stanley B. Gershwin, and Irvin C. Schick. **Modelling and Analysis of Two-and Three-Stage Transfer lines with Unreliable Machines and Finite Buffer.** Massachusetts Institute of technology. Cambridge, Mass. 02139.

Tatchapol Poshyanonda. **Integration of Artificial Intelligence and Operations Research in Production Planning, Scheduling and Control.** University of Missouri-Rolla, 1989.

ภาคผนวก

ภาคผนวก ก

วัตถุดิบที่ใช้ในการผลิตอาหารสัตว์

ตารางแสดงวัตถุดิบที่ใช้ในการผลิตอาหารสัตว์

ลำดับที่	ชื่อภาษาอังกฤษ	ชื่อภาษาไทย	ชื่อย่อ
1		เปลือกหอยลายป่น	
2	ACID OIL	แอซิด ออยล์	AO
3	ALFA STARCH	อัลฟ่าสตาร์ช	ALFA,STCH
4	BLACKBEAN MEAL,EXPELLER	กากถั่วดำ	BOM,E
5	BLACKBEAN MEAL,SOLVENT	เกล็ดถั่วดำ	BOM,S
6	BLOOD MEAL	เลือดป่น	
7	BONE MEAL	กระดูกป่น	BM
8	BRASILIAN SOYBEAN MEAL, PELLETED	เกล็ดถั่วเหลืองบราซิล อัดเม็ด	SOM,B P
9	BRAZILIAN SOYBEAN MEAL,SOLVENT	เกล็ดถั่วเหลืองบราซิล	SOM,B
10	BREWER'S DRIED GRAINS	กากเบียร์	BDG
11	BREWER'S DRIED YEAST		
12	BROKEN GLUTINOUS RICE	ปลายข้าวเหนียว	BGR
13	BROKEN RICE	ปลายข้าวจ้าว	BR
14	CASEIN		
15	CASSAVA MEAL	มันเส้น	CASM
16	CASSAVA MEAL,PELLET	มันอัดเม็ด	CASM,P
17	CHICKEN HEAD MEAL	หัวไก่ป่น	CHM
18	COCONUT MEAL,EXPELLER	กากมะพร้าว	COM,E
19	COCONUT MEAL,SOLVENT	เกล็ดมะพร้าว	COM,S
20	COCONUT OIL	น้ำมันมะพร้าว	CO-OIL
21	CORN GLUTEN MEAL	คอนกลูเตน	CGM
22	COTTONSEED MEAL	กากฝ้าย	CSM
23	COTTONSEED MEAL,PELLET	กากฝ้ายอัดเม็ด	CSM,P

ตารางแสดงวัตถุดิบที่ใช้ในการผลิตอาหารสัตว์ (ต่อ)

ลำดับที่	ชื่อภาษาอังกฤษ	ชื่อภาษาไทย	ชื่อย่อ
24	CRUDE PALM OIL	น้ำมันปาล์ม	CPO
25	CRUDE RICE BRAN OIL	น้ำมันรำ	RBO
26	CRUDE SOYBEAN OIL	น้ำมันถั่วเหลือง	CSO
27	CUTTLEFISH MEAL	ปลาหมึกป่น	CUT-FM
28	DICALCIUM PHOSPHATE,IMPORT	ดีซีพีนอก	DCP,I
29	DISTILLERS GRAINS	ส่าเหล้า	
30	DRIED SKIM MILK	นมผงขาดมันเนย, หางนม	DSM
31	DRIED WHEY	เวย์	WHEY
32	EXTRUDED BROKEN RICE	ปลายข้าวนึ่ง	EXT,BR
33	EXTRUDED CORN	ข้าวโพดนึ่ง	EXT.CORN
34	EXTRUDED CORN+BROKEN RICE	ปลายข้าวผสมข้าวโพดนึ่ง	EXT.CORN/BR
35	EXTRUDED MEAT& BONE MEAL	เนื้อกระดูกป่นนึ่ง	EXT.M&B
36	EXTRUDED SOYBEANS	ถั่วนึ่ง	EXT.SB
37	FISH HEAD MEAL	หัวปลาป่น	FHM
38	FISH LIVER MEAL	ไส้ปลาป่น	FLM
39	FISH MEAL	ปลาป่น	FM
40	FISH OIL	น้ำมันปลา	FO
41	FRESH FISH	ปลาสด	FISH
42	GRIT	หินเกล็ด	GRIT
43	IND. SOYBEAN MEAL,SOLVENT	เกล็ดถั่วเหลืองอินเดีย	SOM,I
44	IPIL IPIL MEAL	กระถินป่น	IPIL
45	JAP. SOYBEAN MEAL,SOLVENT	เกล็ดถั่วเหลืองญี่ปุ่น	SOM,J
46	KAPOK MEAL	กากนุ่น	KOM
47	LARD	น้ำมันหมู	LARD

ตารางแสดงวัตถุดิบที่ใช้ในการผลิตอาหารสัตว์ (ต่อ)

ลำดับที่	ชื่อภาษาอังกฤษ	ชื่อภาษาไทย	ชื่อย่อ
49	MEAT AND BONE MEAL	เนื้อกระดูกป่น	MBM
50	MILO,(SORGHUM),RED	ข้าวฟ่างแดง	MR
51	MILO,(SORGHUM),WHITE	ข้าวฟ่างขาว	MW
52	MILO,(SORGHUM),YELLOW	ข้างฟ่างเหลือง	MY
53	MOLASSES	กากน้ำตาล	
54	MONOCALCIUM PHOSPHATE	โมโนแคลเซียมฟอสเฟต	MCP
55	MUNGBEAN MEAL	กากถั่วเขียว	MUNG
56	OYSTER SHELL	เปลือกหอยนางรมป่น	OS
57	PADDY	ข้าวเปลือก	
58	PALM KERNEL MEAL	กากเมล็ดปาล์ม	PKM
59	PALM MEAL	กากปาล์ม	PM
60	PEANUT MEAL,EXPELLER	กากถั่วลิสง	POM,E
61	PEANUT MEAL,SOLVENT	เกล็ดถั่วลิสง	POM,S
62	PLASTIC CLAY	พลาสติกเคลย์	PL.CLAY
63	POULTRY BY PRODUCT		
64	RAPESEED MEAL	กากองุ่น	RSM
65	RICE BRAN,COARSE	รำหยาบ	RBC
66	RICE BRAN,FRESH	รำขาว, รำสด	RBF
67	RICE BRAN,PELLET	รำขาวอัดเม็ด	RBF,P
68	RICE BRAN,SOLVENT	รำอัด	RBS
69	RICE BRAN,SOLVENT,PELLET	รำอัดอัดเม็ด	RBS,P
70	RICE FLOUR	แป้งข้าวข้าว	RF
71	RICE HULLS	แกลบ	RH
72	RUBBER SEED MEAL	กากเมล็ดยาง	RUBM
73	SALT	เกลือ	

ตารางแสดงวัตถุดิบที่ใช้ในการผลิตอาหารสัตว์ (ต่อ)

ลำดับที่	ชื่อภาษาอังกฤษ	ชื่อภาษาไทย	ชื่อย่อ
74	SEMOLINA FLOUR		
75	SESAME SEED MEAL	กากงา	SSM
76	SHRIMP MEAL	กุ้งป่น	SM
77	SHRIMP SHELL MEAL	เปลือกกุ้งป่น	SHRM
78	SOYBEAN	ถั่วเหลืองเม็ด	SB
79	SOYBEAN MEAL,EXPELLER	กากถั่วเหลือง	SOM,E
80	SOYBEAN MEAL,SOLVENT	เกล็ดถั่วเหลือง	SOM,S
81	SUNFLOWER SEED MEAL	กากเมล็ดทานตะวัน	SFM
82	TALLOW	ไขมันวัว	
83	TAPIOCA OR CASSAVA STARCH, COARSE	แป้งมันหยาบ	
84	TAPIOCA OR CASSAVA STARCH, FINE	แป้งมันละเอียด	
85	THAI BONE MEAL	กระดูป่น อ.ส.ก.	TBM
86	THAI PHOSPHATE	ดีซีพีไทยแลนด์	TP
87	TRICALCIUM PHOSPHATE	ไตรแคลเซี่ยมฟอสเฟส	TCP
88	UNPOLISHED BROKEN RICE	ปลายข้าวกล้อง	UBR
89	UREA		
90	WHEAT BRAN	รำข้าวสาลี	WB
91	WHEAT FLOUR	แป้งข้าวสาลี	WF
92	WHEAT GRAIN	ข้าวสาลี	WG
93	WHEAT MIDDLINGS		
94	YELLOW CORN	ข้าวโพด	YC

ภาคผนวก ข

ผลิตภัณฑ์ในปัจจุบันของโรงงาน

ตารางแสดงผลิตภัณฑ์ในปัจจุบันของโรงงาน

1.หมวดอาหารไก่ (อาหารสำเร็จรูปชนิดผง)			สำหรับ	บรรจุ (กก.)
ไฮโปรไวท	แอนไวโปร์	ซี.พี.		
324	424	824	ไก่ไข่	30
2. หมวดอาหารไก่ (สำเร็จรูปชนิดเม็ด)			สำหรับ	บรรจุ (กก.)
ไฮโปรไวท์	แอนไวโปร์	ซี.พี.		
510	610	910	ไก่เล็ก-เนื้อพิเศษ	30
511	611	911	ไก่เล็ก-เนื้อเล็ก	30
513	613	913	ไก่เล็ก-เนื้อสุดท้าย	30
521	621	921	ไก่รุ่นไข่(จาก 0-6 อาทิตย์)	30
522	622	922	ไก่รุ่นไข่(จาก 7-12 อาทิตย์)	30
523	623	923	ไก่ใหญ่ไข่(จาก 13-22 อาทิตย์)	30
524	624	924	ไก่ไข่ (23 อาทิตย์ขึ้นไป)	30
526	625	925	ไก่ไข่	30
3. หมวดอาหารหมู (หัวอาหาร)			สำหรับ	บรรจุ (กก.)
ไฮโกร	ฮ๊อกโทนัล	ซี.พี.		
151	251	751	หมูรวม	30
152	252	752	หมูเล็ก	30

ตารางแสดงผลิตภัณฑ์ในปัจจุบันของโรงงาน (ต่อ)

3.หมวดอาหารหมู (หัวอาหาร)			สำหรับ	บรรจุ (กก.)
ไฮโกร	ฮ๊อกโทนัล	ซี.พี.		
153	253	753	หมูรุ่น	30
155	255	755	หมูเนื้อ	30
156	256	756	หมูพันธุ์ระยะอุ้มท้อง	30
4. หมวดอาหารหมูขุน (สำเร็จรูปชนิดเม็ด)			สำหรับ	บรรจุ (กก.)
ไฮโกร	ฮ๊อกโทนัล	ซี.พี.		
551	651	951	หมูอ่อน	30
552	652	952	หมูเล็ก	30
553	653	953	หมูรุ่นขุน	30
554	654	954	หมูใหญ่	30
555	655	955	หมูเนื้อ	30
4. หมวดอาหารหมูพันธุ์ (สำเร็จรูปชนิดเม็ด)			สำหรับ	บรรจุ (กก.)
ไฮโกร	ฮ๊อกโทนัล	ซี.พี.		
566	666	966	หมูพันธุ์	30
567	667	967	แม่หมูระยะให้นม	30

ตารางแสดงผลิตภัณฑ์ในปัจจุบันของโรงงาน(ต่อ)

6.หมวดอาหารส่งฟาร์ม
324ND
324NDZ06
324NDZ09
324NDZ16
325ND
325NDZ06
325NDZ09
326ND
335ND
521N
522N
551NS
551X05
552NS
552X09
567NS
567X17
531N
532N
534ND

ภาคผนวก ค

รายงานการผลิตของโรงงาน

โปรแกรมผลิต

วันที่ 17 / 1 / 34

ลำดับ	รหัส	ต้น	ถัง
1	552×09	40	S. 36
2	524	26	รวม
3	525'	40	
4	555	50	
5	511	48	
6	553	50	
7	510	48	S. 9
8	534ND	15	S.
9	552	50	
10			
11			
12	325ND	16	รวม
13	324ND	~~24~~	
14		48	
15			
16			
	รวม		ต้น

ยอดอาหารค้างถัง			
ถัง	จำนวน (กก.)	ถัง	จำนวน (กก.)
17	555 = ลบ	21	552 = ลบ
18	552 = 50,000	22	555 = 49730
19	567 = 20000	23	513 = 36000
20	553 = 13730	24	552×09 = 14670

หมายเหตุ	
	25 534ND = ลบ
	26 513 = ลบ
	27 534ND = 15000

ประจำวันที่ 18-01-34 ผลัดที่ 2.

รายงานการผสม

รหัส	No. สูตร	วันที่สูตร	จำนวน (ตัน)	REMIX (Kgs).	ถังปั้มเม็ด	ถังบรรจุ	เวลาผสม เริ่ม	เวลาผสม หยุด	ช่วงเวลาเสีย เริ่มหยุด	ช่วงเวลาเสีย เริ่มเดิน	เวลาสูญเสีย เปลี่ยนสูตร	รอบด	รอวัตถุดิบ	เครื่องเสีย	อื่น ๆ	หมายเหตุ
									16 00	16 30		50				3129
552	5552	11-1-34	25	-	1,3,4	18 21	16 30	17 22	17 22	17 28	33				10	4129
552X	9595	11-1-34	50	-	15 16	24 22	17 28	17 33	17 33	17 50				28		Cleaner 4267
							17 50	19 17	19 17	19 38					35	4131
551	5551	11-1-34	51	-	4,3,2	18 17	19 38	21 12	21 12	21 20	13					4126
511	1511	11-1-34	60	-	15 16	21	21 20	23 10	23 10	23 20	17					4126
551X	5595	11-1-34	24	-	1 02		23 20	24 00	24 00							
		รวม	210	-						รวมเวลาเสีย	30	50		28	45	1 22

ผู้ควบคุมการผสม

P-PR-01 ต้นฉบับ – สถิติ

ใบรายงานปั๊มเม็ด

ประจำวันที่ 18/01/64 ผลัดที่ 3

รหัส	จำนวนผลิต			หมายเลขถัง				เครื่องเดิน	เวลาปั๊มเม็ด				ช่วงเวลาที่เสีย				เก็บอาหาร	เวลาสูญเสีย						เวลาเสียอื่นๆ		รายละเอียดเครื่องเสียและเสียเวลาอื่นๆ	ขนาดตะแกง มม.	P.I	Rpm	°C	Amp
	หมู	ไก่	รวมนิมิตร์	จากถัง		ลงถัง			เริ่ม		หยุด		เริ่มหยุด		เริ่มเดิน			รอผสม		รอบรรจุ		เครื่องเสีย								
711		30	60000	15	16	21	22	2.03	14	12	33	30	01	30	32	45	0.25													
													01	30	32	45		0.48		0.42										
716	24		14000	15	16	18		1.03.	33	45	34	30	03	35	33	55		0.33												
			1										04	30	04	45	.25.													
													04	45	—	—														

(รวม)	24	30	54000																											

รวมเวลาเสีย 0.50 | 0.33 | 0.42 | 1.55 | ผู้ควบคุม

หมู่อ่อน	
หมายเลขตะแกรง	
ยอด PELLET / ผลัด	
ยกมา	
สะสม	

หมายเลขตะแกรง	23163/4
ยอด PELLET / ผลัด	54000
ยกมา	404490.
สะสม	440890.

M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H	M	H
1	.02	6	.10	11	.18	16	.27	21	.35	26	.43	31	.52	36	.60	41	.68	46	.77	51	.85	56	.93		
2	.03	7	.12	12	.20	17	.28	22	.37	27	.45	32	.53	37	.62	42	.70	47	.79	52	.87	57	.95		
3	.05	8	.13	13	.22	18	.30	23	.38	28	.47	33	.55	38	.63	43	.72	48	.80	53	.88	58	.97		
4	.07	9	.15	14	.23	19	.32	24	.40	29	.48	34	.57	39	.65	44	.73	49	.82	54	.90	59	.98		
5	.08	10	.17	15	.25	20	.33	25	.42	30	.50	35	.58	40	.67	45	.75	50	.83	55	.92	60	1.00		

*M = นาที
*H = ชั่วโมง

R-03

ตัวบรรจุที่ 3 ประจำวันที่ 18-1-34 ผลัด 2

รายงานการบรรจุ

รหัส	ถัง	COUNTER คัดทิ้ง REMIX	ตัด	กก.	หมายเลข COUNTER เริ่ม	หยุด	จำนวนผลิต (ถุง)	ช่วงเวลาบรรจุ เริ่ม	หยุด	ช่วงเวลาที่เสีย เริ่มหยุด	เริ่มเดิน	ไม่มีอาหาร	ไม่มีที่กอง	ไฟฟ้าดับ	เปลี่ยนเบอร์	เปลี่ยนกอง	ปัญหาน้ำหนัก	[illegible]	อื่น ๆ		หมายเหตุ และ รายละเอียด เครื่องจักรเสีย
										18.00	18.50								.50		
653	22				----	450	450	18.50	18.20	18.20	18.25				.05						
553	22		1	0		503	52	18.25	18.35	18.35	18.40				.05						
952	19					803	300	18.40	19.20	19.20	19.45				.35						
652	14		1	0		1115	311	19.40	20.20	20.20	20.25				.08						
913	23	1				1416	300	20.25	21.00	21.00	22.00				1.00						
613	23					1716	300	22.00	22.50	22.50	22.55				.05						
513	23		1	0		1336	619	22.55	24.15	24.15	24.30				.25						
951	18					2636	300	24.30	01.10	01.10	01.25				.15						
651	18		1			2791	154	01.20	01.40	01.40	01.50				12						
611	22		1	2		3114	322	01.50	02.30	02.30											
																					แก้ก้าน ชำรุด
รวม		1	4+1				3108								224				50		2.71

[illegible] - สถิติ

ผู้บรรจุ สายชล + เกียรติ ผู้ควบคุม [illegible]

PR-04

ภาคผนวก ง

เวลาในการลำเลียงอาหารสัตว์แต่ละช่วง

เวลาที่ใช้ในการลำเลียงอาหารสัตว์ในแต่ละช่วง

ช่วงลำเลียง	เวลา(วินาที)
จากเครื่องผสมถึงถังรอบีมเม็ด 15	31.26
จากเครื่องผสมถึงถังรอบีมเม็ด 16	28.88
จากเครื่องผสมถึงถังรอบีมเม็ด 1	42.58
จากเครื่องผสมถึงถังรอบีมเม็ด 2	41.19
จากเครื่องผสมถึงถังรอบีมเม็ด 3	45.17
จากเครื่องผสมถึงถังรอบีมเม็ด 4	44.16
จากเครื่องผสมถึงถังรอบรรจุ 17	44.32
จากเครื่องผสมถึงถังรอบรรจุ 21	44.32
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 17	48.94
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 21	47.62
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 25	46.36
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 18	45.05
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 22	42.53
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 26	41.22
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 19	40.01
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 23	38.7
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 27	37.49
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 20	36.18
จากเครื่องบีมเม็ด 1 ถึงถังรอบรรจุ 24	34.94
จากเครื่องบีมเม็ด 2 ถึงถังรอบรรจุ 17	55.75
จากเครื่องบีมเม็ด 2 ถึงถังรอบรรจุ 21	54.26
จากเครื่องบีมเม็ด 2 ถึงถังรอบรรจุ 25	52.88

เวลาที่ใช้ในการลำเลียงอาหารสัตว์ในแต่ละช่วง (ต่อ)

ช่วงลำเลียง	เวลา(วินาที)
จากเครื่องปั๊มเม็ด 2 ถึงถังรอบรรจุ 18	51.51
จากเครื่องปั๊มเม็ด 2 ถึงถังรอบรรจุ 22	49.45
จากเครื่องปั๊มเม็ด 2 ถึงถังรอบรรจุ 26	47.39
จากเครื่องปั๊มเม็ด 2 ถึงถังรอบรรจุ 19	45.78
จากเครื่องปั๊มเม็ด 2 ถึงถังรอบรรจุ 23	43.72
จากเครื่องปั๊มเม็ด 2 ถึงถังรอบรรจุ 27	41.66
จากเครื่องปั๊มเม็ด 2 ถึงถังรอบรรจุ 20	40.17
จากเครื่องปั๊มเม็ด 1 ถึงถังรอบรรจุ 24	38.57

ภาคผนวก ข

กฎเกณฑ์ที่ได้จากประสบการณ์ของผู้เชี่ยวชาญในการผลิตอาหารสัตว์

กฎเกณฑ์ที่ได้จากประสบการณ์ของผู้เชี่ยวชาญในโรงงานอาหารสัตว์

1. พยายามไม่หยุดเครื่องปีมเม็ด ในกรณีที่กำลังผสมอาหารผงอยู่เมื่อถังรอปีมเม็ดมีปริมาณลดลงให้เครื่องผสมอาหารกลับมาผสมอาหารชนิดเม็ดอีก เพราะจะทำให้ไม่ต้องหยุดเครื่องปีมเม็ดเพื่อรออาหาร

2. การชั่งอาหารเพื่อส่งมายังเครื่องผสมอาหาร จะพยายามชั่งอาหารเบอร์แรกในโปรแกรมการผลิตไปเรื่อย ๆ และจะผลิตอาหารเบอร์เดียวกันให้หมดเป็น lot (หลีกเลี่ยงการเปลี่ยนไปผสมอาหารเบอร์อื่น)

3. กรณีเป็น batch แรกของการผสมและเป็นอาหารชนิดเม็ด ให้เลือกลงถังรอปีมเม็ดที่ว่าง ถ้าเป็น batch ต่อ ๆ ไป ให้ลงในถังที่มีอาหารเบอร์นั้น ๆ อยู่จนเต็มถัง แล้วจึงเลือกไปลงถังอื่น

4. การเลือกผสมอาหารผงหรือหัวอาหารจะทำเมื่อถังรอปีมเม็ดไม่ว่าง ถังรอบรรจุที่มีสำหรับใส่อาหารผงมี 2 ถัง คือถังหมายเลข 17 และ 21

การเลือกลงถังมี 2 กรณีด้วยกัน

1. กรณีที่ผลิตอาหารผงหรือหัวอาหาร เพื่อบรรจุลงถุงให้เลือกลงถังรอบรรจุ 17 ก่อน
2. กรณีที่ผลิตอาหารผงหรือหัวอาหาร เพื่อลงรถไซโลให้เลือกลงถังรอบรรจุ 21 ก่อน

5. การส่งอาหารที่ผสมแล้วไปยังถังรอปีมเม็ด พยายามเลือกเครื่องปีมเม็ดเครื่อง 1 ก่อนส่งผลให้เลือกถัง 15 และถัง 16 ก่อนเลือกลงถัง 1 2 3 และ 4 ระหว่างถัง 15 และถัง 16 เลือกลงถัง 16 ก่อน สำหรับเครื่องปีมเม็ดเครื่อง 2 เลือกลงถังตามลำดับดังนี้ 1 4 3 และ 2

6. เมื่อกำลังทำการผลิตอาหารผงหรือหัวอาหารจะพยายามทำจนหมด lot แล้วจึงสลับกลับมาทำอาหารเม็ด

7. การเลือกอาหารจากถังรอปีมเม็ด เพื่อลงเครื่องปีมเม็ดในการปีมอาหาร batch แรกเลือกอาหารจากถังที่มีอาหารน้อยที่สุดก่อนหลัง จากนั้นทำการปีม batch ต่อ ๆ ไปจนหมดถังแล้วจึงไปเช็คอาหารเบอร์เดียวกันในถังอื่น

8. การผสมอาหารเพื่อส่งไปยังเครื่องปั๊มเม็ดจะพยายามส่งอาหารชนิดเดียวกันไปยังเครื่องเดียวต่อเนื่องกัน จะได้ไม่ต้องเสียเวลาเปลี่ยนตะแกรงกลับไปมาที่เครื่องปั๊มเม็ด

9. การเลือกถังรอบรรจุ จากจำนวนถังรอบรรจุทั้งหมด 11 ถัง คือถังหมายเลข 17 21 25 18 22 26 19 23 27 20 และ 24 เลือกดังนี้

ก) เลือกลงถัง 18 และ 22 ก่อน (เนื่องจากถังรอบรรจุ 2 ถังนี้สามารถลงหัวบรรจุได้ 3 หัว จึงทำให้ถ่ายอาหารออกได้เร็ว จึงทำให้ถังว่างเร็ว) ระหว่างถัง 18 และ 22 ถ้าต้องใช้ถังเดียวถือว่ามีprority เท่ากัน เลือกลงถังไหนก่อนก็ได้แต่ถ้าต้องใช้ทั้ง 2 ถัง (เนื่องจากอาหารมีมาก)

ข) ถ้ามาจากเครื่องปั๊มเม็ด # 1 เลือกลงถัง 22 เต็มก่อน แล้วลงถัง 18

ค) ถ้ามาจากเครื่องปั๊มเม็ด # 2 เลือกลงถัง 18 เต็มก่อน แล้วลงถัง 22 ทั้งนี้เนื่องจากระบบการติดตั้งถังตอนเริ่มเปิดโรงงาน

ง) เลือกลงถัง 19 และ 23 ถัดมา ถ้าถัง 18 และ 22 ไม่สามารถลงอาหารได้อีก โดยที่ถัง 19 และ 23 มี prioity เท่ากัน ถ้าต้องใช้ถังเดียวเลือกลงถังไหนก่อนก็ได้ (แต่จากการศึกษาลักษณะการวางถังน่าจะเลือกลงถัง 23 ก่อน แล้วจึงเลือกลงถัง 19) ถ้าต้องใช้ถังทั้ง 2 ถัง ทั้งอาหารที่มาจากเครื่องปั๊มเม็ดเครื่อง 1 และเครื่องปั๊มเม็ดเครื่อง 2 จะเลือกลงถัง 23 ก่อน แล้วจึงไปลงถัง 19

จ) กรณีที่ไม่สามารถลงอาหารชนิดเม็ดในถัง 18 22 19 และ 23 ได้ และเป็นอาหารที่ต้องบรรจุถุง ถังที่จะเลือกต่อไปมีถัง 17 21 25 และ 26

ฉ) โดยถัง 17 และ 21 มีความสำคัญเท่ากัน (เลือกลงถัง 21 ก่อน เพราะถัง 21 สามารถถ่ายอาหารออกได้เร็วกว่า)

ช) ถัง 25 และ 26 มีความสำคัญเท่ากัน (เลือกลงถัง 26 ก่อนตามลักษณะการวางถัง)

10. จากลักษณะการเรียงถังบรรจุ เป็นดังนี้

ก) 17 21 25 18 22 26 19 23 27 20 24 ถ้าต้องใช้ถังรอบรรจุตั้งแต่ 2 ถังขึ้นไป ทั้งอาหารที่มาจากเครื่องปั๊มเม็ดเครื่อง 1 และเครื่องปั๊มเม็ดเครื่อง 2 อาหารจะลงถังที่อยู่ทางขวาก่อนจนเต็ม แล้วจึงเลื่อนไปลงถังที่อยู่ทางซ้ายมือ (จากลักษณะการวางถัง เบื้องต้น และการติดตั้ง screw conveyor)

ข) ถ้าใช้ถัง 23 และถัง 25 เลือกลงถัง 23 ก่อน

ค) ใช้ถัง 25 และถัง 21 เลือกลงถัง 25 ก่อน

11. จากการศึกษาถึงความสามารถในการบรรจุของหัวบรรจุทั้ง 3 หัว พยายามหลีกเลี่ยงการใช้หัวบรรจุหัว 2 ลงอาหารถุงขนาด 50 kg. ดังนั้นจึงส่งผลให้ อาหารที่ต้องบรรจุถุงขนาด 50 kg. หลีกเลี่ยงการใช้ถัง 18 และ 22

ภาคผนวก ฉ

เวลาที่ใช้ในการผลิตอาหารสัตว์ที่แต่ละเครื่องจักร

กลุ่มอาหาร	Mixer	Pellet #1	Pellet #2	Packing #2 (30 kg)	Packing #2 (50 kg)	Packing #3 (30 kg)	Packing #3 (50 kg)	Packing #4 (30 kg)	Packing #4 (50 kg)
1	4:33.18	00:00.00	00:00.00	11:29.28	21:25.74	08:16.08	08:07.14	09:15.78	06:21.48
2	4:33.18	00:00.00	00:00.00	11:29.28	21:25.74	08:16.08	08:07.14	09:15.78	06:21.48
3	4:33.18	06:48.06	06:41.04	11:29.28	21:25.74	08:16.08	08:07.14	09:15.78	06:21.48
4	4:33.18	07:47.34	06:41.04	11:29.28	21:25.74	08:16.08	08:07.14	09:15.78	06:21.48
5	4:52.44	08:59.52	08:59.52	14:21.54	26:47.16	10:20.01	10:08.94	11:34.74	07:56.82
6	5:01.08	09:31.02	08:38.76	17:13.86	26:47.16	12:24.12	12:10.74	13:53.64	09:32.22
7	5:51.42	10:24.66	09:54.96	17:13.86	26:47.16	12:24.12	12:10.74	13:53.64	09:32.22
8	5:51.42	10:00.00	11:31.74	17:13.86	26:47.16	12:24.12	12:10.74	13:53.64	09:32.22

หน่วย : นาที

ตารางแสดง Processing time แยกตามกลุ่มอาหาร เครื่องจักร และขนาดบรรจุ

ภาคผนวก ช

ผลที่ได้จากการทดสอบทางสถิติด้วยโปรแกรม SAS

SAS 09:42 Friday, January 10, 1986 1

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	26	151 152 153 155 324 324ND 324NDZ06 324NDZ09 325ND 325NDZ06 325NDZ09 325NDZ16 326ND 326NDZ09 335ND 521 521N 522 522N 523 524 525 566 567 567NS 567X17
GROUP	4	C1 M1 P2 P6
TYPE	3	C M P

Number of observations in data set = 271

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	30	1458.0817750	48.6027258	2.30	0.0003
Error	240	5072.1062811	21.1337762		
Corrected Total	270	6530.1880561			

R-Square	C.V.	Root MSE	TIME Mean
0.223283	17.442259	4.5971487	26.35638376

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	25	405.69514	56.22781	2.66	0.0001
GROUP	3	29.56435	9.85478	0.47	0.7061
TYPE	2	2.82228	11.41114	0.54	0.5835

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate, not the experimentwise error rate

Alpha = 0.05 df = 240 MSE = 21.13378

WARNING : Cell sizes are not equal.

Harmonic Mean of cell sizes = 51.39788

Number of Means	2	3	4
Critical Range	1.8014758	1.8943623	1.9541319

Means with the same letter are not significantly different

Duncan Grouping	Mean	N	GROUP
A	26.702	12	M1
A			
A	26.406	47	P6
A			
A	26.163	28	C1
A			
A	5.933	84	P2

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate, not the experimentwise error rate

Alpha= 0.05 df= 240 MSE= 21.13378
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 57.38722

Number of Means	2	3
Critical Range	1.7048786	1.7927844

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	TYPE
A	26.702	112	M
A			
A	26.163	28	C
A			
A	26.103	31	P

SAS 09:42 Friday, January 10, 1986 14

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	6	552 552NS 552X09 553 554 555
GROUP	1	P5

Number of observations in data set = 158

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	103.84547114	20.76909423	1.88	0.1010
Error	152	1679.50965861	11.04940565		
Corrected Total	157	1783.35512975			

R-Square	C.V.	Root MSE	TIME Mean
0.058230	10.801985	3.3240646	0.77272152

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	5	103.84547	20.76909	1.88	0.1010
GROUP	0	0.00000	.	.	.

Duncan's Multiple Range Test for variable: TIME

NOTE : This test controls the type I comparisonwise error rate, not the experimentwise error rate

Alpha= 0.05 df= 152 MSE= 11.04941

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 21.42824

Number of Means	2	3	4	5	6
Critical Range	2.0173847	2.1214037	2.1883368	2.2386201	2.2809364

Means with the same letter are not significantly different

Duncan	Grouping	Mean	N	CODE
	A	31.966	33	553
	A			
B	A	31.175	27	554
B	A			
B	A	30.769	9	552X09
B	A			
B	A	30.561	1	555
B	A			
B	A	30.544	31	552
B				
B		29.419	27	552NS

SAS 09:42 Friday, January 10, 1986 19

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	13	510 511 511EX 513 531N 532N 534ND 534NDZ06 534NDZ09 534NDZ16 551 551NS 551X05
GROUP	3	P1 P3 P4
TYPE	1	P

Number of observations in data set = 192

Dependent Variable: TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	14	3081.9592416	220.1399458	23.33	0.0001
Error	177	1669.8809500	9.4343556		
Corrected Total	191	4751.8401917			

R-Square	C.V.	Root MSE	TIME Mean
0.648582	9.1928433	3.0715396	33.41229167

Dependent Variable: TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
CODE	12	1815.20799	151.26733	16.03	0.0001
GROUP	2	1266.75125	633.37563	67.14	0.0001
TYPE	0	0.00000	.	.	.

`Duncan's Multiple Range Test for variable: TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 177 MSE= 9.434356

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 51.79856

Number of Means	2	3
Critical Range	1.198974	1.2607947

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	GROUP
A	5.876	100	P1
B	30.777	60	P3
B			
B	30.656	32	P4

SAS 09:42 Friday, January 10, 1986 27

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	11	521 521N 522 522N 523 524 525 566 567 567NS 567X17
GROUP	4	P21 P22 P61 P62
MACHINE	2	1 2

Number of observations in data set = 139

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	14	395.20290844	28.22877917	4.42	0.0001
Error	124	792.25215631	6.38913029		
Corrected Total	138	1187.45506475			

R-Square	C.V.	Root MSE	TIME Mean
0.332815	14.670614	2.5276729	17.22949640

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	10	155.149436	15.514944	2.43	0.0112
GROUP	3	163.267657	54.422552	8.52	0.0001
MACHINE	1	76.785815	76.785815	12.02	0.0007

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 124 MSE= 6.38913

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 24.91338

Number of Means	2	3	4

Critical Range	1.4227121	1.496069	1.5432719

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	GROUP
A	18.062	61	P22
A			
A	17.621	31	P21
A			
A	17.302	12	P62
B	15.407	35	P61

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 124 MSE= 6.38913

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 69.20863

Number of Means	2
Critical Range	0.8535977

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	MACHINE
A	17.926	74	2
B	16.436	65	1

SAS 09:42 Friday, January 10, 1986 37

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	6	552 552NS 552X09 553 554 555
GROUP	2	P51 P52
MACHINE	2	1 2

Number of observations in data set = 228

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	7	187.90034311	26.84290616	0.41	0.8973
Error	220	14501.8494236	65.91749738		
CorrectedTotal	227	14689.7497667			

R-Square	C.V.	Root MSE	TIME Mean
0.012791	48.669952	8.1189591	16.68166667

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	5	102.683001	20.536600	0.31	0.9057
GROUP	1	42.608671	42.608671	0.65	0.4223
MACHINE	1	42.608671	42.608671	0.65	0.4223

Duncan's Multiple Range Test for variable: TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 220 MSE= 65.9175

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 30.26868

Number of Means	2	3	4	5	6
Critical Range	4.1458744	4.3596413	4.4971937	4.6005295	4.6874926

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	CODE
A	17.594	45	555
A			
A	17.310	43	552
A			
A	16.452	35	552NS
A			
A	16.400	59	553
A			
A	15.812	33	554
A			
A	15.548	13	552X09

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 220 MSE= 65.9175
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 102.6316

Number of Means	2
Critical Range	2.2515021

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	MACHINE
A	17.281	78	2
A			
A	16.370	150	1

SAS 09:42 Friday, January 10, 1986 44

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	13	510 511 511EX 513 531N 532N 534ND 534NDZ06 534NDZ09 534NDZ16 551 551NS 551X05
GROUP	5	P11 P12 P31 P32 P42
MACHINE	2	1 2

Number of observations in data set = 240

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	17	1727.3187868	101.6069875	62.71	0.0001
Error	222	359.6911428	1.6202304		
Corrected Total	239	2087.0099296			

R-Square	C.V.	Root MSE	TIME Mean
0.827652	6.7979681	1.2728827	18.72445833

Dependent Variable: TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	12	906.559025	75.546585	46.63	0.0001
GROUP	4	785.281408	196.320352	121.17	0.0001
MACHINE	1	35.478354	35.478354	21.90	0.0001

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 222 MSE= 1.62023

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 41.37821

Number of Means	2	3	4	5
Critical Range	.55592351	.58458767	.60303218	.61688856

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	GROUP
A	20.819	81	P12
B	18.908	58	P11
C	18.153	27	P32
D	17.290	39	P31
E	15.613	35	P42

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha = 0.05 df= 222 MSE= 1.62023
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 115.5917

Number of Means	2
Critical Range	33261178

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	MACHINE
A	19.041	143	2
B	18.258	97	1

SAS 09:42 Friday, January 10, 1986 54

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	11	510 511 511EX 524 525 551 552 553 555 566 567
BATCH	3	20t 25t 30t

Number of observations in data set = 215

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	12	65.59112281	5.46592690	1.59	0.0970
Error	202	694.95390975	3.44036589		
Corrected Total	214	760.54503256			

R-Square	C.V.	Root MSE	TIME Mean
0.086242	17.756609	1.8548223	10.44581395

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	10	52.645259	5.264526	1.53	0.1306
BATCH	2	12.945863	6.472932	1.88	0.1550

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 202 MSE= 3.440366

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 67.22401

Number of Means	2	3
Critical Range	.63555423	.66832426

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	BATCH
A	0.857	51	20t
A			
A	10.404	95	30t
A			
A	10.200	69	25t

SAS 09:42 Friday, January 10, 1986 62

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	17	510 511 511EX 513 521 522 523 524 525 551 552 553 554 555 566 567 567X17
BATCH	3	20t 25t 30t

Number of observations in data set = 182

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	18	1214.3836446	67.4657580	0.82	0.6770
Error	163	13440.3868394	82.4563610		
Corrected Total	181	14654.7704841			

R-Square	C.V.	Root MSE	TIME Mean
0.082866	62.564966	9.0805485	14.51379121

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	16	963.596425	60.224777	0.73	0.7601
BATCH	2	250.787220	125.393610	1.52	0.2216

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 163 MSE= 82.45636

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 58.05482

Number of Means 2 3

Critical Range	3.3481541	3.5207895

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	BATCH
A	16.204	59	30t
A			
A	13.837	46	20t
A			
A	13.623	77	25t

SAS 09:42 Friday, January 10, 1986 71

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	14	510 511 511EX 513 522 524 525 551 552 553 554 555 566 567
BATCH	3	20t 25t 30t

Number of observations in data set = 128

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	15	51.91858502	3.46123900	0.97	0.4917
Error	112	399.81279545	3.56975710		
Corrected Total	127	451.73138047			

R-Square	C.V.	Root MSE	TIME Mean
0.114932	14.584616	1.8893801	12.95460938

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	13	50.097933	3.853687	1.08	0.3839
BATCH	2	1.820652	0.910326	0.26	0.7754

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 112 MSE= 3.569757

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 36.20429

Number of Means	2	3
Critical Range	.88216905	.92765486

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	BATCH
A	13.186	23	20t
A			
A	12.971	43	30t
A			
A	12.858	62	25t

SAS 09:42 Friday, January 10, 1986 103

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	6	151 152 153 155 324 324ND

Number of observations in data set = 16

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	64.19981875	12.83996375	2.07	0.1535
Error	10	62.06977500	6.20697750		
Corrected Total	15	126.26959375			

R-Square	C.V.	Root MSE	TIME Mean
0.508435	20.378350	2.4913806	12.22562500

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	5	64.1998188	12.8399638	2.07	0.1535

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 10 MSE= 6.206977

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 2.4

Number of Means	2	3	4	5	6
Critical Range	5.0583575	5.2899626	5.4384989	5.5139206	5.5668631

Means with the same letter are not significantly different.

Duncan	Grouping	Mean	N	CODE
	A	14.203	4	324
	A			
	A	13.825	2	324ND
	A			
B	A	12.320	2	153
B	A			
B	A	11.980	4	151
B	A			
B	A	11.685	2	155
B				
B		7.610	2	152

SAS 09:42 Friday, January 10, 1986 80

Analysis of Variance Procedure

Class Level Information

Class	Levels	Values
CODE	8	324ND 325ND 335ND 521N 531N 552NS 566 567NS
BATCH	3	20t 25t 30t

Number of observations in data set = 35

Dependent Variable : TIME

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	97.52255565	10.83583952	1.61	0.1650
Error	25	167.76983292	6.71079332		
Corrected Total	34	265.29238857			

R-Square	C.V.	Root MSE	TIME Mean
0.367604	13.725751	2.5905199	18.87342857

Dependent Variable : TIME

Source	DF	Anova SS	Mean Square	F Value	Pr > F
CODE	7	93.927175	13.418168	2.00	0.0956
BATCH	2	3.595380	1.797690	0.27	0.7672

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 25 MSE= 6.710793

WARNING: Cell sizes are not equal.

Harmonic Mean of cell sizes= 1.637343

Number of Means	2	3	4	5
Critical Range	5.8912467	6.1899513	6.3955544	6.521753

Number of Means	6	7	8
Critical Range	6.623534	6.7038876	6.7675929

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	CODE
A	20.114	19	324ND
A			
A	18.750	1	325ND
A			
A	18.600	1	566
A			
A	18.450	1	531N
A			
A	18.202	6	552NS
A			
A	18.150	1	521N
A			
A	16.967	3	567NS
A			
A	14.780	3	335ND

Duncan's Multiple Range Test for variable : TIME

NOTE : This test controls the type I comparisonwise error rate,not the experimentwise error rate

Alpha= 0.05 df= 25 MSE= 6.710793
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 2.49505

Number of Means	2	3
Critical Range	4.7724067	5.0143827

Means with the same letter are not significantly different.

Duncan Grouping	Mean	N	BATCH
A	19.033	28	20t
A			
A	18.450	1	30t
A			
A	18.202	6	25t

ภาคผนวก ฆ

ชุดคำสั่งภาษาต้นฉบับ

```
D=================================================================
;FILE FEED.MDL (10/04/1991)
;=================================================================
$
C=(0)
M=(180)
W=(150)
S=(1000)
N=(4000)
O=(FEED.OLY)                    ;get overlay file name FEED.OLY
#SEED=(1357)

;=========================== Counter ============================
@A=(0)          @COUNTPK2=(0)  @KK2=(10)       @PK2=(60)
@Z=(0)          @COUNTPK3=(0)  @KK3=(13)       @PK3=(60)
@V=(0)          @COUNTPK4=(0)  @KK4=(7)        @PK4=(60)
@L=(0)          @COUNTER=(1)   @KKS=(14)       @PKS=(60)
@K=(0)          @COUNTM=(0)    @POSS=(1)       @STEP0=(0)
@B=(0)          @COUNTP=(0)    @POSS1=(0)      @STEP1=(0)
@N=(0)          @COUNTS=(0)    @POSS2=(0)      @MASHNUM=(0)
@P=(0)          @PKK2=(0)      @POSS3=(7)      @PELLNUM=(0)
@W=(0)          @PKK3=(0)      @MF=(0)         @MASHF=(0)
@H=(1)          @PKK4=(0)      @PF=(0)         @PELLF=(0)
@PM1=(1)        @PKKS=(0)      @PET=(0)        @SILOO=(0)
@PM2=(2)        @CONC=(0)      @TPACK=(0)      @DNCODE=(0)
@OBJ2=(0)       @DUMM=(0)      @TONE=(100)     @NOBATCH=(0)
@OBJ3=(0)       @USEPEL=(0)    @DURATION=(1)   @PEET=(0)
@PRBAG=(0)      @PR=(0)        @TMIX=(0)       @OK=(0)
@CS=(1)         @CPK4=(1)      @CPK2=(1)       @CPK3=(1)
@P1=(0)         @P2=(0)        @CPROC=(0)      @CPR=(1)
@MIX1=(0)       @MIXC1=(0)     @PELTEST=(0)    @PMM=(0)
@PMM1=(0)       @MM1=(0)       @CPM1=(1)       @COUNTPM1=(0)
@PMM2=(0)       @MM2=(0)       @CPM2=(1)       @COUNTPM2=(0)
@PM1B=(0)       @PM1C=(0)      @P1MM=(0)       @PM1BIN=(0)
@PM2B=(0)       @PM2C=(0)      @P2MM=(0)       @PM2BIN=(0)
@PLC1=(0)       @PLB1=(0)      @PLC21=(0)      @PLB21=(0)
@PK2N1=(0)      @PK3N1=(0)     @PK4N1=(0)      @PKSN1=(0)
@PK2B1=(0)      @PK3B1=(0)     @PK4B1=(0)      @PKSB1=(0)
@PK2BIN=(0)     @PK3BIN=(0)    @PK4BIN=(0)     @PKSBIN=(0)
@SILO=(0)       @PCODE=(0)     @COUNTS1=(0)    @COUNTPK41=(0)
@PSILO=(0)      @COUNTPK21=(0) @COUNTPK31=(0)

;======================Variable of time==========================
%INTERA=(0:0:10)  %PM1=(0:0:0)       %PM2=(0:0:0)
%CONV=(0:0:0)     %START=(8:0:0)     ;Start at 8:00 am.
%BUFF=(0:0:1)     %FINISH=(32:0:0)   %SHIFT1=(17:0:0)
%DAY=(24:0:0)     %LIMIT=(0:0:0)     %SHIFT2=(24:0:0)
%PRT=(0:0:0)      %PK3=(0:0:0)       %CLOCK=(0:0:0)
%PK4=(0:0:0)      %PK2=(0:0:0)       %PKS=(0:0:0)
%STOP=(0:0:0)     %END1=(0:0:0)      %SET=(0:0:0)
%DAY1=(32:0:0)    %TIME=(0:0:0)      %STO=(0:0:0)
%EMIX=(0:0:0)     %EPM1=(0:0:0)      %EPM2=(0:0:0)
%EPK2=(0:0:0)     %EPK3=(0:0:0)      %EPK4=(0:0:0)
%END=(0:0:0)      %PROC=(0:0:0)
```

```
;========== Display input daily plan & current batch left =======
*U0=(XY(20,6))   *U1=(XY(29,6))    *U2=(XY(38,6))
*U3=(XY(47,6))   *U4=(XY(56,6))    *U5=(XY(65,6))   *U6=(XY(74,6))
*U7=(XY(20,9))   *U8=(XY(29,9))    *U9=(XY(38,9))
*U10=(XY(47,9))  *U11=(XY(56,9))   *U12=(XY(65,9))  *U13=(XY(74,9))
*U14=(XY(20,3))  *U15=(XY(29,3))   *U16=(XY(38,3))
*U17=(XY(47,3))  *U18=(XY(56,3))   *U19=(XY(65,3))  *U20=(XY(74,3))

;============ Display fail symbol at machine ====================
*DSP0=(XY(35,35))            ;mixer
*DSP2=(XY(22,75))            ;PK2
*DSP3=(XY(35,75))            ;PK3
*DSP4=(XY(7,75))             ;PK4
*DSPS=(XY(57,75))            ;SILO
*DS1=(XY(29,50))             ;PM1
*DS2=(XY(50,50))             ;PM2

;=============== Display daily plan onto screen ================
*CONC=(XY(66,12))   *PERCONC=(XY(42,14))  *FINISH=(XY(47,15))
*PET=(XY(66,5))     *PERPET=(XY(42,15))   *FINISH1=(XY(2,14))
*S0=(XY(20,1))      *PERSI=(XY(42,16))    *FINISH2=(XY(2,13))
*T0=(XY(20,2))      *NUMPAC=(XY(55,18))   *PPM2=(XY(56,54))
*S1=(XY(20,4))      *NUMPEL=(XY(55,17))   *PPM1=(XY(26,54))
*T1=(XY(20,5))      *PT=(XY(55,23))       *PROD=(XY(11,35))
*SS1=(XY(20,7))     *PTM=(XY(55,22))      *PPM=(XY(37,37))
*TT1=(XY(20,8))     *PTP=(XY(55,21))      *DISPLAY=(XY(47,19))
*PTS=(XY(55,24))

;======= Display status wait for Mixer, Pellet #1 & #2 ==========
*ALERT0=(XY(34,52))  *ALERT1=(XY(64,52))  *COLUMN=(XY(9,0))
*ALERT=(XY(45,35))

;=================== Machine Location ===========================
*INIT=(XY(1,31))    *MIXER=(XY(35,35))
*PELL1=(XY(24,51))  *PELL2=(XY(54,51))
*PACK2=(XY(22,72))  *PACK3=(XY(35,72))
*PACK4=(XY(07,72))  *SILO=(XY(57,72))

;=========================Bin Array=================================
@@BIN=(16,3)       @@QUAN=(20,4)      @@PACK=(3,0)
@@PRT=(50,0)       @@CODE=(50,0)      @@CODED=(22,0)
@@PK4=(15,10)      @@PKK4=(15,10)     @@PK2=(15,10)
@@PKK2=(15,10)     @@PK3=(15,10)      @@PKK3=(15,10)
@@PKS=(10,10)      @@PKKS=(10,10)     @@PM=(20,1)
@@PMCODE=(20,1)    @@PKCODE=(15,3)    @@CELL=(7,1)
@@BRK=(7,2)        @@DNCODE=(20,0)    @@MIX=(25,8)
@@MIXC=(25,8)      @@PM1B=(15,10)     @@PM1C=(15,10)
@@PM2B=(15,10)     @@PM2C=(15,10)
%%BREAK=(6,1)      %%PRT=(40,0)       %%PM=(20,1)
%%PROC=(8,9)       %%STOP=(20,0)      %%END=(20,0)
%%PK=(15,3)

;================================================================
J=(1,I,1,0,0,0,1)             ;Job for Initialization
J=(2,∞,2,0,1,0,20000)        ; Route PART
```

```
J=(6,4,6,0,0,0,1)              ; Breakdown for PK4
J=(7,4,7,0,0,0,1)              ; Breakdown for PK2
J=(8,4,8,0,0,0,1)              ; Breakdown for PK3

U=(1,MIXER,*MIXER)             ; Machine utilization collection
U=(2,PELLET1,*PELL1)
U=(3,PELLET2,*PELL2)
U=(4,PACK2,*PACK2)
U=(5,PACK3,*PACK3)
U=(6,PACK4,*PACK4)
U=(7,SILO,*SILO)
;================================================================
;          routine for moving in bins :
;          STORE1 => 10 tons
;          STORE2 => 15 tons
;          STORE3 => 22.5-26 tons
;          STORE4 => 36 tons
;================================================================
BL(!STORE1)                    ;for b1, b25, b26, b27
  RM(+0,+4,0)
  MR(1,%BUFF)   MD(1,%BUFF)
  ML(1,%BUFF)   MD(1,%BUFF)
EL
BL(!STORE2)                    ;for b2, b3, b4
  RM(-1,+4,0)
  MR(2,%BUFF)   MD(1,%BUFF)
  ML(2,%BUFF)   MD(1,%BUFF)
EL
BL(!STORE3)                    ;for b15, b16, b17, b21, b18, b22, b19, b23
  RM(+1,+3,0)
  ML(3,%BUFF)   MD(1,%BUFF)   MR(3,%BUFF)
  MD(1,%BUFF)   ML(3,%BUFF)   MD(1,%BUFF)
EL
BL(!STORE4)                    ;for b20, b24
  RM(+3,+3,0)
  ML(5,%BUFF)   MD(1,%BUFF)   MR(5,%BUFF)
  MD(1,%BUFF)   ML(5,%BUFF)   MD(1,%BUFF)
EL
;================================================================
;     Set the code  incoming product to bin
;     Different code cannot be put into the same bin at once
;================================================================
BL(!SET)
  IF(@@BIN(OBJ@5,0),NE,0,:ENDSET)
  SV(@@BIN(OBJ@5,2),OBJ@4)
:ENDSET
EL
;================================================================
;     Processing time at Packing Head #2 #3 and #4
;     is tested to be Significant different
;================================================================
BL(!PROCPK2)
  IF(OBJ@2,EQ,30,:PACK302)
    PV(XY(15,76),%%PROC(OBJ@6,5))  ;Processing time for 50 kg. pack
    WT(%%PROC(OBJ@6,5))
```

```
    JP(:ENDPROC2)
  :PACK302
    PV(XY(15,76),%%PROC(OBJ@6,4))  ;Processing time for 30 kg. pack
    WT(%%PROC(OBJ@6,4))
:ENDPROC2
  PV(XY(20,78),OBJ@2)
  SV(%EPK2,CLOCK)
EL

BL(!PROCPK3)
  IF(OBJ@2,EQ,30,:PACK303)
    PV(XY(29,76),%%PROC(OBJ@6,7))  ;Processing time for 50 kg. pack
    WT(%%PROC(OBJ@6,7))
    JP(:ENDPROC3)
  :PACK303
    PV(XY(29,76),%%PROC(OBJ@6,6))  ;Processing time for 30 kg. pack
    WT(%%PROC(OBJ@6,6))
:ENDPROC3
  PV(XY(33,78),OBJ@2)
  SV(%EPK3,CLOCK)
EL

BL(!PROCPK4)
  IF(OBJ@2,EQ,30,:PACK304)
    PV(XY(1,76),%%PROC(OBJ@6,9))  ;Processing time for 50 kg. pack
    WT(%%PROC(OBJ@6,9))
    JP(:ENDPROC4)
  :PACK304
    PV(XY(1,76),%%PROC(OBJ@6,8))  ;Processing time for 30 kg. pack
    WT(%%PROC(OBJ@6,8))
:ENDPROC4
  PV(XY(5,78),OBJ@2)
  SV(%EPK4,CLOCK)
EL
;=================================================================
;     Sud-routine for bin switching
;       - Pellet Mill #1 => TTP0,TTP1
;       - Pellet Mill #2 => TTP2,TTP3,TTP4,TTP5
;=================================================================
BL(!TTP0)                       ;for bin #16
  IF(@@BIN(1,0),EQ,0,:TOB15)    ;if bin #16 empty then switch to bin #15
    SV(@PM1,1)                  ;continue
    JP(:ETP0)
  :TOB15
  IF(@@BIN(0,0),EQ,0,:TOB16A)   ;if both empty then set to bin #16
    SV(@PM1,0)
    JP(:ETP0)
  :TOB16A
    SV(@PM1,1)
:ETP0
EL

BL(!TTP1)                       ;for bin #15
  IF(@@BIN(0,0),EQ,0,:TOB16)    ;if bin #15 empty then switch to bin #16
    SV(@PM1,0)                  ;continue
```

```
    JP(:ETP1)
  :TOB16
    SV(@PM1,1)
:ETP1
EL
;-------------------------------------------------------------------
BL(!TTP2)                          ;for bin #1
  IF(@@BIN(2,0),EQ,0,:TOB04)       ;if bin #1 empty then switch to bin #4
    SV(@PM2,2)                     ;continue
    JP(:ETP2)
  :TOB04
  IF(@@BIN(5,0),EQ,0,:TOB03)       ;if bin(1&4) empty then switch to bin #3
    SV(@PM2,5)
    JP(:ETP2)
  :TOB03
  IF(@@BIN(4,0),EQ,0,:TOB02)       ;if bin(1,4,3) empty then switch to bin 2
    SV(@PM2,4)
    JP(:ETP2)
  :TOB02
  IF(@@BIN(3,0),EQ,0,:TOB01A)      ;if all are empty then set to bin #1
    SV(@PM2,3)
    JP(:ETP2)
  :TOB01A
    SV(@PM2,2)
:ETP2
EL

BL(!TTP3)                          ;for bin #4
  IF(@@BIN(5,0),EQ,0,:TOB03A)      ;if bin #4 empty then switch to bin #3
    SV(@PM2,5)
    JP(:ETP3)
  :TOB03A
  IF(@@BIN(4,0),EQ,0,:TOB02A)      ;if bin(4&3) empty then switch to bin 2
    SV(@PM2,4)
    JP(:ETP3)
  :TOB02A
  IF(@@BIN(3,0),EQ,0,:TOB01B)      ;if all are empty then set to bin #1
    SV(@PM2,3)
    JP(:ETP3)
  :TOB01B
    SV(@PM2,2)
:ETP3
EL

BL(!TTP4)                          ;for bin #3
  IF(@@BIN(4,0),EQ,0,:TOB02B)      ;if bin #3 empty then switch to bin 2
    SV(@PM2,4)
    JP(:ETP4)
  :TOB02B
  IF(@@BIN(3,0),EQ,0,:TOB01C)      ;if all are empty then set to bin #1
    SV(@PM2,3)
    JP(:ETP4)
  :TOB01C
    SV(@PM2,2)
:ETP4
```

```
EL

BL(!TTP5)                          ;for bin #2
  IF(@@BIN(3,0),EQ,0,:TOB01D) ;if all are empty then set to bin #1
    SV(@PM2,3)                     ;continue
    JP(:ETP5)
  :TOB01D
    SV(@PM2,2)
:ETP5
EL
;=================================================================
;                  Packing head #4 Routing
;=================================================================
BL(!TTP6)                          ;for bin #21
  IF(@@BIN(7,0),EQ,0,:TOB17)   ;if bin 21 empty then switch to bin 17
    SV(@KK4,7)                     ;continue
    JP(:ETP6)
  :TOB17
  IF(@@BIN(6,0),EQ,0,:TOB25)   ;if bin 17 empty then switch to bin 25
    SV(@KK4,6)                     ;continue
    JP(:ETP6)
  :TOB25
  IF(@@BIN(8,0),EQ,0,:TOB21A) ;if all are empty then set to bin #21
    SV(@KK4,8)
    JP(:ETP6)
  :TOB21A
    SV(@KK4,7)
:ETP6
EL

BL(!TTP7)                          ;for bin #17
  IF(@@BIN(6,0),EQ,0,:TOB25A) ;if bin 17 empty then switch to bin 25
    SV(@KK4,6)                     ;continue
    JP(:ETP7)
  :TOB25A
  IF(@@BIN(8,0),EQ,0,:TOB21B) ;if all are empty then set to bin #21
    SV(@KK4,8)
    JP(:ETP7)
  :TOB21B
    SV(@KK4,7)
:ETP7
EL

BL(!TTP8)                          ;for bin #25
  IF(@@BIN(8,0),EQ,0,:TOB21C) ;if all are empty then set to bin #21
    SV(@KK4,8)
    JP(:ETP8)
  :TOB21C
    SV(@KK4,7)
:ETP8
EL
;=================================================================
;                  Packing head #2 Routing
;=================================================================
BL(!TTP9)                          ;for bin #22
```

```
  IF(@@BIN(10,0),EQ,0,:TOB18C) ;if bin 22 empty then switch to bin 18
    SV(@KK2,10)                ;continue
    JP(:ETP9)
  :TOB18C
  IF(@@BIN(9,0),EQ,0,:TOB22B) ;if both are empty then set to bin 22
    SV(@KK2,9)                 ;continue
    JP(:ETP9)
  :TOB22B
    SV(@KK2,10)
:ETP9
EL

BL(!TTP10)                     ;for bin #18
  IF(@@BIN(9,0),EQ,0,:TOB22C) ;if bin 18 empty then switch to bin 22
    SV(@KK2,9)                 ;continue
    JP(:ETP10)
  :TOB22C
    SV(@KK2,10)
:ETP10
EL
;=================================================================
;                Packing head #3 Routing
;=================================================================
BL(!TTP11)                     ;for bin #23
  IF(@@BIN(13,0),EQ,0,:TOB19) ;if bin #23 empty then switch to bin 19
    SV(@KK3,13)                ;continue
    JP(:ETP11)
  :TOB19
  IF(@@BIN(12,0),EQ,0,:TOB26) ;if bin 19 empty then switch to bin 26
    SV(@KK3,12)                ;continue
    JP(:ETP11)
  :TOB26
  IF(@@BIN(11,0),EQ,0,:TOB23A) ;if all are empty then set to bin #23
    SV(@KK3,11)
    JP(:ETP11)
  :TOB23A
    SV(@KK3,13)
:ETP11
EL

BL(!TTP12)                     ;for bin #19
  IF(@@BIN(12,0),EQ,0,:TOB26A) ;if bin 19 empty then switch to bin 26
    SV(@KK3,12)                ;continue
    JP(:ETP12)
  :TOB26A
  IF(@@BIN(11,0),EQ,0,:TOB23B) ;if all are empty then set to bin #23
    SV(@KK3,11)
    JP(:ETP12)
  :TOB23B
    SV(@KK3,13)
:ETP12
EL

BL(!TTP13)                     ;for bin #26
  IF(@@BIN(11,0),EQ,0,:TOB23C) ;if all are empty then set to bin #18
```

```
    SV(@KK3,11)
    JP(:ETP13)
  :TOB23C
    SV(@KK3,13)
:ETP13
EL
;================================================================
;                          Silo Routing
;================================================================
BL(!TTP14)                          ;for bin #27
  IF(@@BIN(14,0),EQ,0,:TOB20)       ;if bin #27 empty then switch to bin 20
    SV(@KKS,14)                     ;continue
    JP(:ETP14)
  :TOB20
  IF(@@BIN(15,0),EQ,0,:TOB24)       ;if bin 20 empty then switch to bin 24
    SV(@KKS,15)                     ;continue
    JP(:ETP14)
  :TOB24
  IF(@@BIN(16,0),EQ,0,:TOB27)       ;if all are empty then set to bin #27
    SV(@KKS,16)
    JP(:ETP14)
  :TOB27
    SV(@KKS,14)
:ETP14
EL

BL(!TTP15)                          ;for bin #20
  IF(@@BIN(15,0),EQ,0,:TOB24A)      ;if bin 20 empty then switch to bin 24
    SV(@KKS,15)                     ;continue
    JP(:ETP15)
  :TOB24A
  IF(@@BIN(16,0),EQ,0,:TOB27A)      ;if all are empty then set to bin #27
    SV(@KKS,16)
    JP(:ETP15)
  :TOB27A
    SV(@KKS,14)
:ETP15
EL

BL(!TTP16)                          ;for bin #24
  IF(@@BIN(16,0),EQ,0,:TOB27B)      ;if all are empty then set to bin #27
    SV(@KKS,16)
    JP(:ETP16)
  :TOB27B
    SV(@KKS,14)
:ETP16
EL
;================================================================
;    display remaining batch & set color for each batch mixed
;================================================================
BL(!SETCOLOR)
  IF(@A,GE,@PELLNUM,:SET2CO,ELSE,NEXT)
  IF(@A,EQ,1,:S1)          IF(@A,EQ,2,:S2)
  IF(@A,EQ,3,:S3)          IF(@A,EQ,4,:S4)
  IF(@A,EQ,5,:S5)          IF(@A,EQ,6,:S6)
```

```
  IF(@A,EQ,7,:S7)          IF(@A,EQ,8,:S8)
  IF(@A,EQ,9,:S9)          IF(@A,EQ,10,:S10)
  IF(@A,EQ,11,:S11)        IF(@A,EQ,12,:S12)
  IF(@A,EQ,13,:S13)        IF(@A,EQ,14,:S14)
:SET2CO
  SV(@PELTEST,15)
  AO(@PELTEST,-,@PELLNUM)
  AO(@PELTEST,+,@A)
  IF(@PELTEST,EQ,15,:S15)        IF(@PELTEST,EQ,16,:S16)
  IF(@PELTEST,EQ,17,:S17)        IF(@PELTEST,EQ,18,:S18)
  IF(@PELTEST,EQ,19,:S19)        IF(@PELTEST,EQ,20,:S20)
  IF(@PELTEST,EQ,21,:S21)
:S1
  PV(*U0,@B)
  SC(0,9)
  JP(:SEND)
:S2
  PV(*U1,@B)
  SC(0,10)
  JP(:SEND)
:S3
  PV(*U2,@B)
  SC(0,11)
  JP(:SEND)
:S4
  PV(*U3,@B)
  SC(0,12)
  JP(:SEND)
:S5
  PV(*U4,@B)
  SC(0,13)
  JP(:SEND)
:S6
  PV(*U5,@B)
  SC(0,14)
  JP(:SEND)
:S7
  PV(*U6,@B)
  SC(0,15)
  JP(:SEND)
:S8
  PV(*U7,@B)
  SC(0,2)
  JP(:SEND)
:S9
  PV(*U8,@B)
  SC(0,11)
  JP(:SEND)
:S10
  PV(*U9,@B)
  SC(0,12)
  JP(:SEND)
:S11
  PV(*U10,@B)
  SC(0,13)
```

```
  JP(:SEND)
:S12
  PV(*U11,@B)
  SC(0,9)
  JP(:SEND)
:S13
  PV(*U12,@B)
  SC(0,10)
  JP(:SEND)
:S14
  PV(*U13,@B)
  SC(0,11)
  JP(:SEND)
:S15
  PV(*U14,@B)
  SV(OBJ@ID,"1)
  SC(0,12)
  JP(:SEND)
:S16
  PV(*U15,@B)
  SV(OBJ@ID,"2)
  SC(0,13)
  JP(:SEND)
:S17
  PV(*U16,@B)
  SV(OBJ@ID,"3)
  SC(0,14)
  JP(:SEND)
:S18
  PV(*U17,@B)
  SV(OBJ@ID,"4)
  SC(0,15)
  JP(:SEND)
:S19
  PV(*U18,@B)
  SV(OBJ@ID,"5)
  SC(0,13)
  JP(:SEND)
:S20
  PV(*U19,@B)
  SV(OBJ@ID,"6)
  SC(0,14)
  JP(:SEND)
:S21
  PV(*U20,@B)
  SV(OBJ@ID,"7)
  SC(0,15)
:SEND
  PV(XY(55,20),@@QUAN(@A,0))  ;Print product code
  PV(XY(56,35),@@QUAN(@A,0))  ;Print product code at mixer
  PV(XY(70,35),@B)            ;Print remain batch at mixer
EL
;================================================================
;                    Capacity of each bins
;================================================================
```

```
BL(!BINCAP)
  SV(@@BIN(0,3),225)      SV(@@BIN(1,3),225)
  SV(@@BIN(2,3),100)      SV(@@BIN(3,3),140)
  SV(@@BIN(4,3),140)      SV(@@BIN(5,3),140)
  SV(@@BIN(6,3),260)      SV(@@BIN(7,3),260)
  SV(@@BIN(8,3),100)      SV(@@BIN(9,3),260)
  SV(@@BIN(10,3),260)     SV(@@BIN(11,3),100)
  SV(@@BIN(12,3),260)     SV(@@BIN(13,3),260)
  SV(@@BIN(14,3),100)     SV(@@BIN(15,3),360)
  SV(@@BIN(16,3),360)
EL
;=================================================================
BL(!BIN2)
  IF(@PK2B1,EQ,9,:BIN21)
  IF(@PK2B1,EQ,10,:BIN22)
:BIN21
  SV(@PK2B1,18)
  JP(:ENDBIN2)
:BIN22
  SV(@PK2B1,22)
:ENDBIN2
EL
;=================================================================
BL(!BIN3)
  IF(@PK3B1,EQ,9,:BIN31)
  IF(@PK3B1,EQ,10,:BIN32)
  IF(@PK3B1,EQ,11,:BIN33)
  IF(@PK3B1,EQ,12,:BIN34)
  IF(@PK3B1,EQ,13,:BIN35)
:BIN31
  SV(@PK3B1,18)
  JP(:ENDBIN3)
:BIN32
  SV(@PK3B1,22)
  JP(:ENDBIN3)
:BIN33
  SV(@PK3B1,26)
  JP(:ENDBIN3)
:BIN34
  SV(@PK3B1,19)
  JP(:ENDBIN3)
:BIN35
  SV(@PK3B1,23)
:ENDBIN3
EL

;=================================================================
BL(!BIN4)
  IF(@PK4B1,EQ,6,:BIN41)
  IF(@PK4B1,EQ,7,:BIN42)
  IF(@PK4B1,EQ,8,:BIN43)
  IF(@PK4B1,EQ,9,:BIN44)
  IF(@PK4B1,EQ,10,:BIN45)
:BIN41
  SV(@PK4B1,17)
```

```
  JP(:ENDBIN4)
:BIN42
  SV(@PK4B1,21)
  JP(:ENDBIN4)
:BIN43
  SV(@PK4B1,25)
  JP(:ENDBIN4)
:BIN44
  SV(@PK4B1,18)
  JP(:ENDBIN4)
:BIN45
  SV(@PK4B1,22)
:ENDBIN4
EL
;=====================================================================
BL(!BINS)
  IF(@PKSB1,EQ,14,:BINS1)
  IF(@PKSB1,EQ,15,:BINS2)
  IF(@PKSB1,EQ,16,:BINS3)
  IF(@PKSB1,EQ,6,:BINS4)
  IF(@PKSB1,EQ,7,:BINS5)
:BINS1
  SV(@PKSB1,27)
  JP(:ENDBINS)
:BINS2
  SV(@PKSB1,20)
  JP(:ENDBINS)
:BINS3
  SV(@PKSB1,24)
  JP(:ENDBINS)
:BINS4
  SV(@PKSB1,17)
  JP(:ENDBINS)
:BINS5
  SV(@PKSB1,21)
:ENDBINS
EL
;=====================================================================
;                    Link for packing bin
;=====================================================================
BL(!L17)                          ;Link for bin 17
  LK(!STORE3)                     ;Move in bin
  IF(OBJ@2,EQ,3000,:SILO17)
  IF(@KK4,NE,OBJ@5,WAIT)
  IF(@@CELL(3,1),NE,0,WAIT)
  TP(R,(+2,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP7)
  RM(+2,+9,0)
  JP(:PK4)
EL
BL(!L21)                          ;Link for bin 21
  LK(!STORE3)                     ;Move in bin
  IF(OBJ@2,EQ,3000,:SILO21)
  IF(@KK4,NE,OBJ@5,WAIT)
```

```
  IF(@@CELL(3,1),NE,0,WAIT)
  TP(R,(-3,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP6)
  RM(-3,+9,0)
  JP(:PK4)
EL
BL(!L25)                          ;Link for bin 25
  LK(!STORE1)                     ;Move in bin
  IF(OBJ@2,EQ,3000,:SILO25)
  IF(@KK4,NE,OBJ@5,WAIT)
  IF(@@CELL(3,1),NE,0,WAIT)
  TP(R,(-8,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP8)
  RM(-8,+9,0)
  JP(:PK4)
EL
BL(!L18)                          ;Link for bin 18
  LK(!STORE3)                     ;Move in bin
  IF(OBJ@2,EQ,3000,:SILO18)
  IF(@KK2,NE,OBJ@5,WAIT)
  IF(@@CELL(4,1),NE,0,WAIT)
  TP(R,(+4,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP10)
  RM(+4,+9,0)
  JP(:PK2)
EL
BL(!L22)                          ;Link for bin 22
  LK(!STORE3)                     ;Move in bin
  IF(OBJ@2,EQ,3000,:SILO22)
  IF(@KK2,NE,OBJ@5,WAIT)
  IF(@@CELL(4,1),NE,0,WAIT)
  TP(R,(-1,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP9)
  RM(-1,+9,0)
  JP(:PK2)
EL
BL(!L26)                          ;Link for bin 26
  LK(!STORE1)                     ;Move in bin
  IF(@KK3,NE,OBJ@5,WAIT)
  IF(@@CELL(5,1),NE,0,WAIT)
  TP(R,(+7,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP13)
  RM(+7,+9,0)
  JP(:PK3)
EL
BL(!L19)                          ;Link for bin 19
  LK(!STORE3)                     ;Move in bin
  IF(OBJ@2,EQ,3000,:SILO19)
  IF(@KK3,NE,OBJ@5,WAIT)
  IF(@@CELL(5,1),NE,0,WAIT)
```

```
  TP(R,(+4,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP12)
  RM(+4,+9,0)
  JP(:PK3)
EL
BL(!L23)                          ;Link for bin 23
  LK(!STORE3)                     ;Move in bin
  IF(OBJ@2,EQ,3000,:SILO23)
  IF(@KK3,NE,OBJ@5,WAIT)
  IF(@@CELL(5,1),NE,0,WAIT)
  TP(R,(-1,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP11)
  RM(-1,+9,0)
  JP(:PK3)
EL

;================================================================
BR(1,XY(0,23),0)
  KB(N)          ; Save 10 hour utilization statistics
  RS(#SEED)                       ; Initialize random seed.
; Initialize array
  IA(@@QUAN)      IA(@@BRK)         IA(@@PACK)
  IA(@@CELL)      IA(@@CODED)       IA(@@PRT)
  IA(@@CODE)      IA(@@DNCODE)      IA(@@BIN)
  IA(@@MIX)       IA(@@MIXC)        IA(@@PM1B)
  IA(@@PM1C)      IA(@@PM2B)        IA(@@PM2C)
  IA(@@PK4)       IA(@@PK3)         IA(@@PK2)       IA(@@PKS)
  IA(@@PKK4)      IA(@@PKK3)        IA(@@PKK2)      IA(@@PKKS)
  IA(@@PMCODE)    IA(@@PMCODE)      IA(@@PM)
  IA(%%PRT)       IA(%%BREAK)       IA(%%PK)
  IA(%%STOP)      IA(%%END)         IA(%%PROC)
  IA(%%PM)

;----------------------------------------------------------------
; Get data about tonnage required for each product code (MRP)
; Able to change these data while running interactively

OF(FEED.DAT)
:GPELL
  GD(@Z)                          ;Get pellet feed rush & ordinary
  IF(@Z,EQ,999,:ENDPELL)
  SV(@@QUAN(@COUNTER,0),@Z)
  GD(@@QUAN(@COUNTER,1))
  GD(@@QUAN(@COUNTER,2))
  GD(@@QUAN(@COUNTER,3))
  GD(@@QUAN(@COUNTER,4))
  IF(@@QUAN(@COUNTER,2),NE,3000,:JUMPP)
  AO(@SILOO,+,@@QUAN(@COUNTER,1))
:JUMPP
  AO(@PET,+,@@QUAN(@COUNTER,1))
  IF(@COUNTER,GE,8,:NEXTLINE)
    PV(*S1,@Z)
    PV(*T1,@@QUAN(@COUNTER,1))
```

```
    AO(*S1,+,*COLUMN)
    AO(*T1,+,*COLUMN)
    IV(@COUNTER)
    JP(:ENDLINE)
  :NEXTLINE
    PV(*SS1,@Z)
    PV(*TT1,@@QUAN(@COUNTER,1))
    AO(*SS1,+,*COLUMN)
    AO(*TT1,+,*COLUMN)
    IV(@COUNTER)
  :ENDLINE
  JP(:GPELL)
:ENDPELL
  SV(@PELLNUM,@COUNTER)
  SV(@MF,@PELLNUM)              ;for start mix mash feed
  SV(@STEP1,1)
  SV(@STEP0,@PELLNUM)
:GMASH
  GD(@Z)                        ;Get mash rush & ordinary
  IF(@Z,EQ,999,:ENDMASH)
  SV(@@QUAN(@COUNTER,0),@Z)
  GD(@@QUAN(@COUNTER,1))
  GD(@@QUAN(@COUNTER,2))
  GD(@@QUAN(@COUNTER,3))
  GD(@@QUAN(@COUNTER,4))
  IF(@@QUAN(@COUNTER,2),NE,3000,:JUMPM)
  AO(@SILOO,+,@@QUAN(@COUNTER,1))
:JUMPM
  AO(@CONC,+,@@QUAN(@COUNTER,1))
  PV(*S0,@Z)
  PV(*T0,@@QUAN(@COUNTER,1))
  AO(*S0,+,*COLUMN)
  AO(*T0,+,*COLUMN)
  IV(@COUNTER)
  JP(:GMASH)
:ENDMASH
  SV(@MASHNUM,@COUNTER)
:BBUS
  SV(@PEET,@PET)
  PV(XY(28,14),@CONC)
  PV(XY(28,15),@PET)
  PV(XY(28,16),@SILOO)
  AO(@DUMM,+,@CONC)
  AO(@DUMM,+,@PET)
  AO(@CONC,*,100)
  AO(@CONC,/,@DUMM)
  AO(@PET,*,100)
  AO(@PET,/,@DUMM)
  AO(@SILOO,*,100)
  AO(@SILOO,/,@DUMM)
  PV(*PERCONC,@CONC)
  PV(*PERPET,@PET)
  PV(*PERSI,@SILOO)
  PV(*NUMPEL,2)
  PV(*NUMPAC,3)
```

```
OF(PROC1.DAT)
:PROC
  GD(@CPROC)
  GD(%%PROC(@CPROC,1))
  GD(%%PROC(@CPROC,2))
  GD(%%PROC(@CPROC,3))
  GD(%%PROC(@CPROC,4))
  GD(%%PROC(@CPROC,5))
  GD(%%PROC(@CPROC,6))
  GD(%%PROC(@CPROC,7))
  GD(%%PROC(@CPROC,8))
  GD(%%PROC(@CPROC,9))
  IV(@CPROC)
  IF(@CPROC,GE,9,:NOPROC)
  JP(:PROC)
:NOPROC
CF(DAT)
ER
;================================================================
BR(2,XY(35,35),%INTERA)      ;Start at Scale 2, 2.5 or 3 ton/batch
  LK(!BINCAP)
  PV(XY(40,11),CLOCK)
  WC(8:0:0)
  IF(@B,EQ,0,:MIXPELL)
  JP(:GOH)
;-----------------------------------------------------------
:MIXMASH
  IF(@STEP0,GE,@MASHNUM,:CHKPELL) ;No mash but pellet may be left
  IF(@STEP1,GE,@PELLNUM,:NOSET)
  IF(@@BIN(1,0),EQ,0,:SETBIN1) ;Switch to mix pellet when bin16 empty
  IF(@@BIN(2,0),EQ,0,:SETBIN2) ;Switch to mix pellet when bin1 empty
:NOSET
  SV(@@QUAN(@A,1),@B)          ;Save the remaining requirement
  IF(@@QUAN(@MF,1),NE,0,:SAMM)
  SV(@A,@STEP0)
  SV(@MF,@A)
  SV(@B,@@QUAN(@A,1))          ;Get the remaining requirement
  IF(@B,EQ,0,:MIXMASH1)
  JP(:GOH)
:SETBIN1                       ;Set swing head to pellet bin16
  SV(@POSS,1)                  ;PM set1
  JP(:MIXPELL)
:SETBIN2                       ;Set swing head to pellet bin1
  SV(@POSS,2)                  ;PM set2
  JP(:MIXPELL)
:MIXMASH1                      ;Increment counter when new
  IV(@STEP0)
  JP(:MIXMASH)
:SAMM
  SV(@A,@MF)
  SV(@B,@@QUAN(@A,1))          ;Get the remaining requirement
  IF(@B,EQ,0,:MIXMASH1)        ;Get new mash when empty
  JP(:GOH)
; -----------------------------------------------------------
:MIXPELL                       ;Switch to mix grain
```

```
  IF(@STEP1,GE,@PELLNUM,:CHKMASH) ; No pellet but mash may be left
  SV(@@QUAN(@A,1),@B)           ;Save the remaining requirement
  IF(@@QUAN(@PF,1),NE,0,:SAME)
  SV(@A,@STEP1)
  SV(@PF,@A)
  SV(@B,@@QUAN(@A,1))           ;Get the remaining requirement
  IF(@B,EQ,0,:MIXPELL1)
  JP(:GOH)
:MIXPELL1                       ;Increment counter when new
  IV(@STEP1)
  JP(:MIXPELL)
:CHKPELL                            ;No mash but pellet may be left
  IF(@@QUAN(@PF,1),EQ,0,:CHKMASH)   ;No orders left then check
  JP(:MIXPELL)                      ;Pellet orders are still left
:CHKMASH                            ;No pellet but mash may be left
  IF(@@QUAN(@MF,1),NE,0,:MIXMASH)   ;No orders left then check
  WE
  IF(@@BIN(6,0),EQ,0,NEXT,:CHKMASH)
  IF(@@BIN(7,0),EQ,0,NEXT,:CHKMASH)
  IF(@@BIN(8,0),EQ,0,NEXT,:CHKMASH)
  IF(@@BIN(9,0),EQ,0,NEXT,:CHKMASH)
  IF(@@BIN(10,0),EQ,0,NEXT,:CHKMASH)
  IF(@@BIN(11,0),EQ,0,NEXT,:CHKMASH)
  IF(@@BIN(12,0),EQ,0,NEXT,:CHKMASH)
  IF(@@BIN(13,0),EQ,0,NEXT,:CHKMASH)
  IF(@@BIN(14,0),EQ,0,NEXT,:CHKMASH) ;there are no batch left in
  IF(@@BIN(15,0),EQ,0,NEXT,:CHKMASH) ;the process else waits till
  IF(@@BIN(16,0),EQ,0,:SUM)          ;none
:SAME
  SV(@A,@PF)
  SV(@B,@@QUAN(@A,1))           ;Get the remaining requirement
:GOH
  SV(OBJ@3,@@QUAN(@A,3))
  SV(OBJ@2,@@QUAN(@A,2))
  SV(OBJ@6,@@QUAN(@A,4))
  SV(OBJ@4,@A)
  IF(CLOCK,GE,%DAY1,:SUM)
  IF(@@CELL(0,1),NE,0,WAIT)    ;Machine failed
  PV(XY(40,32),%%PROC(OBJ@6,1))
  WT(%%PROC(OBJ@6,1))           ;Mixing time
  SV(%EMIX,CLOCK)
  IF(CLOCK,GE,%DAY,:SETEMIX)
:IFMIX
  IF(@A,NE,@N,:RECORD)          ;Record when product change
:NOTP
  IF(@N,GE,@PELLNUM,:MASH)      ;Mix mash
  IF(@POSS,EQ,0,:B15)           ;Mix pellet
  IF(@POSS,EQ,1,:B16)
  IF(@POSS,EQ,2,:B1)
  IF(@POSS,EQ,3,:B2)
  IF(@POSS,EQ,4,:B3)
  IF(@POSS,EQ,5,:B4)
  JP(:PM1)
:RECORD
  IV(@P)
```

```
  SV(@P1,0)
  SV(@P2,0)
  IV(@CPR)
  SV(@@PRT(@P,0),@NOBATCH)     ;Print no of batch mixed
  IF(OBJ@4,LE,@PELLNUM,:PELLET,ELSE,NEXT)
    AO(@MASHF,+,@NOBATCH)        ;Stat for mash feed production
    IF(OBJ@2,EQ,3000,:SILO)
    JP(:RESET)
:PELLET
  AO(@PELLF,+,@NOBATCH)
  IF(OBJ@2,EQ,3000,:SILO)
  JP(:RESET)
:SILO
  AO(@SILO,+,@NOBATCH)
:RESET
  SV(%LIMIT,CLOCK)
  IF(@PELLF,NE,0,:NOSPELL)
  SV(@PELLF,@NOBATCH)
:NOSPELL
  SV(@NOBATCH,0)               ;Reset no of batch count
  IF(CLOCK,GE,%DAY,:SETCLOCK)
:BKSET
  SV(%PRT,%LIMIT)              ;Finishing time of last product
  SV(%%PRT(@P,0),%PRT)
  SV(@N,@A)                    ;Set N equals incoming product
  SV(@@CODE(@P,0),@N)
  JP(:NOTP)
:SETCLOCK
  AO(%LIMIT,-,%DAY)
  JP(:BKSET)
:SETEMIX
  AO(%EMIX,-,%DAY)
  JP(:IFMIX)
;================================================================
;                          Pellet Mill #1
;================================================================
:PM1
:B16
  SV(OBJ@5,1)
  IF(@@BIN(1,0),EQ,0,:GO16)
  IF(OBJ@4,NE,@@BIN(1,2),:B15)
  IF(@@BIN(1,3),LT,OBJ@3,:B16_1)
:B16_2
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(1,0),GE,@@BIN(1,3),:B15)
:GO16
  DV(@B)
  IV(@NOBATCH)                 ;Count no. of batch mixed
  LK(!SETCOLOR)
  IF(OBJ@5,EQ,@PMM,:CONT16) ; Increment counter for dif bin while same code
  IV(@P1)
  SV(@P2,1)
:CONT161
  SV(@PMM,OBJ@5)
  SV(@@MIX(@P,@P1),16)
```

```
  SV(@@MIXC(@P,@P1),@P2)
  PM(*PPM,BIN NO.16)
  PM(*ALERT,"       ")
  LK(!SET)
  SV(@POSS,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(3,1)  ML(11,2)  MD(1,0)  MR(3,1)
  LK(!STORE3)
  IF(@PM1,NE,OBJ@5,WAIT)
  IF(@@CELL(1,1),NE,0,WAIT)   ;Machine failed?
  TP(R,(-1,+6))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP0)
  RM(-1,+6,0)                 ;Move to PM1
  JP(:PM1A)
:CONT16
  IV(@P2)
  IF(@P1,EQ,0,:SETP16,ELSE,:CONT161)
:SETP16
  SV(@P1,1)
  JP(:CONT161)
:B16_1
  SV(@@BIN(1,3),225)
  JP(:B16_2)
:B15
  SV(OBJ@5,0)
  IF(@@BIN(0,0),EQ,0,:GO15)
  IF(OBJ@4,NE,@@BIN(0,2),:PM2)
  IF(@@BIN(0,3),LT,OBJ@3,:B15_1)
:B15_2
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(0,0),GE,@@BIN(0,3),:PM2)
:GO15
  DV(@B)
  IV(@NOBATCH)                ;Count no. of batch mixed
  LK(!SETCOLOR)
  IF(OBJ@5,EQ,@PMM,:CONT15)
  IV(@P1)
  SV(@P2,1)
:CONT151
  SV(@PMM,OBJ@5)
  SV(@@MIX(@P,@P1),15)
  SV(@@MIXC(@P,@P1),@P2)
  PM(*PPM,BIN NO.15)
  PM(*ALERT,"       ")
  LK(!SET)
  SV(@POSS,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(3,2)  ML(11,2)  MD(1,1)  ML(2,1)
  LK(!STORE3)                 ;Move in bin
  IF(@PM1,NE,OBJ@5,WAIT)
  IF(@@CELL(1,1),NE,0,WAIT)   ;Machine failed?
  TP(R,(+4,+6))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP1)
```

```
  RM(+4,+6,0)                    ;Move to PM1
:PM1A
  IF(OBJ@4,NE,@MM1,:REC1)
:NOPM1
  IV(@PMM1)
  PV(XY(30,50),%%PROC(OBJ@6,2))
  WT(%%PROC(OBJ@6,2))            ;Processing time at PM #1
  SV(%EPM1,CLOCK)
  SV(@@PM(@CPM1,0),@PMM1)
  JP(:PMM1A)
:REC1
  IV(@COUNTPM1)
  IV(@CPM1)
  SV(@PM1B,0)
  SV(@PM1C,0)
  SV(@@PM(@COUNTPM1,0),@PMM1) ;Print no. of batch
:RES1
  SV(@PMM1,0)                    ;Reset no. of batch count
  SV(%PM1,CLOCK)                 ;Finishing time of last product
  IF(CLOCK,GE,%DAY,:SETPM1)
:MP1
  SV(%%PM(@COUNTPM1,0),%PM1)
  SV(@MM1,OBJ@4)                 ;Set @MM1 equals incoming product
  SV(@@PMCODE(@COUNTPM1,0),@MM1)
  WT(0:15:0)
  JP(:NOPM1)
:PMM1A
  IF(OBJ@2,EQ,3000,:OK)
    SV(@OK,0)
    JP(:OUT)
  :OK
    SV(@OK,1)
  :OUT
  IF(@POSS1,EQ,6,:B17)           ;Check packing bin from Pellet Mill #1
  IF(@POSS1,EQ,7,:B21)
  IF(@POSS1,EQ,8,:B25)
  IF(@POSS1,EQ,9,:B18)
  IF(@POSS1,EQ,10,:B22)
  IF(@POSS1,EQ,11,:B26)
  IF(@POSS1,EQ,12,:B19)
  IF(@POSS1,EQ,13,:B23)
  IF(@POSS1,EQ,14,:B27)
  IF(@POSS1,EQ,15,:B20)
  IF(@POSS1,EQ,16,:B24)
  JP(:SET1A)
:CONT15
  IV(@P2)
  IF(@P1,EQ,0,:SETP15,ELSE,:CONT151)
:SETP15
  SV(@P1,1)
  JP(:CONT151)
:B15_1
  SV(@@BIN(0,3),225)
  JP(:B15_2)
:SETPM1
```

```
  AO(%PM1,-,%DAY)
  JP(:MP1)

;=================================================================
;                        Pellet Mill #2
;=================================================================
:PM2
  IF(@USEPEL,EQ,1,:MIXMASH)
  IF(CLOCK,GE,%FINISH,:MIXMASH)   ;Operate PM2 till ending time
  IF(@@BIN(3,0),EQ,0,:SETBACK)
  JP(:B1)
:SETBACK
  SV(@POSS,2)
:B1
  SV(OBJ@5,2)
  IF(@@BIN(2,0),EQ,0,:GO1)
  IF(OBJ@4,NE,@@BIN(2,2),:B4)
  IF(@@BIN(2,3),LT,OBJ@3,:B1_1)
:B1_2
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(2,0),GE,@@BIN(2,3),:B4)
:GO1
  DV(@B)
  IV(@NOBATCH)                ;Count no. of batch mixed
  LK(!SETCOLOR)
  IF(OBJ@5,EQ,@PMM,:CONT1)
  IV(@P1)
  SV(@P2,1)
:CONT11
  SV(@PMM,OBJ@5)
  SV(@@MIX(@P,@P1),1)
  SV(@@MIXC(@P,@P1),@P2)
  PM(*PPM,BIN NO. 1)
  PM(*ALERT,"        ")
  LK(!SET)
  SV(@POSS,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(3,2)  MR(19,1)  MD(1,2)  ML(5,3)
  LK(!STORE1)                 ;move in bin
  IF(@PM2,NE,OBJ@5,WAIT)
  IF(@@CELL(2,1),NE,0,WAIT)   ;Machine failed?
  TP(R,(+5,+6))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP2)
  RM(+5,+6,0)                 ;Move to PM2
  JP(:PM2A)
:CONT1
  IV(@P2)
  IF(@P1,EQ,0,:SETP1,ELSE,:CONT11)
:SETP1
  SV(@P1,1)
  JP(:CONT11)
:B1_1
  SV(@@BIN(2,3),100)
  JP(:B1_2)
```

```
:B4
  SV(OBJ@5,5)
  IF(@@BIN(5,0),EQ,0,:GO4)
  IF(OBJ@4,NE,@@BIN(5,2),:B3)
  IF(@@BIN(5,3),LT,OBJ@3,:B4_1)
:B4_2
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(5,0),GE,@@BIN(5,3),:B3)
:GO4
  DV(@B)
  IV(@NOBATCH)                  ;Count no. of batch mixed
  LK(!SETCOLOR)
  IF(OBJ@5,EQ,@PMM,:CONT4)
  IV(@P1)
  SV(@P2,1)
:CONT41
  SV(@PMM,OBJ@5)
  SV(@@MIX(@P,@P1),4)
  SV(@@MIXC(@P,@P1),@P2)
  PM(*PPM,BIN NO. 4)
  PM(*ALERT,"        ")
  LK(!SET)
  SV(@POSS,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(3,1)  MR(19,2)  MD(1,0)  ML(1,0)
  LK(!STORE2)                   ;move in bin
  IF(@PM2,NE,OBJ@5,WAIT)
  IF(@@CELL(2,1),NE,0,WAIT)     ;Machine failed?
  TP(R,(+2,+6))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP3)
  RM(+2,+6,0)                   ;Move to PM2
  JP(:PM2A)
:CONT4
  IV(@P2)
  IF(@P1,EQ,0,:SETP4,ELSE,:CONT41)
:SETP4
  SV(@P1,1)
  JP(:CONT41)
:B4_1
  SV(@@BIN(5,3),140)
  JP(:B4_2)
:B3
  SV(OBJ@5,4)
  IF(@@BIN(4,0),EQ,0,:GO3)
  IF(OBJ@4,NE,@@BIN(4,2),:B2)
  IF(@@BIN(4,3),LT,OBJ@3,:B3_1)
:B3_2
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(4,0),GE,@@BIN(4,3),:B2)
:GO3
  DV(@B)
  IV(@NOBATCH)                  ;Count no. of batch mixed
  LK(!SETCOLOR)
  IF(OBJ@5,EQ,@PMM,:CONT3)
```

```
  IV(@P1)
  SV(@P2,1)
:CONT31
  SV(@PMM,OBJ@5)
  SV(@@MIX(@P,@P1),3)
  SV(@@MIXC(@P,@P1),@P2)
  PM(*PPM,BIN NO. 3)
  PM(*ALERT,"        ")
  LK(!SET)
  SV(@POSS,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(3,1)  MR(19,2)  MD(1,1)  MR(3,1)
  LK(!STORE2)                    ;Move in bin
  IF(@PM2,NE,OBJ@5,WAIT)
  IF(@@CELL(2,1),NE,0,WAIT)      ;Machine failed?
  TP(R,(-2,+6))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP4)
  RM(-2,+6,0)                    ;Move to PM2
  JP(:PM2A)
:CONT3
  IV(@P2)
  IF(@P1,EQ,0,:SETP3,ELSE,:CONT31)
:SETP3
  SV(@P1,1)
  JP(:CONT31)
:B3_1
  SV(@@BIN(4,3),140)
  JP(:B3_2)
:B2
  SV(OBJ@5,3)
  IF(@@BIN(3,0),EQ,0,:GO2)
  IF(OBJ@4,NE,@@BIN(3,2),:MIXMASH)
  IF(@@BIN(3,3),LT,OBJ@3,:B2_1)
:B2_2
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(3,0),GE,@@BIN(3,3),:MIXMASH,ELSE,NEXT)
:GO2
  DV(@B)
  IV(@NOBATCH)                   ;Count no. of batch mixed
  LK(!SETCOLOR)
  IF(OBJ@5,EQ,@PMM,:CONT2)
  IV(@P1)
  SV(@P2,1)
:CONT21
  SV(@PMM,OBJ@5)
  SV(@@MIX(@P,@P1),2)
  SV(@@MIXC(@P,@P1),@P2)
  PM(*PPM,BIN NO. 2)
  PM(*ALERT,"        ")
  LK(!SET)
  SV(@POSS,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(3,3)  MR(19,1)  MD(1,2)  MR(7,2)
  LK(!STORE2)                    ;Move in bin
```

```
  IF(@PM2,NE,OBJ@5,WAIT)
  IF(@@CELL(2,1),NE,0,WAIT)    ;Machine failed?
  TP(R,(-6,+6))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP5)
  RM(-6,+6,0)                  ;Move to PM2
:PM2A
  IF(OBJ@4,NE,@MM2,:REC2)
:NOPM2
  IV(@PMM2)
  PV(XY(60,50),%%PROC(OBJ@6,3))
  WT(%%PROC(OBJ@6,3))          ;Processing time at PM #2
  SV(%EPM2,CLOCK)
  SV(@@PM(@CPM2,1),@PMM2)
  JP(:PMM2A)
:REC2
  IV(@COUNTPM2)
  IV(@CPM2)
  SV(@PM2B,0)
  SV(@PM2C,0)
  SV(@@PM(@COUNTPM2,1),@PMM2) ;Print no. of batch
:RES2
  SV(@PMM2,0)                  ;Reset no. of batch count
  SV(%PM2,CLOCK)               ;Finishing time of last product
  IF(CLOCK,GE,%DAY,:SETPM2)
:MP2
  SV(%%PM(@COUNTPM2,1),%PM2)
  SV(@MM2,OBJ@4)               ;Set @MM2 equals incoming product
  SV(@@PMCODE(@COUNTPM2,1),@MM2)
  WT(0:20:0)
  JP(:NOPM2)
:PMM2A
  IF(OBJ@2,EQ,3000,:OK1)
    SV(@OK,0)
    JP(:OUT1)
  :OK1
    SV(@OK,1)
  :OUT1
  IF(@POSS2,EQ,6,:B17B)        ;Check packing bin from Pellet Mill #2
  IF(@POSS2,EQ,7,:B21B)
  IF(@POSS2,EQ,8,:B25B)
  IF(@POSS2,EQ,9,:B18B)
  IF(@POSS2,EQ,10,:B22B)
  IF(@POSS2,EQ,11,:B26B)
  IF(@POSS2,EQ,12,:B19B)
  IF(@POSS2,EQ,13,:B23B)
  IF(@POSS2,EQ,14,:B27B)
  IF(@POSS2,EQ,15,:B20B)
  IF(@POSS2,EQ,16,:B24B)
  JP(:SET2A)
:CONT2
  IV(@P2)
  IF(@P1,EQ,0,:SETP2,ELSE,:CONT21)
:SETP2
  SV(@P1,1)
```

```
  JP(:CONT21)
:B2_1
  SV(@@BIN(3,3),140)
  JP(:B2_2)
:SETPM2
  AO(%PM2,-,%DAY)
  JP(:MP2)
;=================================================================
:SET1A
:B22
  IF(@OK,EQ,1,:B27)
  SV(OBJ@5,10)
  IF(@POSS2,EQ,OBJ@5,:B18)
  IF(@@BIN(10,0),EQ,0,:GO22)
  IF(OBJ@4,NE,@@BIN(10,2),:B18)
  IF(@@BIN(10,3),LT,OBJ@3,:B221)
:B222
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(10,0),GE,@@BIN(10,3),:B18)
:GO22
  SV(@PM1BIN,10)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_22)
:CKP1_22
  IF(OBJ@5,EQ,@P1MM,:CONT22)  ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT221
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),22)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.22)
  PM(*ALERT0,"       ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,8)   MR(1,10)   RM(+0,+2,0)
  LK(!L22)
:CONT22
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP22,ELSE,:CONT221)
:SETP22
  SV(@PM1B,1)
  JP(:CONT221)
:CKP_22
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_22,ELSE,:B18)
:B221
  SV(@@BIN(10,3),260)
  JP(:B222)
:B18
  IF(@OK,EQ,1,:B27)
  SV(OBJ@5,9)
  IF(@@BIN(9,0),EQ,0,:GO18)
  IF(OBJ@4,NE,@@BIN(9,2),:B23)
  IF(@@BIN(9,3),LT,OBJ@3,:B181)
:B182
```

```
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(9,0),GE,@@BIN(9,3),:B23)
:GO18
  SV(@PM1BIN,9)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_18)
:CKP1_18
  IF(OBJ@5,EQ,@P1MM,:CONT18)  ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT181
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),18)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.18)
  PM(*ALERT0,"        ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,8)   ML(4,3)   RM(+0,+2,0)
  LK(!L18)
:CONT18
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP18,ELSE,:CONT181)
:SETP18
  SV(@PM1B,1)
  JP(:CONT181)
:CKP_18
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_18,ELSE,:B23)
:B181
  SV(@@BIN(9,3),260)
  JP(:B182)
:B23
  IF(@OK,EQ,1,:B27)
  SV(OBJ@5,13)
  IF(@@BIN(13,0),EQ,0,:GO23)
  IF(OBJ@4,NE,@@BIN(13,2),:B19)
  IF(@@BIN(13,3),LT,OBJ@3,:B231)
:B232
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(13,0),GE,@@BIN(13,3),:B19)
:GO23
  SV(@PM1BIN,13)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_23)
:CKP1_23
  IF(OBJ@5,EQ,@P1MM,:CONT23) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT231
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),23)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.23)
  PM(*ALERT0,"        ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
```

```
  IV(@@BIN(OBJ@5,0))
  MD(4,6)   MR(14,1)   RM(+0,+2,0)
  LK(!L23)
:CONT23
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP23,ELSE,:CONT231)
:SETP23
  SV(@PM1B,1)
  JP(:CONT231)
:CKP_23
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_23,ELSE,:B19)
:B231
  SV(@@BIN(13,3),260)
  JP(:B232)
:B19
  IF(@OK,EQ,1,:B27)
  SV(OBJ@5,12)
  IF(@@BIN(12,0),EQ,0,:GO19)
  IF(OBJ@4,NE,@@BIN(12,2),:B21)
  IF(@@BIN(12,3),LT,OBJ@3,:B191)
:B192
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(12,0),GE,@@BIN(12,3),:B21)
:GO19
  SV(@PM1BIN,12)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_19)
:CKP1_19
  IF(OBJ@5,EQ,@P1MM,:CONT19) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT191
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),19)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.19)
  PM(*ALERT0,"       ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,1)   MR(9,4)   RM(+0,+2,0)
  LK(!L19)
:CONT19
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP19,ELSE,:CONT191)
:SETP19
  SV(@PM1B,1)
  JP(:CONT191)
:CKP_19
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_19,ELSE,:B21)
:B191
  SV(@@BIN(12,3),260)
  JP(:B192)
:B21
  IF(@OK,EQ,1,:B27)
  SV(OBJ@5,7)
```

```
  IF(@@BIN(7,0),EQ,0,:GO21)
  IF(OBJ@4,NE,@@BIN(7,2),:B17)
  IF(@@BIN(7,3),LT,OBJ@3,:B211)
:B212
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(7,0),GE,@@BIN(7,3),:B17)
:GO21
  SV(@PM1BIN,7)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_21)
:CKP1_21
  IF(OBJ@5,EQ,@P1MM,:CCONT21) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT211
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),21)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.21)
  PM(*ALERT0,"       ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,9)   ML(12,1)   RM(+0,+2,0)
  LK(!L21)
:CCONT21
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP21,ELSE,:CONT211)
:SETP21
  SV(@PM1B,1)
  JP(:CONT211)
:CKP_21
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_21,ELSE,:B17)
:B211
  SV(@@BIN(7,3),260)
  JP(:B212)
:B17
  IF(@OK,EQ,1,:B27)
  SV(OBJ@5,6)
  IF(@@BIN(6,0),EQ,0,:GO17)
  IF(OBJ@4,NE,@@BIN(6,2),:B26)
  IF(@@BIN(6,3),LT,OBJ@3,:B171)
:B172
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(6,0),GE,@@BIN(6,3),:B26)
:GO17
  SV(@PM1BIN,6)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_17)
:CKP1_17
  IF(OBJ@5,EQ,@P1MM,:CONT17) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT171
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),17)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
```

```
  PM(*PPM1,BIN NO.17)
  PM(*ALERT0,"       ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,8)   ML(17,1)   RM(+0,+2,0)
  LK(!L17)
:CONT17
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP17,ELSE,:CONT171)
:SETP17
  SV(@PM1B,1)
  JP(:CONT171)
:CKP_17
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_17,ELSE,:B26)
:B171
  SV(@@BIN(6,3),260)
  JP(:B172)
:B26
  IF(@OK,EQ,1,:B27)
  SV(OBJ@5,11)
  IF(@@BIN(11,0),EQ,0,:GO26)
  IF(OBJ@4,NE,@@BIN(11,2),:B25)
  IF(@@BIN(11,3),LT,OBJ@3,:B261)
:B262
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(11,0),GE,@@BIN(11,3),:B25)
:GO26
  SV(@PM1BIN,11)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_26)
:CKP1_26
  IF(OBJ@5,EQ,@P1MM,:CONT26) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT261
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),26)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.26)
  PM(*ALERT0,"       ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,8)   MR(4,2)   RM(+0,+2,0)
  LK(!L26)
:CONT26
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP26,ELSE,:CONT261)
:SETP26
  SV(@PM1B,1)
  JP(:CONT261)
:CKP_26
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_26,ELSE,:B25)
:B261
  SV(@@BIN(11,3),100)
```

```
  JP(:B262)
:B25
  IF(@OK,EQ,1,:B27)
  SV(OBJ@5,8)
  IF(@@BIN(8,0),EQ,0,:GO25)
  IF(OBJ@4,NE,@@BIN(8,2),:PM1WAIT_D)
  IF(@@BIN(8,3),LT,OBJ@3,:B251)
:B252
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(8,0),GE,@@BIN(8,3),:PM1WAIT_D)
:GO25
  SV(@PM1BIN,8)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_25)
:CKP1_25
  IF(OBJ@5,EQ,@P1MM,:CONT25) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT251
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),25)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.25)
  PM(*ALERT0,"        ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,8)   ML(9,2)   RM(+0,+2,0)
  LK(!L25)
:CONT25
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP25,ELSE,:CONT251)
:SETP25
  SV(@PM1B,1)
  JP(:CONT251)
:CKP_25
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_25,ELSE,:PM1WAIT_D)
:B251
  SV(@@BIN(8,3),100)
  JP(:B252)
:B27
  IF(@OK,EQ,0,:B22)
  SV(OBJ@5,14)
  IF(@@BIN(14,0),EQ,0,:GO27)
  IF(OBJ@4,NE,@@BIN(14,2),:B20)
  IF(@@BIN(14,3),LT,OBJ@3,:B271)
:B272
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(14,0),GE,@@BIN(14,3),:B20)
:GO27
  SV(@PM1BIN,14)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_27)
:CKP1_27
  IF(OBJ@5,EQ,@P1MM,:CONT27) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
```

```
:CONT271
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),27)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.27)
  PM(*ALERT0,"        ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,5)   MR(17,1)   RM(+0,+2,0)
  LK(!STORE1)                  ;move in bin
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+16,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP14)
  RM(+16,+9,0)
  JP(:SILOPACK)
:CONT27
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP27,ELSE,:CONT271)
:SETP27
  SV(@PM1B,1)
  JP(:CONT271)
:CKP_27
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_27,ELSE,:B20)
:B271
  SV(@@BIN(14,3),100)
  JP(:B272)
:B20
  IF(@OK,EQ,0,:B22)
  SV(OBJ@5,15)
  IF(@@BIN(15,0),EQ,0,:GO20)
  IF(OBJ@4,NE,@@BIN(15,2),:B24)
  IF(@@BIN(15,3),LT,OBJ@3,:B201)
:B202
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(15,0),GE,@@BIN(15,3),:B24)
:GO20
  SV(@PM1BIN,15)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_20)
:CKP1_20
  IF(OBJ@5,EQ,@P1MM,:CONT20) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT201
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),20)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.20)
  PM(*ALERT0,"        ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,3)   MR(22,1)   RM(+0,+2,0)
```

```
  LK(!STORE4)                ;move in bin
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+13,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP15)
  RM(+13,+9,0)
  JP(:SILOPACK)
:CONT20
  IV(@PM1C)
  IF(@PM1B,EQ,0,:SETP20,ELSE,:CONT201)
:SETP20
  SV(@PM1B,1)
  JP(:CONT201)
:CKP_20
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_20,ELSE,:B24)
:B201
  SV(@@BIN(15,3),360)
  JP(:B202)
:B24
  IF(@OK,EQ,0,:B22)
  SV(OBJ@5,16)
  IF(@@BIN(16,0),EQ,0,:GO24)
  IF(OBJ@4,NE,@@BIN(16,2),:PM1WAIT_D)
  IF(@@BIN(16,3),LT,OBJ@3,:B241)
:B242
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(16,0),GE,@@BIN(16,3),:PM1WAIT_D)
:GO24
  SV(@PM1BIN,16)
  IF(@PM1BIN,EQ,@PM2BIN,:CKP_24)
:CKP1_24
  IF(OBJ@5,EQ,@P1MM,:CONT24) ;Increment counter for dif bin while same code
  IV(@PM1B)
  SV(@PM1C,1)
:CONT241
  SV(@P1MM,OBJ@5)
  SV(@@PM1B(@COUNTPM1,@PM1B),24)
  SV(@@PM1C(@COUNTPM1,@PM1B),@PM1C)
  PM(*PPM1,BIN NO.24)
  PM(*ALERT0,"        ")
  LK(!SET)
  SV(@POSS1,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(4,1)   MR(29,1)   RM(+0,+2,0)
  LK(!STORE4)                ;move in bin
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+6,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP16)
  RM(+6,+9,0)
  JP(:SILOPACK)
:CONT24
  IV(@PM1C)
```

```
  IF(@PM1B,EQ,0,:SETP24,ELSE,:CONT241)
:SETP24
  SV(@PM1B,1)
  JP(:CONT241)
:CKP_24
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM2,1),:CKP1_24,ELSE,:PM1WAIT_D)
:B241
  SV(@@BIN(16,3),360)
  JP(:B242)
;========================== Mash Feed ==========================
:MASH
  IF(@POSS3,EQ,6,:B17A)
  IF(@POSS3,EQ,7,:B21A)
:B17A
  SV(OBJ@5,6)
  IF(@@BIN(6,0),EQ,0,:GO17A)
  IF(OBJ@4,NE,@@BIN(6,2),:B21A)
  IF(@@BIN(6,3),LT,OBJ@3,:B171A)
:B172A
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(6,0),GE,@@BIN(6,3),:B21A)
:GO17A
  DV(@B)
  IV(@NOBATCH)                ;Count no. of batch mixed
  LK(!SETCOLOR)
  IF(OBJ@5,EQ,@PMM,:CONT17A)
  IV(@P1)
  SV(@P2,1)
:CONT17A1
  SV(@PMM,OBJ@5)
  SV(@@MIX(@P,@P1),17)
  SV(@@MIXC(@P,@P1),@P2)
  PM(*PPM,BIN NO.17)
  PM(*ALERT,"       ")
  LK(!SET)
  SV(@POSS3,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(2,0)  ML(28,1)  MD(16,1)
  RM(+0,+4,0)
  LK(!L17)
:CONT17A
  IV(@P2)
  IF(@P1,EQ,0,:SETP17A,ELSE,:CONT17A1)
:SETP17A
  SV(@P1,1)
  JP(:CONT17A1)
:B171A
  SV(@@BIN(6,3),260)
  JP(:B172A)
:B21A
  SV(OBJ@5,7)
  IF(@@BIN(7,0),EQ,0,:GO21A)
  IF(OBJ@4,NE,@@BIN(7,2),:MIXWAIT0)
  IF(@@BIN(7,3),LT,OBJ@3,:B211A)
:B212A
```

```
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(7,0),GE,@@BIN(7,3),:MIXWAIT0)
:GO21A
  DV(@B)
  IV(@NOBATCH)                    ;Count no. of batch mixed
  LK(!SETCOLOR)
  IF(OBJ@5,EQ,@PMM,:CONT21A)
  IV(@P1)
  SV(@P2,1)
:CONT21A1
  SV(@PMM,OBJ@5)
  SV(@@MIX(@P,@P1),21)
  SV(@@MIXC(@P,@P1),@P2)
  PM(*PPM,BIN NO.21)
  PM(*ALERT,"       ")
  LK(!SET)
  SV(@POSS3,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(2,0)  ML(28,1)  MD(16,1)  MR(5,0)
  RM(+0,+4,0)
  LK(!L21)
:CONT21A
  IV(@P2)
  IF(@P1,EQ,0,:SETP21A,ELSE,:CONT21A1)
:SETP21A
  SV(@P1,1)
  JP(:CONT21A1)
:B211A
  SV(@@BIN(7,3),260)
  JP(:B212A)
;=================================================================
;                     Wait at Pellet Mill #1
;=================================================================
:PM1WAIT_D                        ;Different product code & Over Capacity
  PM(*ALERT0,WAIT)
:JUMP1_D
  WE
  IF(@@BIN(10,0),EQ,0,:B22)
  IF(@@BIN(09,0),EQ,0,:B18)
  IF(@@BIN(13,0),EQ,0,:B23)
  IF(@@BIN(12,0),EQ,0,:B19)
  IF(@@BIN(07,0),EQ,0,:B21)
  IF(@@BIN(06,0),EQ,0,:B17)
  IF(@@BIN(11,0),EQ,0,:B26)
  IF(@@BIN(08,0),EQ,0,:B25)
  IF(@@BIN(14,0),EQ,0,:B27)
  IF(@@BIN(15,0),EQ,0,:B20)
  IF(@@BIN(16,0),EQ,0,:B24,ELSE,:JUMP1_D)
;=================================================================
;                  Packing Set => 1
;=================================================================
:PK4
  IF(OBJ@4,NE,@PK4,:RECORD4)      ;When prod change
  IF(OBJ@5,NE,@PK4BIN,:RECORD41) ;When bin change
:NOPK4
```

```
  SV(@@PACK(0,0),OBJ@4)          ;Set Product code being packed
  SV(@PK4BIN,OBJ@5)
  IV(@PKK4)
  IF(@COUNTPK41,EQ,0,:SETPK41,ELSE,NEXT)
:NOPK41
  LK(!PROCPK4)                   ;Get the appropriate processing time
  RM(+0,+1,0)
  SV(@@PK4(@COUNTPK4,@COUNTPK41),@PKK4)
  SV(@@PKK4(@COUNTPK4,@COUNTPK41),@PK4BIN)
  JP(:EXIT)
:SETPK41
  SV(@COUNTPK41,1)
  JP(:NOPK41)
:RECORD4                         ;When prod change
  IV(@COUNTPK4)
  SV(@PKK4,1)
  SV(@COUNTPK41,1)
  JP(:ENDPK4)
:RECORD41                        ;When bin change
  IV(@COUNTPK41)
  SV(@PKK4,1)
:ENDPK4
  SV(@PKK4,0)                         ; Reset no. of batch count
  SV(%PK4,CLOCK)                      ; Finishing time of last prod.
  IF(CLOCK,GE,%DAY,:SETPK4)
:KP4
  SV(%%PK(@COUNTPK4,0),%PK4)
  SV(@PK4,OBJ@4)                      ; Set @PK4 equals incoming prod.
  SV(@@PKCODE(@COUNTPK4,0),@PK4) ; Print prod. code
  WT(0:5:0)
  JP(:NOPK4)
:PK2
  IF(OBJ@4,NE,@PK2,:RECORD2)     ;When prod change
  IF(OBJ@5,NE,@PK2BIN,:RECORD21) ;When bin change
:NOPK2
  SV(@@PACK(1,0),OBJ@4)          ;Set Product code being packed
  SV(@PK2BIN,OBJ@5)
  IV(@PKK2)
  IF(@COUNTPK21,EQ,0,:SETPK21,ELSE,NEXT)
:NOPK21
  LK(!PROCPK2)                   ;Get the appropriate processing time
  RM(+0,+1,0)
  SV(@@PK2(@COUNTPK2,@COUNTPK21),@PKK2) ; Print no. of batch packed
  SV(@@PKK2(@COUNTPK2,@COUNTPK21),@PK2BIN)  ; Print bin no.
  JP(:EXIT)
:SETPK21
  SV(@COUNTPK21,1)
  JP(:NOPK21)
:RECORD2                         ;When prod change
  IV(@COUNTPK2)
  SV(@PKK2,1)
  SV(@COUNTPK21,1)
  JP(:ENDPK2)
:RECORD21                        ;When bin change
  IV(@COUNTPK21)
```

```
  SV(@PKK2,1)
:ENDPK2
  SV(@PKK2,0)                      ;Reset no. of batch count
  SV(%PK2,CLOCK)                   ;Finishing time of last prod.
  IF(CLOCK,GE,%DAY,:SETPK2)
:KP2
  SV(%%PK(@COUNTPK2,1),%PK2)
  SV(@PK2,OBJ@4)                   ;Set @PK4 equals incoming prod.
  SV(@@PKCODE(@COUNTPK2,1),@PK2) ; Print prod. code
  WT(0:5:0)
  JP(:NOPK2)
:PK3
  IF(OBJ@4,NE,@PK3,:RECORD3)       ;When prod change
  IF(OBJ@5,NE,@PK3BIN,:RECORD31) ;When bin change
:NOPK3
  SV(@@PACK(2,0),OBJ@4)        ;Set Product code being packed
  SV(@PK3BIN,OBJ@5)
  IV(@PKK3)
  IF(@COUNTPK31,EQ,0,:SETPK31,ELSE,NEXT)
:NOPK31
  LK(!PROCPK3)                 ;Get the appropriate processing time
  RM(+0,+1,0)
  SV(@@PK3(@COUNTPK3,@COUNTPK31),@PKK3)
  SV(@@PKK3(@COUNTPK3,@COUNTPK31),@PK3BIN)
  JP(:EXIT)
:SETPK31
  SV(@COUNTPK31,1)
  JP(:NOPK31)
:RECORD3                           ;When prod change
  IV(@COUNTPK3)
  SV(@PKK3,1)
  SV(@COUNTPK31,1)
  JP(:ENDPK3)
:RECORD31                          ;When bin change
  IV(@COUNTPK31)
  SV(@PKK3,1)
:ENDPK3
  SV(@PKK3,0)                       ; Reset no. of batch count
  SV(%PK3,CLOCK)                   ; Finishing time of last prod.
  IF(CLOCK,GE,%DAY,:SETPK3)
:KP3
  SV(%%PK(@COUNTPK3,2),%PK3)
  SV(@PK3,OBJ@4)                   ; Set @PK4 equals incoming prod.
  SV(@@PKCODE(@COUNTPK3,2),@PK3)   ; Print prod. code
  WT(0:5:0)
  JP(:NOPK3)
:SILOPACK
  IF(OBJ@4,NE,@PKS,:RECORDS)       ; When prod change
  IF(OBJ@5,NE,@PKSBIN,:RECORDS1)   ; When bin change
:NOSILO
  SV(@@PACK(3,0),OBJ@4)        ;Set Product code being packed
  SV(@PKSBIN,OBJ@5)
  IV(@PKKS)
  IV(@PSILO)
  IF(@COUNTS1,EQ,0,:SETS1,ELSE,NEXT)
```

```
:NOSILO1
  WT(0:5:0)                    ;Processing time of silo
  RM(+0,+1,0)
  PV(XY(32,83),@PKSBIN)
  PV(XY(55,83),OBJ@4)
  PV(XY(49,85),@PSILO)
  SV(@@PKS(@COUNTS,@COUNTS1),@PKKS)
  SV(@@PKKS(@COUNTS,@COUNTS1),@PKSBIN)
  PV(*PTS,@PSILO)
  JP(:ENDROUT)
:SETS1
  SV(@COUNTS1,1)
  JP(:NOSILO1)
:RECORDS                                ; When prod change
  IV(@COUNTS)
  SV(@PKKS,1)
  SV(@COUNTS1,1)
  JP(:ENDS)
:RECORDS1                               ; When bin change
  IV(@COUNTS1)
  SV(@PKKS,1)
:ENDS
  SV(@PKKS,0)                           ; Reset no. of batch count
  SV(%PKS,CLOCK)                        ; Finishing time of last prod.
  IF(CLOCK,GE,%DAY,:SETPKS)
:KPS
  PV(XY(24,85),%PKS)
  SV(%%PK(@COUNTS,3),%PKS)
  SV(@PKS,OBJ@4)                        ; Set @PK4 equals incoming prod.
  SV(@@PKCODE(@COUNTS,3),@PKS) ; Print prod. code
  WT(0:5:0)
  JP(:NOSILO)
:SETPK4
  AO(%PK4,-,%DAY)
  JP(:KP4)
:SETPK2
  AO(%PK2,-,%DAY)
  JP(:KP2)
:SETPK3
  AO(%PK3,-,%DAY)
  JP(:KP3)
:SETPKS
  AO(%PKS,-,%DAY)
  JP(:KPS)
:EXIT
  IF(OBJ@4,LT,@PELLNUM,:COUNTPELL)
  IV(@COUNTM)
  PV(*PTM,@COUNTM)
  JP(:TOTALP)
:COUNTPELL
  IV(@COUNTP)
  PV(*PTP,@COUNTP)
  JP(:TOTALP)
;================================================================
:SILO17
```

```
  IF(@@BIN(14,0),EQ,0,NEXT)
  IF(@@BIN(15,0),EQ,0,NEXT)
  IF(@@BIN(16,0),EQ,0,NEXT)
  SV(@KKS,6)
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+52,+9))
  DV(@@BIN(OBJ@5,0))
  RM(+52,+9,0)
  JP(:SILOPACK)
:SILO21
  IF(@@BIN(14,0),EQ,0,NEXT)
  IF(@@BIN(15,0),EQ,0,NEXT)
  IF(@@BIN(16,0),EQ,0,NEXT)
  SV(@KKS,7)
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+47,+9))
  DV(@@BIN(OBJ@5,0))
  RM(+47,+9,0)
  JP(:SILOPACK)
:SILO25
  IF(@@BIN(14,0),EQ,0,NEXT)
  IF(@@BIN(15,0),EQ,0,NEXT)
  IF(@@BIN(16,0),EQ,0,NEXT)
  SV(@KKS,8)
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+42,+9))
  DV(@@BIN(OBJ@5,0))
  RM(+42,+9,0)
  JP(:SILOPACK)
:SILO18
  IF(@@BIN(14,0),EQ,0,NEXT)
  IF(@@BIN(15,0),EQ,0,NEXT)
  IF(@@BIN(16,0),EQ,0,NEXT)
  SV(@KKS,9)
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+39,+9))
  DV(@@BIN(OBJ@5,0))
  RM(+39,+9,0)
  JP(:SILOPACK)
:SILO22
  IF(@@BIN(14,0),EQ,0,NEXT)
  IF(@@BIN(15,0),EQ,0,NEXT)
  IF(@@BIN(16,0),EQ,0,NEXT)
  SV(@KKS,10)
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+34,+9))
  DV(@@BIN(OBJ@5,0))
  RM(+34,+9,0)
  JP(:SILOPACK)
:SILO19
```

```
  IF(@@BIN(14,0),EQ,0,NEXT)
  IF(@@BIN(15,0),EQ,0,NEXT)
  IF(@@BIN(16,0),EQ,0,NEXT)
  SV(@KKS,12)
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+26,+9))
  DV(@@BIN(OBJ@5,0))
  RM(+26,+9,0)
  JP(:SILOPACK)
:SILO23
  IF(@@BIN(14,0),EQ,0,NEXT)
  IF(@@BIN(15,0),EQ,0,NEXT)
  IF(@@BIN(16,0),EQ,0,NEXT)
  SV(@KKS,13)
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+21,+9))
  DV(@@BIN(OBJ@5,0))
  RM(+21,+9,0)
  JP(:SILOPACK)
;=================================================================
:SET2A
:B18B
  IF(@OK,EQ,1,:B27B)
  SV(OBJ@5,9)
  IF(@POSS1,EQ,OBJ@5,:B22B)
  IF(@@BIN(9,0),EQ,0,:GO18B)
  IF(OBJ@4,NE,@@BIN(9,2),:B22B)
  IF(@@BIN(9,3),LT,OBJ@3,:B181B)
:B182B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(9,0),GE,@@BIN(9,3),:B22B)
:GO18B
  SV(@PM2BIN,9)
  IF(@PM2BIN,EQ,@PM1BIN,:CKP_18B)
:CKP1_18B
  IF(OBJ@5,EQ,@P2MM,:CONT18B) ; Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT18B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),18)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.18)
  PM(*ALERT1,"        ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,3)   ML(34,1)
  LK(!L18)
:CONT18B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP18B,ELSE,:CONT18B1)
:SETP18B
```

```
  SV(@PM2B,1)
  JP(:CONT18B1)
:CKP_18B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_18B,ELSE,:B22B)
:B181B
  SV(@@BIN(9,3),260)
  JP(:B182B)
:B22B
  IF(@OK,EQ,1,:B27B)
  SV(OBJ@5,10)
  IF(@POSS1,EQ,OBJ@5,:B23B)
  IF(@@BIN(10,0),EQ,0,:GO22B)
  IF(OBJ@4,NE,@@BIN(10,2),:B23B)
  IF(@@BIN(10,3),LT,OBJ@3,:B221B)
:B222B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(10,0),GE,@@BIN(10,3),:B23B)
:GO22B
  SV(@PM2BIN,10)
  IF(@PM2BIN,EQ,@PM1BIN,:CKP_22B)
:CKP1_22B
  IF(OBJ@5,EQ,@P2MM,:CONT22B) ; Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT22B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),22)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.22)
  PM(*ALERT1,"        ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,3)   ML(29,1)
  LK(!L22)
:CONT22B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP22B,ELSE,:CONT22B1)
:SETP22B
  SV(@PM2B,1)
  JP(:CONT22B1)
:CKP_22B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_22B,ELSE,:B23B)
:B221B
  SV(@@BIN(10,3),260)
  JP(:B222B)
:B23B
  IF(@OK,EQ,1,:B27B)
  SV(OBJ@5,13)
  IF(@POSS1,EQ,OBJ@5,:B19B)
  IF(@@BIN(13,0),EQ,0,:GO23B)
  IF(OBJ@4,NE,@@BIN(13,2),:B19B)
  IF(@@BIN(13,3),LT,OBJ@3,:B231B)
:B232B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
```

```
  IF(@@BIN(13,0),GE,@@BIN(13,3),:B19B)
:GO23B
  SV(@PM2BIN,13)
  IF(@PM2BIN,EQ,@PM1BIN,:CKP_23B)
:CKP1_23B
  IF(OBJ@5,EQ,@P2MM,:CONT23B) ; Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT23B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),23)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.23)
  PM(*ALERT1,"       ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,2)   ML(16,2)
  LK(!L23)
:CONT23B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP23B,ELSE,:CONT23B1)
:SETP23B
  SV(@PM2B,1)
  JP(:CONT23B1)
:CKP_23B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_23B,ELSE,:B19B)
:B231B
  SV(@@BIN(13,3),260)
  JP(:B232B)
:B19B
  IF(@OK,EQ,1,:B27B)
  SV(OBJ@5,12)
  IF(@@BIN(12,0),EQ,0,:GO19B)
  IF(OBJ@4,NE,@@BIN(12,2),:B21B)
  IF(@@BIN(12,3),LT,OBJ@3,:B191B)
:B192B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(12,0),GE,@@BIN(12,3),:B21B)
:GO19B
  SV(@PM2BIN,12)
  IF(@PM2BIN,EQ,@PM1BIN,:CKP_19B)
:CKP1_19B
  IF(OBJ@5,EQ,@P2MM,:CONT19B) ; Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT19B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),19)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.19)
  PM(*ALERT1,"       ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
```

```
  MD(6,4)   ML(21,1)
  LK(!L19)
:CONT19B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP19B,ELSE,:CONT19B1)
:SETP19B
  SV(@PM2B,1)
  JP(:CONT19B1)
:CKP_19B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_19B,ELSE,:B21B)
:B191B
  SV(@@BIN(12,3),260)
  JP(:B192B)
:B21B
  IF(@OK,EQ,1,:B27B)
  SV(OBJ@5,7)
  IF(@@BIN(7,0),EQ,0,:GO21B)
  IF(OBJ@4,NE,@@BIN(7,2),:B17B)
  IF(@@BIN(7,3),LT,OBJ@3,:B211B)
:B212B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(7,0),GE,@@BIN(7,3),:B17B)
:GO21B
  SV(@PM2BIN,7)
  IF(@PM2BIN,EQ,@PM2BIN,:CKP_21B)
:CKP1_21B
  IF(OBJ@5,EQ,@P2MM,:CONT21B) ; Increment counter for dif bin while same coc
  IV(@PM2B)
  SV(@PM2C,1)
:CONT21B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),21)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.21)
  PM(*ALERT1,"       ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,2)   ML(42,1)
  LK(!L21)
:CONT21B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP21B,ELSE,:CONT21B1)
:SETP21B
  SV(@PM2B,1)
  JP(:CONT21B1)
:CKP_21B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_21B,ELSE,:B17B)
:B211B
  SV(@@BIN(7,3),260)
  JP(:B212B)
:B17B
  IF(@OK,EQ,1,:B27B)
  SV(OBJ@5,6)
  IF(@@BIN(6,0),EQ,0,:GO17B)
```

```
  IF(OBJ@4,NE,@@BIN(6,2),:B26B)
  IF(@@BIN(6,3),LT,OBJ@3,:B171B)
:B172B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(6,0),GE,@@BIN(6,3),:B26B)
:GO17B
  SV(@PM2BIN,6)
  IF(@PM2BIN,EQ,@PM2BIN,:CKP_17B)
:CKP1_17B
  IF(OBJ@5,EQ,@P2MM,:CONT17B) ; Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT17B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),17)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.17)
  PM(*ALERT1,"        ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,1.5)   ML(47,1)
  LK(!L17)
:CONT17B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP17B,ELSE,:CONT17B1)
:SETP17B
  SV(@PM2B,1)
  JP(:CONT17B1)
:CKP_17B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_17B,ELSE,:B26B)
:B171B
  SV(@@BIN(6,3),260)
  JP(:B172B)
:B26B
  IF(@OK,EQ,1,:B27B)
  SV(OBJ@5,11)
  IF(@@BIN(11,0),EQ,0,:GO26B)
  IF(OBJ@4,NE,@@BIN(11,2),:B25B)
  IF(@@BIN(11,3),LT,OBJ@3,:B261B)
:B262B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(11,0),GE,@@BIN(11,3),:B25B)
:GO26B
  SV(@PM2BIN,11)
  IF(@PM2BIN,EQ,@PM2BIN,:CKP_26B)
:CKP1_26B
  IF(OBJ@5,EQ,@P2MM,:CONT26B) ;Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT26B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),26)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.26)
```

```
  PM(*ALERT1,"       ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,3.5)   ML(26,1)
  LK(!L26)
:CONT26B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP26B,ELSE,:CONT26B1)
:SETP26B
  SV(@PM2B,1)
  JP(:CONT26B1)
:CKP_26B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_26B,ELSE,:B25B)
:B261B
  SV(@@BIN(11,3),100)
  JP(:B262B)
:B25B
  IF(@OK,EQ,1,:B27B)
  SV(OBJ@5,8)
  IF(@@BIN(8,0),EQ,0,:GO25B)
  IF(OBJ@4,NE,@@BIN(8,2),:PM2WAIT_D)
  IF(@@BIN(8,3),LT,OBJ@3,:B251B)
:B252B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(8,0),GE,@@BIN(8,3),:PM2WAIT_D)
:GO25B
  SV(@PM2BIN,8)
  IF(@PM2BIN,EQ,@PM2BIN,:CKP_25B)
:CKP1_25B
  IF(OBJ@5,EQ,@P2MM,:CONT25B) ; Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT25B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),25)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.25)
  PM(*ALERT1,"       ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,2.3)   ML(39,1)
  LK(!L25)
:CONT25B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP25B,ELSE,:CONT25B1)
:SETP25B
  SV(@PM2B,1)
  JP(:CONT25B1)
:CKP_25B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_25B,ELSE,:PM2WAIT_D)
:B251B
  SV(@@BIN(8,3),100)
  JP(:B252B)
```

```
:B27B
  IF(@OK,EQ,0,:B22B)
  SV(OBJ@5,14)
  IF(@@BIN(14,0),EQ,0,:GO27B)
  IF(OBJ@4,NE,@@BIN(14,2),:B20B)
  IF(@@BIN(14,3),LT,OBJ@3,:B271B)
:B272B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(14,0),GE,@@BIN(14,3),:B20B)
:GO27B
  SV(@PM2BIN,14)
  IF(@PM2BIN,EQ,@PM2BIN,:CKP_27B)
:CKP1_27B
  IF(OBJ@5,EQ,@P2MM,:CONT27B) ; Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT27B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),27)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.27)
  PM(*ALERT1,"        ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,5)   ML(13,1)
  LK(!STORE1)                    ;move in bin
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+16,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP14)
  RM(+16,+9,0)
  JP(:SILOPACK)
:CONT27B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP27B,ELSE,:CONT27B1)
:SETP27B
  SV(@PM2B,1)
  JP(:CONT27B1)
:CKP_27B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_27B,ELSE,:B20B)
:B271B
  SV(@@BIN(14,3),100)
  JP(:B272B)
:B20B
  IF(@OK,EQ,0,:B22B)
  SV(OBJ@5,15)
  IF(@@BIN(15,0),EQ,0,:GO20B)
  IF(OBJ@4,NE,@@BIN(15,2),:B24B)
  IF(@@BIN(15,3),LT,OBJ@3,:B201B)
:B202B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(15,0),GE,@@BIN(15,3),:B24B)
:GO20B
```

```
  SV(@PM2BIN,15)
  IF(@PM2BIN,EQ,@PM2BIN,:CKP_20B)
:CKP1_20B
  IF(OBJ@5,EQ,@P2MM,:CONT20B) ; Increment counter for dif bin while same code
  IV(@PM2B)
  SV(@PM2C,1)
:CONT20B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),20)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
  PM(*PPM2,BIN NO.20)
  PM(*ALERT1,"       ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,4)   ML(8,2)
  LK(!STORE4)                ;move in bin
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+13,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP15)
  RM(+13,+9,0)
  JP(:SILOPACK)
:CONT20B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP20B,ELSE,:CONT20B1)
:SETP20B
  SV(@PM2B,1)
  JP(:CONT20B1)
:CKP_20B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_20B,ELSE,:B24B)
:B201B
  SV(@@BIN(15,3),360)
  JP(:B202B)
:B24B
  IF(@OK,EQ,0,:B22B)
  SV(OBJ@5,16)
  IF(@@BIN(16,0),EQ,0,:GO24B)
  IF(OBJ@4,NE,@@BIN(16,2),:PM2WAIT_D)
  IF(@@BIN(16,3),LT,OBJ@3,:B241B)
:B242B
  AO(@@BIN(OBJ@5,3),/,OBJ@3)
  IF(@@BIN(16,0),GE,@@BIN(16,3),:PM2WAIT_D)
:GO24B
  SV(@PM2BIN,16)
  IF(@PM2BIN,EQ,@PM2BIN,:CKP_24B)
:CKP1_24B
  IF(OBJ@5,EQ,@P2MM,:CONT24B) ; Increment counter for dif bin while same cod
  IV(@PM2B)
  SV(@PM2C,1)
:CONT24B1
  SV(@P2MM,OBJ@5)
  SV(@@PM2B(@COUNTPM2,@PM2B),24)
  SV(@@PM2C(@COUNTPM2,@PM2B),@PM2C)
```

```
  PM(*PPM2,BIN NO.24)
  PM(*ALERT1,"       ")
  LK(!SET)
  SV(@POSS2,OBJ@5)
  IV(@@BIN(OBJ@5,0))
  MD(6,6)   ML(1,2)
  LK(!STORE4)                  ;move in bin
  IF(@KKS,NE,OBJ@5,WAIT)
  IF(@@CELL(7,1),NE,0,WAIT)
  TP(R,(+6,+9))
  DV(@@BIN(OBJ@5,0))
  LK(!TTP16)
  RM(+6,+9,0)
  JP(:SILOPACK)
:CONT24B
  IV(@PM2C)
  IF(@PM2B,EQ,0,:SETP24B,ELSE,:CONT24B1)
:SETP24B
  SV(@PM2B,1)
  JP(:CONT24B1)
:CKP_24B
  IF(OBJ@4,LE,@@PMCODE(@COUNTPM1,0),:CKP1_24B,ELSE,:PM2WAIT_D)
:B241B
  SV(@@BIN(16,3),360)
  JP(:B242B)
;=================================================================
;                     Time lost at Mixer
;=================================================================
:MIXWAIT0                      ; Over capacity at bin #21
  SV(%STOP,CLOCK)
  PM(*ALERT,WAIT )
  IF(CLOCK,GE,%DAY,:REMIX0)
:MIX0
; -------- Heuristics when to make mixer up again -----------
  WE
  ;Pellet bins
    IF(@STEP1,GE,@PELLNUM,:WTPACK)
    IF(@@BIN(1,0),EQ,0,:PAC0)
    IF(@@BIN(0,0),EQ,0,:PAC1)
    IF(@@BIN(2,0),EQ,0,:PAC2)
  ;Packing bins
  :WTPACK
    IF(@@BIN(6,0),EQ,0,:PAC6)      ;Utilization rate at PKs is
    IF(@@BIN(7,0),EQ,0,:PAC7,ELSE,:MIX0) ;important - hence - need
                                          ;to keep them busy
:REMIX0
  AO(%STOP,-,%DAY)
  JP(:MIX0)
:PAC1
  SV(@POSS,0)
  JP(:PAC)
:PAC0
  SV(@POSS,1)
  JP(:PAC)
:PAC2
```

```
  SV(@POSS,2)
  JP(:PAC)
:PAC
  SV(%END,CLOCK)
  IF(CLOCK,GE,%DAY,:REPAC)
:PPAC
  IF(%STOP,GT,%END,:REEND)
:CONPAC
  AO(%END,-,%STOP)
  AO(%SET,+,%END)
  PV(*PROD,%SET)
  PV(*DISPLAY,%SET)
  SV(@@CODED(@V,0),@A)
  SV(%%STOP(@V,0),%STOP)      ; Time that downtime begins
  SV(%%END(@V,0),%END)        ; Duration of downtime
  SV(@@DNCODE(@V,0),1)        ; Schedule to Pellet mill
  IV(@V)
  JP(:MIXPELL)
:REEND
  AO(%END,+,%DAY)
  JP(:CONPAC)
:REPAC
  AO(%END,-,%DAY)
  JP(:PPAC)
:PAC6
  SV(%END,CLOCK)
  IF(CLOCK,GE,%DAY,:REPAC1)
:PPAC1
  IF(%STOP,GT,%END,:REEND1)
:CONPAC1
  AO(%END,-,%STOP)
  AO(%SET,+,%END)
  PV(*PROD,%SET)
  PV(*DISPLAY,%SET)
  SV(@@CODED(@V,0),@A)
  SV(%%STOP(@V,0),%STOP)      ; Time that downtime begins
  SV(%%END(@V,0),%END)        ; Duration of downtime
  SV(@@DNCODE(@V,0),2)        ; Schedule to B17
  IV(@V)
  SV(@POSS3,6)
  JP(:B17A)
:REEND1
  AO(%END,+,%DAY)
  JP(:CONPAC1)
:REPAC1
  AO(%END,-,%DAY)
  JP(:PPAC1)
:PAC7
  SV(%END,CLOCK) ; Time lost is not saved when same code is mixed
  IF(CLOCK,GE,%DAY,:REPAC2)
:PPAC2
  IF(%STOP,GT,%END,:REEND2)
:CONPAC2
  AO(%END,-,%STOP)
  AO(%SET,+,%END)
```

```
  PV(*PROD,%SET)
  PV(*DISPLAY,%SET)
  SV(@@CODED(@V,0),@A)
  SV(%%STOP(@V,0),%STOP)      ; Time that downtime begins
  SV(%%END(@V,0),%END)        ; Duration of downtime
  SV(@@DNCODE(@V,0),3)        ; Schedule to B21
  IV(@V)
  SV(@POSS3,7)
  JP(:GO21A)
:REEND2
  AO(%END,+,%DAY)
  JP(:CONPAC2)
:REPAC2
  AO(%END,-,%DAY)
  JP(:PPAC2)
;=================================================================
;                     Wait at Pellet Mill #2
;=================================================================
:PM2WAIT_D                  ;Different product code & over Capacity
  PM(*ALERT1,WAIT)
:JUMP2_D
  WE
  IF(@@BIN(09,0),EQ,0,:B18B)
  IF(@@BIN(10,0),EQ,0,:B22B)
  IF(@@BIN(13,0),EQ,0,:B23B)
  IF(@@BIN(12,0),EQ,0,:B19B)
  IF(@@BIN(07,0),EQ,0,:B21B)
  IF(@@BIN(06,0),EQ,0,:B17B)
  IF(@@BIN(11,0),EQ,0,:B26B)
  IF(@@BIN(08,0),EQ,0,:B25B)
  IF(@@BIN(14,0),EQ,0,:B27B)
  IF(@@BIN(15,0),EQ,0,:B20B)
  IF(@@BIN(16,0),EQ,0,:B24B,ELSE,:JUMP2_D)

;=================================================================
:SUM
  SV(@@PRT(@CPR,0),@NOBATCH) ; Print no of batch mixed
  AO(@MASHF,+,@NOBATCH)
  SV(@@PK4(@COUNTPK4,@COUNTPK41),@PKK4)
  SV(@@PK2(@COUNTPK2,@COUNTPK21),@PKK2) ; Print no. of batch packed
  SV(@@PK3(@COUNTPK3,@COUNTPK31),@PKK3)
OF(MIX.RPT)
  PM(F,"                    SCHEDULE AT MIXER")
  PM(F," ")
  PM(F,"=============================================================")
  PM(F,"CODE       START TIME       ORDER   NO. BATCH     PELLET BIN")
  PM(F,"=============================================================")
  PM(F," ")
:LOO
  IV(@K)
  IV(@H)
    SV(@MIXC1,@@MIXC(@K,1))
    SV(@MIX1,@@MIX(@K,1))
  IF(@MIXC1,EQ,0,:LOO1)
    SV(@PCODE,@@CODE(@K,0))
```

```
    PV(F,@@QUAN(@PCODE,0),)
    PM(F,"  ",)
    SV(%TIME,%%PRT(@K,0))
    PV(F,%TIME,)
    PM(F,"   ",)
    SV(@PR,@@PRT(@H,0))
    PV(F,@PR,)
    PM(F,"   ",)
    PV(F,@MIXC1,)
    PM(F,"       ",)
    PV(F,@MIX1)
    SV(@MIXC1,@@MIXC(@K,2))
    SV(@MIX1,@@MIX(@K,2))
  IF(@MIXC1,EQ,0,:LOO1)
    PM(F,"                                    ",)
    PV(F,@MIXC1,)
    PM(F,"       ",)
    PV(F,@MIX1)
    SV(@MIXC1,@@MIXC(@K,3))
    SV(@MIX1,@@MIX(@K,3))
  IF(@MIXC1,EQ,0,:LOO1)
    PM(F,"                                    ",)
    PV(F,@MIXC1,)
    PM(F,"       ",)
    PV(F,@MIX1)
    SV(@MIXC1,@@MIXC(@K,4))
    SV(@MIX1,@@MIX(@K,4))
  IF(@MIXC1,EQ,0,:LOO1)
    PM(F,"                                    ",)
    PV(F,@MIXC1,)
    PM(F,"       ",)
    PV(F,@MIX1)
    SV(@MIXC1,@@MIXC(@K,5))
    SV(@MIX1,@@MIX(@K,5))
  IF(@MIXC1,EQ,0,:LOO1)
    PM(F,"                                    ",)
    PV(F,@MIXC1,)
    PM(F,"       ",)
    PV(F,@MIX1)
    SV(@MIXC1,@@MIXC(@K,6))
    SV(@MIX1,@@MIX(@K,6))
  IF(@MIXC1,EQ,0,:LOO1)
    PM(F,"                                    ",)
    PV(F,@MIXC1,)
    PM(F,"       ",)
    PV(F,@MIX1)
    SV(@MIXC1,@@MIXC(@K,7))
    SV(@MIX1,@@MIX(@K,7))
  IF(@MIXC1,EQ,0,:LOO1)
    PM(F,"                                    ",)
    PV(F,@MIXC1,)
    PM(F,"       ",)
    PV(F,@MIX1)
:LOO1
  PM(F," ")
```

```
  IF(@K,LT,@P,:LOO)
    IF(@P,EQ,0,:PRINTMIX)
      DV(@P)
:PRINTMIX
  PM(F,"============================================================")
  PM(F,"  FINISH TIME OF MIXER IS : ",)
  PV(F,%EMIX)
  PM(F,"  NO. OF CHANGES IN PRODUCT MIXED = ",)
  PV(F,@P)
  PM(F," ")
CF(RPT)
OF(PM1.RPT)
  PM(F,"               SCHEDULE AT PELLET MILL #1")
  PM(F," ")
  PM(F,"============================================================")
  PM(F,"CODE      START TIME    NO. BATCH              PACKING BIN  ")
  PM(F,"============================================================")
  PM(F," ")
  SV(@K,0)
  SV(@H,1)
:LOPM1
  IV(@K)
  IV(@H)
    SV(@PLC1,@@PM1C(@K,1))
    SV(@PLB1,@@PM1B(@K,1))
  IF(@PLC1,EQ,0,:LOPM11)
    SV(@PCODE,@@PMCODE(@K,0))
    PV(F,@@QUAN(@PCODE,0),)
    PM(F,"   ",)
    SV(%TIME,%%PM(@K,0))
    PV(F,%TIME,)
    SV(@PR,@@PM(@H,0))
    PV(F,@PR,)
    PM(F,"    ",)
    PV(F,@PLC1,)
    PM(F,"        ",)
    PV(F,@PLB1)
    SV(@PLC1,@@PM1C(@K,2))
    SV(@PLB1,@@PM1B(@K,2))
  IF(@PLC1,EQ,0,:LOPM11)
    PM(F,"                                    ",)
    PV(F,@PLC1,)
    PM(F,"        ",)
    PV(F,@PLB1)
    SV(@PLC1,@@PM1C(@K,3))
    SV(@PLB1,@@PM1B(@K,3))
  IF(@PLC1,EQ,0,:LOPM11)
    PM(F,"                                    ",)
    PV(F,@PLC1,)
    PM(F,"        ",)
    PV(F,@PLB1)
    SV(@PLC1,@@PM1C(@K,4))
    SV(@PLB1,@@PM1B(@K,4))
  IF(@PLC1,EQ,0,:LOPM11)
    PM(F,"                                    ",)
```

```
    PV(F,@PLC1,)
    PM(F,"       ",)
    PV(F,@PLB1)
:LOPM11
  PM(F," ")
  IF(@K,LT,@COUNTPM1,:LOPM1)
    IF(@COUNTPM1,EQ,0,:PRINTPM1)
      DV(@COUNTPM1)
:PRINTPM1
  PM(F,"============================================================")
  PM(F,"  FINISH TIME OF PELLET MILL #1 : ",)
  PV(F,%EPM1)
  PM(F,"  NO. OF CHANGES IN PRODUCT PELLET = ",)
  PV(F,@COUNTPM1)
  PM(F," ")
CF(RPT)
OF(PM2.RPT)
  PM(F,"               SCHEDULE AT PELLET MILL #2")
  PM(F," ")
  PM(F,"============================================================")
  PM(F,"CODE     START TIME    NO. BATCH              PACKING BIN  ")
  PM(F,"============================================================")
  PM(F," ")
  SV(@K,0)
  SV(@H,1)
:LOPM2
  IV(@K)
  IV(@H)
    SV(@PLC21,@@PM2C(@K,1))
    SV(@PLB21,@@PM2B(@K,1))
  IF(@PLC21,EQ,0,:LOPM21)
    SV(@PCODE,@@PMCODE(@K,1))
    PV(F,@@QUAN(@PCODE,0),)
    PM(F,"   ",)
    SV(%TIME,%%PM(@K,1))
    PV(F,%TIME,)
    SV(@PR,@@PM(@H,1))
    PV(F,@PR,)
    PM(F,"    ",)
    PV(F,@PLC21,)
    PM(F,"       ",)
    PV(F,@PLB21)
    SV(@PLC21,@@PM2C(@K,2))
    SV(@PLB21,@@PM2B(@K,2))
  IF(@PLC21,EQ,0,:LOPM21)
    PM(F,"                                    ",)
    PV(F,@PLC21,)
    PM(F,"       ",)
    PV(F,@PLB21)
    SV(@PLC21,@@PM2C(@K,3))
    SV(@PLB21,@@PM2B(@K,3))
  IF(@PLC21,EQ,0,:LOPM21)
    PM(F,"                                    ",)
    PV(F,@PLC21,)
    PM(F,"       ",)
```

```
    PV(F,@PLB21)
    SV(@PLC21,@@PM2C(@K,4))
    SV(@PLB21,@@PM2B(@K,4))
  IF(@PLC21,EQ,0,:LOPM21)
    PM(F,"                                    ",)
    PV(F,@PLC21,)
    PM(F,"       ",)
    PV(F,@PLB21)
:LOPM21
  PM(F," ")
  IF(@K,LT,@COUNTPM2,:LOPM2)
    IF(@COUNTPM2,EQ,0,:PRINTPM2)
      DV(@COUNTPM2)
:PRINTPM2
  PM(F,"============================================================")
  PM(F,"  FINISH TIME OF PELLET MILL #2 : ",)
  PV(F,%EPM2)
  PM(F,"  NO. OF CHANGES IN PRODUCT PELLET = ",)
  PV(F,@COUNTPM2)
  PM(F," ")
CF(RPT)
OF(PK2.RPT)
  PM(F,"                          VARIATION AT PK2")
  PM(F," ")
  PM(F,"=============================================================================='
  PM(F,"PRODUCT CODE   START TIME    BATCHS  FROM BIN  PACK SIZE  NO.OF BAG"
  PM(F,"=============================================================================='
  SV(@K,0)
  SV(@H,1)
:LOPK2
  IV(@K)
  IV(@H)
  SV(@PK2N1,@@PK2(@K,1))
  SV(@PK2B1,@@PKK2(@K,1))
  IF(@PK2N1,EQ,0,:LOPK21)
    PM(F," ")
    PM(F,"    ",)
    SV(@PCODE,@@PKCODE(@K,1))
    PV(F,@@QUAN(@PCODE,0),)
    PM(F,"     ",)
    SV(%TIME,%%PK(@K,1))
    PV(F,%TIME,)
    PM(F," ",)
    PV(F,@PK2N1,)
    PM(F,"   ",)
    LK(!BIN2)
    PV(F,@PK2B1,)
    PM(F,"   ",)
    SV(@OBJ2,@@QUAN(@PCODE,2))
    SV(@OBJ3,@@QUAN(@PCODE,3))
    PV(F,@OBJ2,)
    PM(F,"     ",)
    SV(@PRBAG,@PK2N1)
    AO(@PRBAG,*,@OBJ3)
    AO(@PRBAG,*,10)
```

```
    AO(@OBJ2,/,10)
    AO(@PRBAG,/,@OBJ2)
    PV(F,@PRBAG)
    SV(@PK2N1,@@PK2(@K,2))
    SV(@PK2B1,@@PKK2(@K,2))
    IF(@PK2N1,EQ,0,:LOPK21)
      PM(F,"                                    ",)
      PV(F,@PK2N1,)
      PM(F,"  ",)
      LK(!BIN2)
      PV(F,@PK2B1,)
      PM(F,"  ",)
      SV(@OBJ2,@@QUAN(@PCODE,2))
      SV(@OBJ3,@@QUAN(@PCODE,3))
      PV(F,@OBJ2,)
      PM(F,"    ",)
      SV(@PRBAG,@PK2N1)
      AO(@PRBAG,*,@OBJ3)
      AO(@PRBAG,*,10)
      AO(@OBJ2,/,10)
      AO(@PRBAG,/,@OBJ2)
      PV(F,@PRBAG)
      SV(@PK2N1,@@PK2(@K,3))
      SV(@PK2B1,@@PKK2(@K,3))
      IF(@PK2N1,EQ,0,:LOPK21)
        PM(F,"                                    ",)
        PV(F,@PK2N1,)
        PM(F,"  ",)
        LK(!BIN2)
        PV(F,@PK2B1,)
        PM(F,"  ",)
        SV(@OBJ2,@@QUAN(@PCODE,2))
        SV(@OBJ3,@@QUAN(@PCODE,3))
        PV(F,@OBJ2,)
        PM(F,"    ",)
        SV(@PRBAG,@PK2N1)
        AO(@PRBAG,*,@OBJ3)
        AO(@PRBAG,*,10)
        AO(@OBJ2,/,10)
        AO(@PRBAG,/,@OBJ2)
        PV(F,@PRBAG)
:LOPK21
  IF(@K,LT,@COUNTPK2,:LOPK2)
    IF(@COUNTPK2,EQ,0,:PRINTPK2)
      DV(@COUNTPK2)
:PRINTPK2
  PM(F," ")
  PM(F,"=============================================================================")
  PM(F,"  FINISH TIME OF PACKING HEAD #2 : ",)
  PV(F,%EPK2)
  PM(F,"  NO. OF CHANGES IN PRODUCT PACKED = ",)
  PV(F,@COUNTPK2)
  PM(F," ")
CF(RPT)
OF(PK3.RPT)
```

```
  PM(F,"                          VARIATION AT PK3")
  PM(F," ")
  PM(F,"=============================================================================")
  PM(F,"PRODUCT CODE   START TIME    BATCHS  FROM BIN  PACK SIZE  NO.OF BAG")
  PM(F,"=============================================================================")
  SV(@K,0)
  SV(@H,1)
:LOPK3
  IV(@K)
  IV(@H)
  SV(@PK3N1,@@PK3(@K,1))
  SV(@PK3B1,@@PKK3(@K,1))
  IF(@PK3N1,EQ,0,:LOPK31)
    PM(F," ")
    PM(F,"    ",)
    SV(@PCODE,@@PKCODE(@K,2))
    PV(F,@@QUAN(@PCODE,0),)
    PM(F,"     ",)
    SV(%TIME,%%PK(@K,2))
    PV(F,%TIME,)
    PM(F," ",)
    PV(F,@PK3N1,)
    PM(F,"  ",)
    LK(!BIN3)
    PV(F,@PK3B1,)
    PM(F,"  ",)
    SV(@OBJ2,@@QUAN(@PCODE,2))
    SV(@OBJ3,@@QUAN(@PCODE,3))
    PV(F,@OBJ2,)
    PM(F,"     ",)
    SV(@PRBAG,@PK3N1)
    AO(@PRBAG,*,@OBJ3)
    AO(@PRBAG,*,10)
    AO(@OBJ2,/,10)
    AO(@PRBAG,/,@OBJ2)
    PV(F,@PRBAG)
    SV(@PK3N1,@@PK3(@K,2))
    SV(@PK3B1,@@PKK3(@K,2))
    IF(@PK3N1,EQ,0,:LOPK31)
      PM(F,"                                  ",)
      PV(F,@PK3N1,)
      PM(F,"  ",)
      LK(!BIN3)
      PV(F,@PK3B1,)
      PM(F,"  ",)
      SV(@OBJ2,@@QUAN(@PCODE,2))
      SV(@OBJ3,@@QUAN(@PCODE,3))
      PV(F,@OBJ2,)
      PM(F,"     ",)
      SV(@PRBAG,@PK3N1)
      AO(@PRBAG,*,@OBJ3)
      AO(@PRBAG,*,10)
      AO(@OBJ2,/,10)
      AO(@PRBAG,/,@OBJ2)
      PV(F,@PRBAG)
```

```
      SV(@PK3N1,@@PK3(@K,3))
      SV(@PK3B1,@@PKK3(@K,3))
      IF(@PK3N1,EQ,0,:LOPK31)
        PM(F,"                                    ",)
        PV(F,@PK3N1,)
        PM(F,"  ",)
        LK(!BIN3)
        PV(F,@PK3B1,)
        PM(F,"  ",)
        SV(@OBJ2,@@QUAN(@PCODE,2))
        SV(@OBJ3,@@QUAN(@PCODE,3))
        PV(F,@OBJ2,)
        PM(F,"     ",)
        SV(@PRBAG,@PK3N1)
        AO(@PRBAG,*,@OBJ3)
        AO(@PRBAG,*,10)
        AO(@OBJ2,/,10)
        AO(@PRBAG,/,@OBJ2)
        PV(F,@PRBAG)
:LOPK31
  IF(@K,LT,@COUNTPK3,:LOPK3)
    IF(@COUNTPK3,EQ,0,:PRINTPK3)
      DV(@COUNTPK3)
:PRINTPK3
  PM(F," ")
  PM(F,"=============================================================================")
  PM(F,"  ")
  PM(F,"  FINISH TIME OF PACKING HEAD #3 : ",)
  PV(F,%EPK3)
  PM(F," ")
  PM(F,"  NO. OF CHANGES IN PRODUCT PACKED = ",)
  PV(F,@COUNTPK3)
CF(RPT)
OF(PK4.RPT)
  PM(F,"                          VARIATION AT PK4")
  PM(F,"=============================================================================")
  PM(F,"PRODUCT CODE   START TIME    BATCHS  FROM BIN  PACK SIZE  NO.OF BAG")
  PM(F,"=============================================================================")
  SV(@K,0)
  SV(@H,1)
:LOPK4
  IV(@K)
  IV(@H)
  SV(@PK4N1,@@PK4(@K,1))
  SV(@PK4B1,@@PKK4(@K,1))
  IF(@PK4N1,EQ,0,:LOPK41)
    PM(F," ")
    PM(F,"    ",)
    SV(@PCODE,@@PKCODE(@K,0))
    PV(F,@@QUAN(@PCODE,0),)
    PM(F,"     ",)
    SV(%TIME,%%PK(@K,0))
    PV(F,%TIME,)
    PM(F," ",)
    PV(F,@PK4N1,)
```

```
    PM(F,"  ",)
    LK(!BIN4)
    PV(F,@PK4B1,)
    PM(F,"  ",)
    SV(@OBJ2,@@QUAN(@PCODE,2))
    SV(@OBJ3,@@QUAN(@PCODE,3))
    PV(F,@OBJ2,)
    PM(F,"    ",)
    SV(@PRBAG,@PK4N1)
    AO(@PRBAG,*,@OBJ3)
    AO(@PRBAG,*,10)
    AO(@OBJ2,/,10)
    AO(@PRBAG,/,@OBJ2)
    PV(F,@PRBAG)
    SV(@PK4N1,@@PK4(@K,2))
    SV(@PK4B1,@@PKK4(@K,2))
    IF(@PK4N1,EQ,0,:LOPK41)
      PM(F,"                                ",)
      PV(F,@PK4N1,)
      PM(F,"  ",)
      LK(!BIN4)
      PV(F,@PK4B1,)
      PM(F,"  ",)
      SV(@OBJ2,@@QUAN(@PCODE,2))
      SV(@OBJ3,@@QUAN(@PCODE,3))
      PV(F,@OBJ2,)
      PM(F,"    ",)
      SV(@PRBAG,@PK4N1)
      AO(@PRBAG,*,@OBJ3)
      AO(@PRBAG,*,10)
      AO(@OBJ2,/,10)
      AO(@PRBAG,/,@OBJ2)
      PV(F,@PRBAG)
      SV(@PK4N1,@@PK4(@K,3))
      SV(@PK4B1,@@PKK4(@K,3))
      IF(@PK4N1,EQ,0,:LOPK41)
        PM(F,"                              ",)
        PV(F,@PK4N1,)
        PM(F,"  ",)
        LK(!BIN4)
        PV(F,@PK4B1,)
        PM(F,"  ",)
        SV(@OBJ2,@@QUAN(@PCODE,2))
        SV(@OBJ3,@@QUAN(@PCODE,3))
        PV(F,@OBJ2,)
        PM(F,"    ",)
        SV(@PRBAG,@PK4N1)
        AO(@PRBAG,*,@OBJ3)
        AO(@PRBAG,*,10)
        AO(@OBJ2,/,10)
        AO(@PRBAG,/,@OBJ2)
        PV(F,@PRBAG)
:LOPK41
  IF(@K,LT,@COUNTPK4,:LOPK4)
    IF(@COUNTPK4,EQ,0,:PRINTPK4)
```

```
      DV(@COUNTPK4)
:PRINTPK4
  PM(F," ")
  PM(F,"=========================================================================="
  PM(F,"  FINISH TIME OF PACKING HEAD #4 : ",)
  PV(F,%EPK4)
  PM(F,"  NO. OF CHANGES IN PRODUCT PACKED = ",)
  PV(F,@COUNTPK4)
CF(RPT)
OF(SILO.RPT)
  PM(F,"                VARIATION OF SILO")
  PM(F," ")
  PM(F,"============================================")
  PM(F,"PRODUCT CODE  START TIME    BATCHS  FROM BIN")
  PM(F,"============================================")
  SV(@K,0)
  SV(@H,1)
:LOSILO
  IV(@K)
  IV(@H)
  SV(@PKSN1,@@PKS(@K,1))
  SV(@PKSB1,@@PKKS(@K,1))
  IF(@PKSN1,EQ,0,:LOSILO1)
    PM(F," ")
    PM(F,"    ",)
    SV(@PCODE,@@PKCODE(@K,3))
    PV(F,@@QUAN(@PCODE,0),)
    PM(F,"    ",)
    SV(%TIME,%%PK(@K,3))
    PV(F,%TIME,)
    PM(F," ",)
    PV(F,@PKSN1,)
    PM(F,"   ",)
    LK(!BINS)
    PV(F,@PKSB1)
    SV(@PKSN1,@@PKS(@K,2))
    SV(@PKSB1,@@PKKS(@K,2))
    IF(@PKSN1,EQ,0,:LOSILO1)
      PM(F,"                                  ",)
      PV(F,@PKSN1,)
      PM(F,"   ",)
      LK(!BINS)
      PV(F,@PKSB1)
      SV(@PKSN1,@@PKS(@K,3))
      SV(@PKSB1,@@PKKS(@K,3))
      IF(@PKSN1,EQ,0,:LOSILO1)
        PM(F,"                                  ",)
        PV(F,@PKSN1,)
        PM(F,"   ",)
        LK(!BINS)
        PV(F,@PKSB1)
:LOSILO1
  IF(@K,LT,@COUNTS,:LOSILO)
    IF(@COUNTS,EQ,0,:PRINTS)
      DV(@COUNTS)
```

```
:PRINTS
  PM(F," ")
  PM(F,"=============================================")
  PM(F,"  NO. OF CHANGES IN PRODUCT SILO = ",)
  PV(F,@COUNTS)
CF(RPT)
OF(PROD.RPT)
:DPAC
  PM(F," ")
  PM(F,"============================================================")
  PM(F,"                          NO.BATCH     TOTAL TONNAGE  ")
  PM(F,"============================================================")
  PM(F," ")
  PM(F,"PELLET FEED MIXED       =  ",)
  PV(F,@PELLF,)
  PM(F,"     ",)
  AO(@PELLF,*,3)
  PV(F,@PELLF,)
  PM(F,"  (24 HOURS)")
  PM(F,"MASH FEED MIXED         =  ",)
  PV(F,@MASHF,)
  PM(F,"     ",)
  AO(@MASHF,*,2)
  PV(F,@MASHF,)
  PM(F,"  (24 HOURS)")
  PM(F,"SILO LOADING            =  ",)
  PV(F,@SILO)
  PM(F," ")
  PM(F,"TOTAL DAILY PRODUCTION =  ",)
  PV(F,@TPACK,)
  AO(@TPACK,*,3)
  PM(F,"     ",)
  PV(F,@TPACK)
  PM(F,"  ")
  PM(F,"============================================================")
CF(RPT)
OF(DOWN.RPT)
  PM(F," ")
  PM(F,"================================================================")
  PM(F,"                            DOWNTIME AT MIXER")
  PM(F,"================================================================")
  PM(F," PRODUCT CODE  DOWNTIME START     DURATION    SOLVE SCHEDULED")
:LOOPL
  SV(@PCODE,@@CODED(@L,0))
  PM(F,"      ",)
  PV(F,@@QUAN(@PCODE,0),)
  SV(%STO,%%STOP(@L,0))
  SV(%END1,%%END(@L,0))
  SV(@DNCODE,@@DNCODE(@L,0))
  PM(F,"    ",)
  PV(F,%STO,)
  PM(F,"  ",)
  PV(F,%END1,)
  PM(F,"     ",)
  PV(F,@DNCODE)
```

```
  IV(@L)
  IF(@L,LT,@V,:LOOPL)
    PM(F,"  ")
    PM(F,"   TOTAL DOWNTIME AT MIXER =  ",)
    PV(F,%SET)
    PM(F,"============================================================")
CF(RPT)
:LOOP
  GS(@TONE,@DURATION)
  IV(@TONE)
  IV(@DURATION)
  IF(@TONE,LE,200,:LOOP)
:LOOP1
  GS(@TONE,@DURATION)
  AO(@TONE,-,3)
  DV(@DURATION)
  IF(@TONE,GT,150,:LOOP1)
    GS
    WK(I)
  JP(:ENDROUT)
:TOTALP
  IV(@TPACK)
  SV(%CLOCK,CLOCK)
  IF(%CLOCK,GE,%DAY1,:SUM)
    PV(*PT,@TPACK)
:ENDROUT
ER
;================================================================
BR(6,XY(7,75),0)              ; PK4 downtime
  WT(12:0:0)                  ; Lunch break
  SV(@@CELL(3,1),1)
  PM(*DSP4,Meal)
  GS
  WT(1:00:0)                  ; duration
  SV(@@CELL(3,1),0)
  PM(*DSP4,     )
  WT(8:0:0)                   ; Night break
  SV(@@CELL(3,1),1)
  PM(*DSP4,Meal)
  GS
  WT(1:00:0)                  ; duration
  SV(@@CELL(3,1),0)
  PM(*DSP4,     )
ER
BR(7,XY(22,75),0)             ; PK2 downtime
  WT(12:0:0)                  ; Lunch break
  SV(@@CELL(4,1),1)
  PM(*DSP2,Meal)
  GS
  WT(1:00:0)                  ; duration
  SV(@@CELL(4,1),0)
  PM(*DSP2,     )
  WT(8:0:0)                   ; Night break
  SV(@@CELL(4,1),1)
  PM(*DSP2,Meal)
```

```
  GS
  WT(1:00:0)                    ; duration
  SV(@@CELL(4,1),0)
  PM(*DSP2,     )
ER
BR(8,XY(35,75),0)               ; PK3 downtime
  WT(12:0:0)                    ; Lunch break
  SV(@@CELL(5,1),1)
  PM(*DSP3,Meal)
  GS
  WT(1:00:0)                    ; duration
  SV(@@CELL(5,1),0)
  PM(*DSP3,     )
  WT(8:0:0)                     ; Night break
  SV(@@CELL(5,1),1)
  PM(*DSP3,Meal)
  GS
  WT(1:00:0)                    ; duration
  SV(@@CELL(5,1),0)
  PM(*DSP3,     )
ER
```

ภาคผนวก ฌ

ตารางการผลิตที่ได้จากแบบจำลอง

SCHEDULE AT MIXER

CODE	START TIME	ORDER	NO. BATCH	PELLET BIN
00510	0008:05:01.00	00008	00008	00016
00511	0008:46:29.00	00013	00007	00015
			00004	00001
			00002	00004
00551	0009:54:42.00	00020	00004	00003
			00004	00002
			00012	00016
00552	0011:59:54.00	00023	00009	00015
			00005	00001
			00005	00004
			00004	00003
00531	0013:56:39.00	00016	00004	00002
			00011	00016
			00001	00015
00534	0015:38:46.00	00021	00003	00001
			00004	00004
			00004	00003
			00004	00002
			00006	00016
00522	0017:49:40.00	00016	00012	00015
			00004	00001
00523	0019:05:08.00	00015	00007	00004
			00007	00003
			00001	00002
00525	0020:15:53.00	00009	00009	00016
00555	0021:03:12.00	00018	00010	00015
			00005	00001
			00003	00004
00566	0022:33:29.00	00024	00007	00003
			00007	00002
			00010	00016
00567	0000:31:14.00	00030	00012	00015
			00007	00001
			00007	00004
			00004	00003
00151	0002:52:44.00	00008	00008	00021

CODE	START TIME	ORDER	NO. BATCH	PELLET BIN
00326	0003:49:39.00	00010	00010	00017
00152	0004:36:49.00	00012	00012	00017
00153	0005:37:11.00	00005	00005	00021

FINISH TIME OF MIXER IS : 0006:13:47.00

NO. OF CHANGES IN PRODUCT MIXED = 00015

SCHEDULE AT PELLET MILL #1

CODE	START TIME	NO. BATCH		PACKING BIN
00510	0008:05:41.00	00008	00008	00022
00511	0009:28:49.00	00007	00007	00027
00551	0010:49:21.00	00012	00008	00018
			00004	00023
00552	0013:04:21.00	00009	00008	00019
			00001	00021
00531	0014:40:12.00	00012	00006	00017
			00003	00026
			00003	00025
00534	0017:15:32.00	00006	00006	00022
00522	0018:33:14.00	00012	00012	00019
00525	0020:21:06.00	00009	00009	00027
00555	0021:37:18.00	00010	00010	00019
00566	0023:44:44.00	00010	00008	00021
			00002	00017
00567	0001:07:34.00	00012	00004	00026
			00003	00025
			00005	00022

FINISH TIME OF PELLET MILL #1 : 0002:43:59.00

NO. OF CHANGES IN PRODUCT PELLET = 00010

SCHEDULE AT PELLET MILL #2

CODE	START TIME	NO. BATCH		PACKING BIN
00511	0009:23:32.00	00006	00004	00027
			00002	00020
00551	0010:35:20.00	00008	00008	00022
00552	0012:45:58.00	00014	00006	00019
			00008	00021
00531	0015:11:44.00	00004	00003	00017
			00001	00026
00534	0016:11:20.00	00015	00008	00018
			00007	00023
00522	0019:00:34.00	00004	00004	00023
00523	0019:47:18.00	00015	00013	00018
			00002	00021
00555	0021:54:18.00	00008	00008	00021
00566	0023:26:10.00	00014	00013	00021
			00001	00017
00567	0001:28:36.00	00018	00001	00026
			00002	00025
			00015	00018

FINISH TIME OF PELLET MILL #2 : 0003:48:55.00

NO. OF CHANGES IN PRODUCT PELLET = 00009

VARIATION AT PK2

PRODUCT CODE	START TIME	BATCHS	FROM BIN	PACK SIZE	NO.OF BAG
00510	0008:30:06.00	00008	00022	00030	00800
00551	0014:09:10.00	00008	00022	00030	00800
		00008	00018	00030	00800
00534	0020:27:05.00	00001	00022	00030	00100
		00008	00018	00030	00800
		00005	00022	00030	00500
00523	0022:34:26.00	00013	00018	00030	00866
00567	0005:31:29.00	00001	00022	00030	00066
		00015	00018	00030	01000
		00004	00022	00030	00266

FINISH TIME OF PACKING HEAD #2 : 0006:22:25.00

NO. OF CHANGES IN PRODUCT PACKED = 00004

VARIATION AT PK3

PRODUCT CODE	START TIME	BATCHS	FROM BIN	PACK SIZE	NO.OF BAG
00551	0013:00:18.00	00004	00023	00030	00400
00552	0013:54:54.00	00014	00019	00030	01166
00531	0016:24:34.00	00004	00026	00030	00400
00534	0018:02:06.00	00007	00023	00030	00700
00522	0022:08:16.00	00012	00019	00030	00800
		00004	00023	00030	00266
00555	0022:46:20.00	00010	00019	00030	00833

FINISH TIME OF PACKING HEAD #3 : 0000:34:40.00

NO. OF CHANGES IN PRODUCT PACKED = 00005

VARIATION AT PK4

PRODUCT CODE	START TIME	BATCHS	FROM BIN	PACK SIZE	NO.OF BAG
00552	0014:09:57.00	00009	00021	00030	00750
00531	0018:09:00.00	00009	00017	00030	00900
		00003	00025	00030	00300
00523	0022:00:00.00	00002	00021	00030	00133
00555	0022:24:23.00	00008	00021	00030	00666
00566	0003:21:10.00	00021	00021	00030	01400
		00003	00017	00030	00200
00567	0003:53:55.00	00005	00025	00030	00333
00151	0004:45:10.00	00008	00021	00030	00533
00152	0006:04:10.00	00012	00017	00030	00800

FINISH TIME OF PACKING HEAD #4 : 0008:00:10.00

NO. OF CHANGES IN PRODUCT PACKED = 00007

VARIATION OF SILO

PRODUCT CODE	START TIME	BATCHS	FROM BIN
00511	0011:02:57.00	00010	00027
		00002	00020
		00001	00027
00525	0020:43:35.00	00009	00027
00326	0003:50:35.00	00010	00017

NO. OF CHANGES IN PRODUCT SILO = 00002

	NO.BATCH	TOTAL TONNAGE	
PELLET FEED MIXED =	00213	00639	(24 HOURS)
MASH FEED MIXED =	00035	00070	(24 HOURS)
SILO LOADING =	00031		
TOTAL DAILY PRODUCTION =	00206	00618	

DOWNTIME AT MIXER

PRODUCT CODE	DOWNTIME START	DURATION	SOLVE SCHEDULED
00151	0002:52:44.00	0000:19:11.00	00003
00326	0003:49:39.00	0000:00:00.00	00002
00152	0004:36:49.00	0000:03:46.00	00002
00153	0005:37:11.00	0000:17:44.00	00003

TOTAL DOWNTIME AT MIXER = 0000:40:41.00

ประวัติผู้เขียน

นางสาวนภิสพร คีนตัก เกิดวันที่ 1 พฤษภาคม พ.ศ.2506 ที่จังหวัดตรัง สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต (สาขาวิศวกรรมอุตสาหการ) จากคณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ ในปีการศึกษา 2527 และเข้าศึกษาต่อในระดับปริญญามหาบัณฑิต ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2531 ปัจจุบันทำงานรับราชการเป็นอาจารย์ประจำภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่