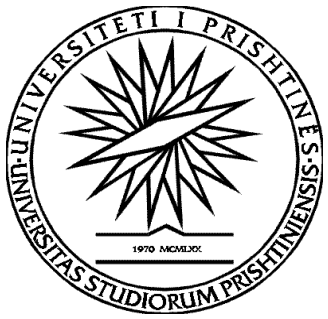
UNIVERSITETI I PRISHTINËS
FAKULTETI I INXHINIERISË MEKANIKE
PRISHTINË



Mr. Sc. Mirlind Bruçi , inxhi. i dipl.

Optimalizimi i proceseve prodhuese në linjat fleksibile asembluese të reparteve të elektromotorëve në fabrikën ATB-Wiena

-PUNIM I DOKTORATURËS-

Prishtinë, 2005

Përmbajtja

HYRJE

Në kohën e sotme, moderne, konsumatorët e kanë ndryshuar idenë e blerjes. Ata sot dëshirojnë një produkt, i cili do të kryente funksionin e vet për aq sa ka nevojë ai vetë dhe të kushtojë sa më lirë. Nga ana e tjetër, konsumatori dëshiron që në treg të ketë shumëllojshmëri të produkteve në pikëpamje të gabariteve si dhe të jenë universale nga aspekti i shfrytëzimit. Kërkesat njerëzore janë të ndryshme, prandaj inxhinierëve ju parashtrohen detyra të ndryshme për projektimin e sistemeve fleksibile të përpunimit dhe të montimit.

Sistemet fleksibile të montimit bëjnë pjesë në grupin e sistemeve më të sofistikuara të kohës. Si të tilla ato kanë mundësi që për kohë sa më të shkurtë të kalojnë prej një programi të montimit në tjetrin. D.m.th. për kohë sa më të shkurtë mund të adaptohen për programe të reja, çka nuk mund ta bëjnë automatet dhe linjat automatike të ndryshme. Për këtë arsye edhe quhen sisteme fleksibile. Fleksibiliteti si veti qenësore e këtyre sistemeve erdhi në shprehje duke i falënderuar zbulimit të robotëve dhe kompjuterit. Në këtë mënyrë për të kaluar nga një sistem i montimit në tjetrin, mjafton që të ndërrohen në mënyrë automatike, gjysmë automatike ose manuale pajisjet ndihmese të robotëve dhe makinave.

Mirëpo, përveç anëve pozitive, këto sisteme kanë edhe anët e tyre negative-menaxhimin e sistemit. Me ndihmën e këtyre sistemeve shumë shpesh realizohet montimi i llojeve të ndryshme të produkteve. Pra nga ana e menaxhimit duhet më herët të kryhen të gjitha përgatitjet për prodhim. Këto përgatitje përfshijnë detyrat e zgjedhjes së një skenari optimal respektivisht suboptimal të rrjedhjes së materialit, energjisë, informatës dhe organizimit të fuqisë punëtore. Zgjedhja e skenarit do të thotë se nga një numër shumë i madh i skenarëve të ndryshëm, për shkak të llojllojshmërisë së tipave të produkteve që marrin pjesë në një program të montimit, të zgjidhet ai më optimali. Zgjedhja e këtyre skenarëve bëhet nëpërmjet algoritmeve të ndryshëm për optimalizim, të cilët janë të integruar në software, d.m.th. duhet të zgjidhet dhe programohet një modul.

Pasi ky modul ndryshon prej një sistemi në një sistem tjetër, atëherë ky modul duhet të ndërtohet për një sistem specifik. Në rastin tonë, në repartin e montimit të

elektromotorëve në fabrikën “ATB-Wiena-Austria”, duhet të përpunohet një modul i tillë, i cili duhet ta kryej funksionin e montimit. Ky modul do të duhej të vendoset në kompjuterin qendror të repartit, i cili i koordinon të gjitha aktivitete On-Line në parimin shkak-pasojë.

Në repartin e elektromotorëve ne fabrikën ATB-Wiena janë parashtruar katër detyra kryesore :

1. Rritja e vëllimit të prodhimit, respektivisht montimit, në krahasim me sistemin paraprak
2. Rritja e kualitetit të prodhimit
3. Shfrytëzimi më racional i makinave dhe pajisjeve tjera duke përfshirë edhe fuqinë njerëzore
4. Përgatitja e makinave dhe e Software-ve duhet të jenë ashtu të organizuara që në kohën sa më të shkurtër të kalojnë nga një program i prodhimit në tjetrin.

Një kërkesë e cila nuk është cekur këtu, por që ka rëndësi të madhe, është metoda e zgjidhjes së detyrave e cila është e rastit. Këtu është me rëndësi se ajo nuk do ta pengojë procesin e prodhimit, pra duhet të jetë Off-Line. Pra nga kërkesat e parashtruara shihet qartë se kjo përbënë një detyrë shumë të rëndë. Deri para më pak se dy decenie, kjo detyrë ka qenë detyrë ekipore. Sot kjo është e mundshme vetëm nga ndonjë individ duke përdorë software përkatëse. Në rastin tonë do të përdoret programi i ashtuquajtur ARENA-Rockwell Automation.

Në mënyrë që të vërtetojmë tezën e parashtruar, skenarët e ndryshëm do të simulohen me Arenë. Për kushtet e parashtruara do të gjendet skenari më i përshtatshëm duke e aplikuar veglën OptQuest for Arena. Ky skenar apo model i optimalizimit paraqet modulin i cili do të integrohet në programin e kompjuterit qendrorë -Host Computer. Në bazë të eksperimenteve të ndryshme nxirren përfundimet e hulumtimit. Këto përfundime do të paraqiten nëpërmjet përkufizimeve, tabelave dhe diagrameve përkatëse.

Literatura

[1] A. Bunjaku. Automatizimi i Prodhimtarisë, ligjërata të autorizuara, Prishtinë 2005

[2] A. Bunjaku. Përpunimi me prerje I**,** ligjërata të autorizuara, Prishtinë 2005

[3] A. Bunjaku. Matja dhe kontrolli, ligjërata të autorizuara, Prishtinë 2005

[4] I. Bordoniqi, A. Bunjaku, M. Bruçi. Hulumtimi mundësisë për automatizimin e disa elementeve të procesit projektues dhe konstruktues të teknologjisë së farkëtimit, Makineria-Revistë shkencore për teori dhe praktikë në makineri, Viti I, Nr.1, Prishtinë, 1997, fq.55-60

[5] M. Bruçi, I. Bordoniqi, A. Bunjaku. Sistemi programues për llogaritjen automatike te disa parametrave optimal të procesit të farkëtimit, Makineria-Revistë shkencore për teori dhe praktikë në makineri, Viti II, Nr.2, Prishtinë, 1998, fq.43-48

[6] M. Bruçi, M. Stopper, A. Bunjaku, K. Stuja, A. Kubat. Optimisation of Flexible Assembly System for Electrical Motors using Optquest for Arena, Published by DAAAM International pp. 071-072, Vienna, Austria 2003

[7] M. Bruçi, A. Bunjaku, K. Stuja, Sh. Buza. Optimizing of Flexible Palletizing Lines Usning Simulation Tools, TMT 2005, Antalya, Turkey, 2005

[8] K. Stuja, M.Stopper, A. Bunjaku, M. Bruçi. A concept for scheduling of flexible assembly for elecrical motors, Chapter 51, DAAAM International Scientific Book 2003, B. Katalinic (ED) pp. 591-598, ISBN 3-901509-36-4, ISSN 1726-9687, Vienna, Austria

[9] Sh. Buza, K. Buza, M. Bruçi. Optimizimi i statorit **t**ë elektromotorit, Makineria-Revistë shkencore për teori dhe praktikë në makineri, Viti IV, Nr.1, Prishtinë, 2202, fq.119-124

[10] Sh. Buza. CAD/CAM sistemet, ligjerata të autorizuara, Prishtinë 2005

[11] Sh. Buza. Teoria dhe metodat e konstruktimit**,** ligjërata të autorizuara, Prishtinë 2005

[12] ARENA Variables Guide, System Modeling Corporation. The Park Building 504, Beavier Street, Sewickley, USA, 2002

Park Building 504, Beavier Street, Sewickley, USA, 1994

[13] Eric Klavins, Toward the Control of Self-Assembling Systems, California Institute Of Technology, CA 91125, USA, 2002

[14] David Carney, Assembling Large Systems from COTS Components: Opportunities, cautions,and Complexities, Pittsburgh, PA 15213-3890,1997

[**15] J. Kosturiak, M.Gregor**. FMS Simulation: Some experience and recommendations, Simulation Practice and Theory 6 pp. 423-442, Elsevier, 1998

[16] K. Boonlertvanich, Extended-Cowip-Kanban System: Control and Performance Analysis, Georgia Institute of Technology, April 2005

[17] Becker, B. D.; Schulte, J. W. (1994). Simulation-based Integrated Planning and Control Systems for Manufacturing, Proceedings of the WORKSIMS'94, ISBN 974-8256-154, pp 34-38, Bangkok, Thailand, Nov. 9-11th 1994

[18] J. Balic. Contribution to Integrated Manufacturing. Editor B. Katalinic, pp. 73-110, ISBN 3-901509-03-8, 1996

[19] Centeno, M. A. (1996) An Introduction to Simulation Modeling, Proceedings of the 1996 Winter Simulation Conference, ISBN 0-7803-3383-7, pp 15-22, Coronado, California, December 8-11, 1996

[20] Chan, F. T. S.; Smith, A. M. Simulation Approach to Assembly Line Modification: A Case Study. Journal of Manufacturing Systems, Vol. 12/No. 3, Springer Verlag, Berlin Heidelberg, 1992

[21] Fishwick P. A. Simulation Model Design And Execution, Building Digital Worlds, Prentice Hall, New Jersey, 1995

[22] Katalinic B., Design Methodology of Scheduling Strategies and Scenarios of Complex Flexible Systems, Proceedings of the 29th CIRP International Seminar on Manufacturing Systems, pp 179-184, Osaka University, Japan, May 11-13, 1997

[23] Katalinic, B. & Celar, S. Discrete-event Based Computer Simulation of Complex Flexible Manufacturing Systems, Proceedings of the 13th International Conference BIAM'96, ISBN 953-6037-12-2, pp J-1 J-4, June18-21 st, 1996, Zagreb, Croatia

[24] Stuja,K.; Katalinic, B.; Ljolic, B. & Kordic,V. Scheduling of Flexibile Manufacturing Systems for Rotational Parts in a Case of Limited Production Means, Chapter 57, DAAAM International Scientific Book 2002, B. Katalinic (ED), pp 555-564, Published by DAAAM International, ISBN: 3-901509-30-5, Viena, Austria

[25] Katalinic, B. Complex Flexible Manufacturing Systems-Design of Control Structures and Scheduling Strategies, Annals of CIRP, Manufacturing Systems, Vol. 23 (1994), No.1, pp 65-70, ISBN 0176-3377, WISSU Verlag, Aachen & Faculty Press International, Aachen 1994

[26] Katalinic, B., Design of Scheduling Structures and Strategies for Complex Flexible Manufacturing Systems, International Conference on Computer Integrated Manufacturing, CIM 96, May 14-17, 1996. Poland

Oldenbourg Verlag, München Wien 1993

[27] Kusiak A.; Dagli, C. Intelligent Systems in Design and Manufacturing, New York, 1994

[28] Kusiak, A. Intelligent Manufacturing Systems, Prentice Hall, New Jersey, 1990

[29] Law, A.M. & McComas, M.G. Total Support for Simulation Input Modeling, Proceedings of the 1996 Winter Simulation Conference, ISBN 0-7803-7, pp 588-593, Coronado, California, December 8-11, 1996

[30] Ueda, K.. Biological Manufacturing Systems and IMS Program, 7th International DAAAM Symposium, ISBN 3-901509-02-X, Editor B. Katalinic, pp 449 - 452, Wien, 17.-19. Oktober 1996

[31] Katalinic, B. Design and Scheduling of Next Generation of Self-Organising Assembly Sytsems, Chapter 31, DAAAM International Scientific Book 2002, B. Katalinic (ED), pp 299-314, Published by DAAAM International, ISBN: 3-901509-30-5, Viena, Austria

[32] Tadeus Sawik. Production Planning and Scheduling in Flexible Assembly Systems, Springer, ISBN 3540649980, Berlin 1996

[33] F. Gerrit, C. Anna. Overview on sequencing in mixed model flowshop production line with static and dynamic context, Universitat Politecnica de Catalunya, IOC-DT-P-2005-7 Febrer 2005

[34] F.T.S. Chan. Effect of kanban size on just-in-time manufacturing systems, Journal of Materials Processing Technology 116 pp.146-160, Elsevier, 2001

[35] B. Tan, S. Gershwin. On Production and Subcontracting Strategies for Manufacturers with Limited Capacity and Backlog-Dependet Demand, Massachusetts 02139-4307, USA, 2001

[36] S. Griffith. Tools for Designing Programatic Self-Assembling systems, Dec 2002

[37] R. **Salzman.** Manufacturing System Design: Flexible Manufacturing Systems and Value Stream Mapping, Massachusetts Institute of Technology, 2000

[38] G.C.I Lin, S.V. Nagalingam. CIM Justification and optimization.2000. ISBN 0-748-40858-4

[39] R. Bieckert, D. Berling, R.J. Evans, D.G. Kelley. CIM technology: Fundamentals and applications.1998.ISBN 1-566-37426-X

[40] D.Y. Yang, D.G.Ahn, C.H. Lee, C.H. Park, T.J. Kim. Integration of CAD/CAM/CAE/RP for the developoment of metal forming process, Journal of Materials Processing Technology 125-126 pp.26-34, Elsevier, 2002

[41] A. Claudio Garavelli, M. Gorgoglione. A Knowledge-Based Model For The Analysis of Manufacturing Problems, department of Innovation and Enginieering, University of Lecce, Lecce, Italy, POM-2001

[42] T. Sawik. Monolithic vs. Hierarchical balancing and scheduling or a flexible assembly line, Europian Journal of Operational Research 143 pp.115-124, Elseivier,2002

[43] Cheng WU, Yushun FAN, Deyun XIAO. Computer Integrated Manufacturing, Part 1 Handbook of Industrial Engieering, 3rd Edition, to be published by John Wiley& Sons

[44] Hung-Yao HSU, Grier C.I. Lin. Quantitative measurement of component accessibility and product assemblability for design for assembly application, Pergamon, Robotics and Computer Integrated Manufacturing 18 pp. 13-27, 2002

[45] S. Hum, C. Lee. JIT Scheduling Rules: a Simulation Evalution, Pergamon, Omega. Int. Mgmt. Sci. Vol 26, No.3 pp 381-395, Grat Britain, 1998

[46] J. Balic, I. Pahole. Optimisation of intelligent FMS using the data flow matrix method, Journal of Materials Processing Technology 133 pp. 13-20, Elsevier 2003

[47] H. Golnabi. Role of laser sensor systems in automation and flexible manufacturing, Robotics and Computer Integrated Manufacturinf 19 pp.201-210, Pergamon, 2003

[48] G. Vrecer, F.Cus. Planning of tool supply to a group of machines, Journal of Materials Processing Technology 133 pp. 214-217, Elsevier 2003

[49] Ljolic B.; Katalinic B.; Stuja, K; Kordic,V.& Stopper, M. Performance Evalution of Flexible Assembly System for Electrical Motors, Published by DAAAM Internation, Vienna, Austria 2001, Editor B. Katalinic

[50] H. Shin, G. Olling, Y. Chung, B. Kim, S. Cho. An integrated CAAP/CAM system for stamping die pattern machining, Computer Aided Design 35, pp. 203-213, Elsevier, 2003

[51] G. Vieira, J. Herrmann, E. Lin. Predicting the Performance of Rescheduling Strategies for Parallel Machine Systems, Journal of Manufacturing Systems Vol. 19/No.4 Maryland, USA, 2000

[52] G. Vieira, J. Herrmann, E. Lin. Rescheduling Manufacturing Systems: a framework of strategies, policies, and methods, Maryland, USA, MD 20742

[53] K. Mak, Y. Wong, F. Chan. A genetic algorithm for facility layout problems, Computers Integrated Manufacturing Systems Vol. 11, No 1-2, pp 113-127 Elsevier 1998

[54] M. Gamila, S. Motacvalli. A modeling for loading and scheduling problems in FMS, Robotics and Computer Integrated Manufacturing 19 pp.45-55, Pergamon 2003

[55] M. Gamila, S. Motavalli, A modeling technique for loading and scheduling problems in FMS, Robotics and Computer Integrated Manufacturing 19 pp. 45-54

[56] M. Anvar, R. Nagi, Integration of Just-In-Time Production and Material Handling for an Assembly Environment, Buffalo, NY 14260

[57] A.Law, W. Kelton, Simulation Modeling and Analysis 3rd edition

[58] Ch. Sexton, D. Anthony, S. Lewis, C. Please, a. Keane. Design of experiment algorithms for assembled products, University of Southampton, Southampton, England, 2004 S0171BJ

[59] Genetic algorithms for the optimisation of assembly sequences

[60] P.Kochel, U.Nielander, M. Sturm, KASIMIR-object-oriented Kanban SIMulation Imagining Reality, Chemnitzer, 2001

[61] B. Ram, G Patel. Modelling furnace operations using simulation and heuristics, Greesboro, nC 27411, USA 1998

[62] O. Ulgen, G. Coffman, M. Rohrer, J. Shore, D.Sly, D.Wood. Inreasin The Power and Value of Manufacturing Simulation via Collaboration with other Analytical Tools: A Panel Discussion, Procedings of the 1999 Winter Simulation Conference

[63] D. Spinellis, Ch. Papadopoulos. Explore: A Modular Architecture for Production Line Optimisation, DSI 99 pages 1446-1449, Athens, Greece, July 1999

[64] B. Silva, C. Rao. Simultaneous Scheduling of Machines and Material Handling Systems in an FMS, Warangel-506015 (A.P) India

[65] M. Kampf, P.Kochel. Simulation-Based sequencing and lot size optimisation for a production-and-inventory system with multiple items, D—09107 Chemnitz

[66] A. Allen. Probability, Statistics, and Queueing Theory with Computer Science Aplications, Second edition, Academic Press, 1990

[67] F. Chance, J. Robinson, J. Fowler. Supporting Manufacturing with Simulation: Model Design, Development, and Deployment, San Diego, 1996

[68] G. Krieg, H. Kuhn. Analysis of Multi-Class Queuing Systems with Finite Buffers and Setup Times Using Decomposition Methods, 85049 Ingolstadt, Germany

[69] C. Poli. Design for Manufacturing: A Structured Approach, ISBN 0-7506-7341-9, USA 2001

[70] B. Katalinic, K. Stuja, S. Pllana. Arena: Enterprise Wide Modelling and Simulation, Published by DAAAM International pp. 217-218, Vienna, Austria 2000

[71] D. Pedgen. Future directions in simulation. The 11th International DAAAM symposium, „ Intellegent Manufacturing & Automation Man- Machine-Nature“, 19-21st October 2000

[72] W. Dawid, K. Andrew. Design of Assembly Systems for Modular Products, IEEE Transactions on Robotics and Automation, Vol.13, N.5, October 1997

[73] J. Rooda J. Vervoort. Analysis of Manufacturing Systems, Netherlnds, September 2004

[74] T. Tobby, H. John. A Genetic Algorithm for Configuring Reconfigurable Conveyor Components in a Flexible Assembly Line System, Department of Manufacturing Engineering & Engineering Management City University of Hong Kong

[75] P. Song, C. Hicks, F. Earl. Dynamic production planning and rescheduling for complex assemblies , Imperial College, London, SW7 2BP, UK

[76] F. Anwar, R. Nagi. Integration of Just-In-Time Production and Material Handling for an Assembly Environment, Buffalo, New York, NY 14260

[77] W. Cheng, Ch. Paredis. On-LinePlanning of Flexible Assebly Systems: An Agent-Baset Approach, Pittsburgh, Pensylvania 15213-3890

[78] http://ubmail.ubalt.edu/~harsham/simulation/sim.htm

[79] www.doi.ieecomputersociety.org

[80] www.maths.soton.ac.uk

[81] www.nait.org

[82] www.findarticles.com

[83] www.farm.ecs.umass.edu

[84] www.twocw.net

[85] www.home.ubalt.edu

[86] www.eng.morgan.edu

[87] www.citebase.eprints.org

[88] www.arenasimulation.com

PËRFUNDIMI

Sistemi fleksibil montues ATB është një sistem i avancuar i prodhimit. Si i tillë ende klasifikohet si një ndër sistemet me të sofistikuara prodhuese. Në kuadrin e këtij sistemi, ashtu siç u sqarua deri me tani, i gjejmë të gjitha komponentët moderne të ndërlidhura me njëra tjetrën sipas parimit shkak–pasojë. Pra këto sisteme janë shumë komplekse. Ky atribut i këtij sistemi mund të kuptohet si reaksion i tregut të jashtëm me kërkesa shumë të rrepta. Këto kërkesa e diktojnë prodhuesin të marrë kahen e duhur në interes të mbijetimit të tij. Në këtë mënyrë kërkesat e tregut e diktojnë prodhuesin që në kohën sa më të shkurtër të sjellë në treg produkte të shumëllojshme edhe nëse ato janë të destinuara për të njëjtin qëllim. Ky efekt pra ndikon në evolucionin makinerik, logjistik, informatik etj. të sistemeve. Me fjalë tjera ato sisteme duhet të jenë fleksibile.

Në bazë te detyrave të parashtruara në punim, hulumtimeve dhe analizave, mund të konkludohet si vijon:

1. Rritja e vëllimit të prodhimit, respektivisht montimit në krahasim me sistemin paraprak. Kjo detyrë paraqet edhe shkakun e kësaj analize. Në këtë pikëpamje ka ndihmuar shumë Arena dhe OptQuest for Arena për ti gjetur fytat e ngushtë d.m.th. për ti gjetur ato stacione punuese në të cilat vonohet procesi i montimit. Siç vërehet nga tabelat pas simulimit, disa stacione (konkretisht p.sh. stacioni 2,3 dhe 4) janë shumë të ngarkuara (100%). Në ato stacione vendoset stafi plotësues njerëzorë i cili do të lëvizë nga një stacion në tjetrin konform grumbullimit të copave të punës në konvejer. Në këtë mënyrë rritet fluksi i qarkullimit të materialit në konvejer, e më këtë rritet edhe vëllimi i prodhimit.

2. Rritja e kualitetit të prodhimit. Rritja e kualitetit të prodhimit është ndihmuar me implementimin e stacioneve kontrolluese në çdo stacion montues, si dhe me ndihmën e senzorëve dhe kamerave të ndryshme të integruara në stacione të ndryshme. Këto kamera, përveç qëllimit të tyre primar, kryejnë edhe këto detyra duke rritur në këtë mënyrë kualitetin e prodhimit në mënyrë automatike. Ato copa të punës që janë jashtë tolerancave dërgohen në stacionin Repairman. Pasi të kryhet operacioni me sukses, atëherë përcillet në stacionin e ardhshëm, nëse jo del jashtë sistemit. Konkretisht Arena nuk ndihmon në këtë aspekt drejtpërdrejt.

3. Shfrytëzimi më racional i makinerisë dhe pajisjeve tjera duke përfshirë edhe fuqinë njerëzore. Me fjalë të tjera, kjo detyrë nënkuptohet si shfrytëzim më optimal i resurseve për të cilën është diskutuar në kapitullin e fundit. Me OptQuest for Arena arrihet të optimalizohet sistemi. Në stacionet më të ngarkuara vendosen disa resurse shtesë që u definuan në kushtet kufitare në modelin e optimalizimit. Në kushtet kufitare të modelit sigurisht se ekzistojnë kondita kundërthënëse. P.sh. duhet gjetur numrin sa më të vogël të resurseve duke kërkuar në të njëjtën kohë rritja e vëllimit të prodhimit.

Në këtë mënyrë duhet që të gjendet një kompromis në mes këtyre kontradiktave. Në të shumtën e rasteve ato jepen me koeficientet e "rëndësisë". Kështu, nëse kushtit të parë ia japim koeficientin 0,7 dhe të dytit koeficientin 0,3 atëherë optimalizimi do të shkojë në favor të kushtit të parë kufitarë.

Në rastin tonë specifik është optimalizuar sistemi ashtu për të gjitha kushtet janë marrë këta koeficiente të barabartë.

4. Renditja e makinave dhe i Software-ve duhet të jenë ashtu të organizuara që në kohë sa më të shkurtër të kalojnë nga një program prodhimi në tjetrin. Sistemi fleksibil është i pajisur në atë mënyrë që në kohë sa më të shkurtër të riorganizohet sipas programit të ri të montimit. Me fjalë të tjera, ky sistem është projektuar për një spektër të gjerë të produkteve. Nëse arrin ndonjë produkt i ri nevojitet vetëm në rastin më të keq ndërrimi i Endefekt-orit (kapësja, dorëza e Robotit).

Çka është më e rëndësishmja, me anë të OptQuest for Arena është ndërtuar një Modul gjegjësisht Modeli i optimalizimit, i cili për renditje të tillë është në gjendje në kohë shumë të shkurtër (varësisht nga perioda e planifikimit dhe kushtet kufitare-për disa minuta) të gjejë:

1. Renditjen optimale të porosive sipas kohës me të cilën ato fillojnë të punohen si dhe numrin e tyre në grup.
2. Numrin optimal të fuqisë njerëzore për atë periudhë të planifikimit.
3. Fytet e ngushtë dhe ti eliminojë gjatë kësaj periudhe të planifikimit.

Dhe e tëra kjo bëhet Off-Line paraprakisht, paralel në aspektin kohor përderisa sistemi

është duke montuar me programin e vjetër të prodhimit d.m.th. ende nuk ka kaluar në programin e ri.

Pa dyshim se çdo sistem i prodhimit, respektivisht i montimit, sjellë përparësitë si dhe të metat e veta. Ndër të metat qenësore të këtij sistemi numërohen:

1. Lidhja serike e komponentëve. Kjo do të thotë se makina më e ngadalshme e dikton shpejtësinë e tërë fluksit të rrjedhjes së materialit.
2. Pastaj nga aspekti i besueshmërisë së sistemit, rënia e një makine e shpie sistemin në bllokadë të pjesërishme respektivisht nëse riparimi i sistemit zgjat shumë edhe në bllokadën totale të sistemit. Për këtë arsye në të ardhmen duhet të kihet parasysh për stacionet më të ngarkuara të ketë mundësi zgjerimi për një makinë paralele.
3. Kur ndonjë operacion nuk kryhet me sukses, atëherë ai dërgohet në linjën riparuese (konkretisht në sistemin tonë është paraqitur si Repairman). Kjo do të thotë sikur edhe tek simulimi se pjesët nga operacioni jo i suksesshëm duhet të kaloj një numër të caktuar stacionesh varësisht nga gabimi derisa të arrijë tek stacioni riparues.
4. Rritja e fleksibilitetit të sistemit shkon në favor të kompleksicitetit për menaxhimin e këtyre sistemeve. Kjo do të thotë se duhet të disponohet me një Software e cila bën Scheduling-un (renditjen e porosive sipas fillimit të montimit), dhe me këtë pra nënkuptohet se duhet të licencohet Softwaret dhe të disponohet me stafin e ekspertëve përkatës. Këto edhe ndikojnë në rritjen e harxhimeve-kostove.

Propozimi i të ardhshmes së sistemit shkon në drejtim të robotëve Mobil. Kështu robotët mobilë përveç që e bëjnë transportin e paletës prej fillimit të montimit deri në fund ato zgjedhin rrugën më të shkurtë deri te stacioni i ardhshëm. Në këtë mënyrë ky sistem montues do të organizohej në formë rrethore ku në qendër të rrethit do të ishte stacioni riparues dhe në periferi stacionet punuese. Kështu, e meta numër tre e këtij sistemi, e cekur më lartë, do të eliminohej sepse në qendër të rrethit ndodhet stacioni riparues dhe pa marrë parasysh se ku ndodhë defekti rruga e kaluar nga roboti mobil është e barabartë më rrezen e këtij rrethi. Pra rruga e kaluar është e njëjtë pa marrë parasysh se ku ndodhë defekti.

Pastaj robotët do të ndihmojnë në gjetjen e stacionit të lirë për montim. Kështu nëse në robot mobil ndodhet një kombinim i cili është në operacionin 6 si dhe operacioni i ardhshëm mund të kryhet ose në stacionin 7 apo në stacionin 12. Supozojmë se në stacioni 7 është i zënë kurse ai 12 është i lirë ose më pak pjesë presin në rend në atë stacion, atëherë roboti mobil merr sinjalin dhe shkon në atë drejtim. Kjo e lehtëson shumë menaxhimin e këtij sistemi. D.m.th. kjo ndikon direkt në të metën e katërt të këtij sistemi.

Këto pra ishin pikat kryesore të analizimit të këtij sistemi. Kjo analizë ka të bëjë me kohën e tashme. Ka të bëjë me gjendjen e tanishme të sistemit fleksibil montues dhe gjendjen e tanishme të zhvillimit të kompjuterit si dhe të veglës për simulim. Çfarëdo ndryshimi në aspektin e sistemit fleksibil montues nga ato për të cilat është punuar kjo tezë, do të tregojnë përputhje të pjesërishme apo përputhje shumë të vogla me rezultatet e simulimit.

Summary

The flexible assembly system ATB is an advanced productive system. It is rated along with the most sophisticated productive systems. Within this system, as it was described sooner, all the modern components inter connected to each other could be found based in the **supply-outcome** principle. Therefore, these are very complex systems. This system attribute could be perceived as the reflection of the external market with its strict requirements. These requirements dictate the manufacturer taking the right direction on his existing/survival interest. It means that market demands dictate the manufacturer to release into market different products in shorter possible time even those to be dedicated for the same purpose. This outcome affects the evolution of mechanical, logistical, and information systems. Furthermore these systems should be flexible.

Based in the objectives of the work, researches and analysis, it could be concluded as follows:

1. The increase of production volume, respectively the assembling in comparison with previous system. This task presents the objective of this analysis. This was supported to a great extent by Arena and OptQuest for Arena for finding the bottle-necks i.e. for finding the working stations where assembling process is slow. As it could be seen in the tables after the simulation some of the stations (concretely for example, stations 2, 3 and 4) are very loaded/busy (100%). In these stations the additional staff has been placed that could move from one station to another, depending on amassment of production parts in the conveyer. In this way the flow of the material afflux in conveyer has increased, increasing at the same time the production volume.

2. The increase of the production quality. The increase of the production quality was supported by the implementation of the control stations at every assembling station, and also by sensors and special cameras integrated in different stations. These cameras, besides their primer role, do their tasks in a manner of automatic enlargement of production quality. The working pieces which are over the tolerance will be sent to the Repairman station. If operation is successfully completed these pieces will be sent to the further station, if not will be taken out of the system. The Arena doesn't help directly in this aspect.

3. Rational utilization of machines and other equipments including also the human resources. This task means optimal utilization of the resources discussed in the last chapter. The Opt Quest for Arena has helped to succeed in optimizing the system. In the most overloaded stations some additional resources were put which were defined as boundary constraints of the optimizing model. Within these constraints, of course

exists some contradicts, for example – for minimal number of resources the increase of production volume is needed to be achieved.

So, a compromise solution between these contradicts is needed to be found. In most cases these were given with the coefficient of "importance". Therefore, if we give to the first condition the 0,7 coefficient, and to next one the coefficient 0,3 then the optimization will go in favor of first limited condition.

In specific case under discussion the system has been optimized taking into account that these coefficients are equal under all conditions.

4. The layout of machines and software should be organized in such a way that in a very short time the move from one production program to another is possible. The flexible system is equipped that it could be reorganized in short time according to the new assembling program. This system is designed for a wide number of products. If a new product comes up the change of Endeffector (fastener, Robot handle) will be needed only at the worst case.

What is most important thing, by the OptQuest for Arena a module can be built up, respectively a model of optimization, which is ready for such a schedule and at a very short time (depending on planning period and boundary conditions- for few minutes) finds:

1. The optimal scheduling of messages according to time they start to b manufactured and their number in a group.
2. The optimal number of human resources for such a planning period.
3. The bottle-necks and eliminates them during the planning period.

All this will be done of-line in advance, by the same time when system is assembling with the old production program i.e. before transferring to the new program.

No doubt that every production system, particularly assembling one, brings the advantages and its disadvantages. The most disadvantages of this system are as follows:

1. Serial connection of components. This means that slowest machine dictates the speed of the flow process of the material.

2. Then, in the aspect of system reliability, the failure of one machine makes the partial obstruction of the whole system, in particular if the system repair takes to long to the total blockade of the system. For this reason in future needs to be taken into consideration possibility of extension with the additional parallel machine to the overloaded stations.

3. When an operation was not successfully completed, then it should be sent to the reparation line (concretely in our system it is presented like a Repairman). This means that like in a simulation, the pieces from unsuccessful operation should pass certain stations depending on the error until they arrive at reparation station.

4. The increase of the system flexibility goes in favor of complexity for managing of those systems. It means that the support of a software for the Scheduling (to schedule the instructions based on assembling start) is needed, a software to be licensed and handled by respective staff of experts. These have impacts on cost raise too.

The future suggestion for the system goes in the direction to the mobile robots. The mobile robots not only can do the transportation of the pallets from the assembling start until the end but they can find the shortest way to the next station. In this manner the assembling system would be organized in the circle form, where in the centre of the circle will be the repairing station, while the working stations in it perimeter. Therefore, the limitation 3 of this system, mentioned before would be eliminated, because in the circle center is the repairing station and no matter where a defect is, the way run by the mobile robot it is equally with the circle radius. So, the race/distance is the same no matter where that defect happened.

Afterwards, the robots will help in finding the unoccupied station for assembling. If in the mobile robot exists a combination which is in the operation 6, in the meantime the next operation could be done also in the station 7, or in the station 12. For instance, if the mobile robot is busy in the station 7, while the number 12 is free, or fewer pieces are waiting in the row in that station, than the mobile robot gets the signal and moves to that direction. This makes the managing of the system very easy. This has direct impact in the fourth defect of this system.

Therefore, these are the crucial points of system analysis. This analysis has to do for the current situation. It describes the actual situation of the flexible assembling system and the current situation of the computer development and the stimulation tool. No matter what changes in the aspect of the flexible assembling system from this thesis has been done; they will show partial compatibility or slight correspondence with the simulation results.

SH T O J C A

Software për sistemin e elektromotorëve në fabrikën ATB – Wiena
i programuar përmes ARENA 7.0

SHTOJCA

Model statements for module: Create 1

227$ CREATE, 1, SecondstoBaseTime(0.0), Entity 1:SecondstoBaseTime(3), 200:
NEXT(228$)
228$ ASSIGN: Create 1.NumberOut=Create 1.NumberOut + 1:
NEXT(29$)

Model statements for module: Assign 1

29$ ASSIGN: OPERATION=0:
NEXT(0$)

Model statements for module: Station 1

0$ STATION, ENTRY
233$ DELAY: 0.0, , VA:
NEXT(1$)

Model statements for module: ACCESS 1

1$ QUEUE, ACCESS EntryNode1.Queue
ACCESS: Conveyor 1, 1:
NEXT(3$)

Model statements for module: Convey 1

3$ CONVEY: Conveyor 1, NODE 1

Model statements for module: Station 2

4$ STATION, NODE 1
236$ DELAY: 0.0, , VA:
NEXT(5$)

Model statements for module: ACCESS 2

5$ QUEUE, ACCESS Node1Node2.Queue
ACCESS: Conveyor 2, 1:

NEXT(86$)

Model statements for module: Decide 9

86$ BRANCH, 1:
If, OPERATION==0, 9$, Yes:
Else, 85$, Yes

Model statements for module: Exit 20

85$ EXIT: Conveyor 17:
NEXT(7$)

Model statements for module: Convey 2

7$ CONVEY: Conveyor 2, NODE 2

Model statements for module: Exit 1

9$ EXIT: Conveyor 1:
NEXT(7$)

Model statements for module: Station 3

8$ STATION, NODE 2
241$ DELAY: 0.0, , VA:
NEXT(11$)

Model statements for module: Decide 1

11$ BRANCH, 1:
If, OPERATION==0, 12$, Yes:
Else, 16$, Yes

Model statements for module: ACCESS 4

16$ QUEUE, ACCESS Node2RLine.Queue
ACCESS: Conveyor 4, 1:
NEXT(177$)

Model statements for module: Decide 52

177$ BRANCH, 1:
If, OPERATION==(OPERATION >= 1) || (OPERATION <= 8), 19$, Yes:
Else, 178$, Yes

Model statements for module: Exit 29

178$ EXIT: Conveyor 2:
NEXT(18$)

Model statements for module: Convey 4

18$ CONVEY: Conveyor 4, RLine

Model statements for module: Exit 3

19$ EXIT: Conveyor 15:
NEXT(18$)

Model statements for module: ACCESS 3

12$ QUEUE, ACCESS Node2Node3.Queue
ACCESS: Conveyor 3, 1:
NEXT(15$)

Model statements for module: Exit 2

15$ EXIT: Conveyor 2:
NEXT(14$)

Model statements for module: Convey 3

14$ CONVEY: Conveyor 3, NODE 3

Model statements for module: Station 4

10$ STATION, NODE 3
248$ DELAY: 0.0, , VA:
NEXT(20$)

Model statements for module: Decide 2

20$ BRANCH, 1:
If, OPERATION>=10, 21$, Yes:
Else, 25$, Yes

Model statements for module: ACCESS 6

25$ QUEUE, ACCESS Node3Node4.Queue
ACCESS: Conveyor 5, 1:
NEXT(87$)

Model statements for module: Decide 14

87$ BRANCH, 1:
If, OPERATION==0, 28$, Yes:
Else, 88$, Yes

Model statements for module: Exit 24

88$ EXIT: ConRLN3:
NEXT(27$)

Model statements for module: Convey 6

27$ CONVEY: Conveyor 5, NODE 4

Model statements for module: Exit 5

28$ EXIT: Conveyor 3:
NEXT(27$)

Model statements for module: ACCESS 5

21$ QUEUE, ACCESS Node3Node8.Queue
ACCESS: Conveyor 6, 1:
NEXT(24$)

Model statements for module: Exit 4

24$ EXIT: ConRLN3:
NEXT(23$)

Model statements for module: Convey 5

23$ CONVEY: Conveyor 6, NODE 8

Model statements for module: Station 5

30$ STATION, NODE 4
255$ DELAY: 0.0, , VA:
NEXT(31$)

Model statements for module: Decide 3

31$ BRANCH, 1:
If, OPERATION>=4, 256$, Yes:
Else, 257$, Yes
256$ ASSIGN: DECIDE NODE 4.NumberOut True=DECIDE NODE 4.NumberOut True + 1:
NEXT(32$)

257$ ASSIGN: DECIDE NODE 4.NumberOut False=DECIDE NODE 4.NumberOut False + 1:
NEXT(36$)

Model statements for module: ACCESS 7

32$ QUEUE, ACCESS Node5.Queue
ACCESS: Conveyor 8, 1:
NEXT(35$)

Model statements for module: Exit 6

35$ EXIT: Conveyor 5:
NEXT(34$)

Model statements for module: Convey 7

34$ CONVEY: Conveyor 8, NODE 5

Model statements for module: ACCESS 8

36$ QUEUE, ACCESS Node5Long.Queue
ACCESS: Conveyor 7, 1:
NEXT(39$)

Model statements for module: Exit 7

39$ EXIT: Conveyor 5:
NEXT(38$)

Model statements for module: Convey 8

38$ CONVEY: Conveyor 7, 1ENDSHIELD

Model statements for module: Station 6

40$ STATION, NODE 5
260$ DELAY: 0.0, , VA:
NEXT(41$)

Model statements for module: ACCESS 9

41$ QUEUE, ACCESS Node5Node6.Queue
ACCESS: Conveyor 9, 1:
NEXT(172$)

Model statements for module: Decide 49

172$ BRANCH, 1:
If, OPERATION>=4, 261$, Yes:
Else, 262$, Yes
261$ ASSIGN: DECIDE NODE 5.NumberOut True=DECIDE NODE 5.NumberOut True + 1:
NEXT(44$)

262$ ASSIGN: DECIDE NODE 5.NumberOut False=DECIDE NODE 5.NumberOut False + 1:
NEXT(174$)

Model statements for module: Exit 8

44$ EXIT: Conveyor 8:
NEXT(43$)

Model statements for module: Convey 9

43$ CONVEY: Conveyor 9, NODE 6

Model statements for module: Exit 27

174$ EXIT: ConST4N5:
NEXT(173$)

Model statements for module: Decide 51

173$ BRANCH, 1:
If, OPERATION==0.4, 263$, Yes:
Else, 264$, Yes
263$ ASSIGN: DECIDE NODE 5P.NumberOut True=DECIDE NODE 5P.NumberOut True + 1:
NEXT(175$)

264$ ASSIGN: DECIDE NODE 5P.NumberOut False=DECIDE NODE 5P.NumberOut False + 1:
NEXT(43$)

Model statements for module: Assign 20

175$ ASSIGN: OPERATION=4:
NEXT(43$)

Model statements for module: Station 7

45$ STATION, NODE 6
267$ DELAY: 0.0, , VA:
NEXT(46$)

Model statements for module: Decide 5

46$ BRANCH, 1:
If, OPERATION<=3, 47$, Yes:
Else, 51$, Yes

Model statements for module:ACCESS 12

51$ QUEUE, ACCESS Node6Node7.Queue
ACCESS: Conveyor 10, 1:
NEXT(54$)

Model statements for module: Exit 11

54$ EXIT: Conveyor 9:
NEXT(53$)

Model statements for module: Convey 12

53$ CONVEY: Conveyor 10, 2ENDSHIELD

Model statements for module: ACCESS 11

47$ QUEUE, ACCESS Node6Long.Queue
ACCESS: Conveyor 11, 1:
NEXT(50$)

Model statements for module: Exit 10

50$ EXIT: Conveyor 9:
NEXT(49$)

Model statements for module: Convey 11

49$ CONVEY: Conveyor 11, NODE 7

Model statements for module: Station 8

55$ STATION, NODE 7
272$ DELAY: 0.0, , VA:
NEXT(56$)

Model statements for module:ACCESS 13

56$ QUEUE, ACCESS Node7Node8.Queue
ACCESS: Conveyor 12, 1:

NEXT(89$)

Model statements for module: Decide 15

89$ BRANCH, 1:
If, OPERATION<=3, 59$, Yes:
Else, 90$, Yes

Model statements for module: Exit 25

90$ EXIT: ConST9N7:
NEXT(58$)

Model statements for module: Convey 13

58$ CONVEY: Conveyor 12, NODE 8

Model statements for module: Exit 12

59$ EXIT: Conveyor 11:
NEXT(58$)

Model statements for module: Station 9

60$ STATION, NODE 8
277$ DELAY: 0.0, , VA:
NEXT(61$)

Model statements for module: ACCESS 14

61$ QUEUE, ACCESS Node8 Node9.Queue
ACCESS: Conveyor 13, 1:
NEXT(91$)

Model statements for module: Decide 16

91$ BRANCH, 1:
If, OPERATION<=9, 64$, Yes:
Else, 92$, Yes

Model statements for module: Exit 26

92$ EXIT: Conveyor 6:
NEXT(63$)

Model statements for module: Convey 14

63$ CONVEY: Conveyor 13, NODE 9

Model statements for module: Exit 13

64$ EXIT: Conveyor 12:
NEXT(63$)

Model statements for module: Station 10

65$ STATION, NODE 9
282$ DELAY: 0.0, , VA:
NEXT(66$)

Model statements for module: Decide 6

66$ BRANCH, 1:
If, OPERATION>8, 67$, Yes:
Else, 71$, Yes

Model statements for module: ACCESS 16

71$ QUEUE, ACCESS Node9Node2.Queue
ACCESS: Conveyor 15, 1:
NEXT(74$)

Model statements for module: Exit 15

74$ EXIT: Conveyor 13:
NEXT(73$)

Model statements for module: Convey 16

73$ CONVEY: Conveyor 15, NODE 2

Model statements for module: ACCESS 15

67$ QUEUE, ACCESS Node9Node10.Queue
ACCESS: Conveyor 14, 1:
NEXT(70$)

Model statements for module: Exit 14

70$ EXIT: Conveyor 13:
NEXT(69$)

Model statements for module: Convey 15

69$ CONVEY: Conveyor 14, STATION 10

Model statements for module: Station 11

75$ STATION, NODE 10
287$ DELAY: 0.0, , VA:
NEXT(76$)

Model statements for module: Decide 7

76$ BRANCH, 1:
If, OPERATION==14, 77$, Yes:
Else, 81$, Yes

Model statements for module: ACCESS 18

81$ QUEUE, ACCESS Node10Node1.Queue
ACCESS: Conveyor 17, 1:
NEXT(84$)

Model statements for module: Exit 17

84$ EXIT: ConST14N10:
NEXT(83$)

Model statements for module: Convey 18

83$ CONVEY: Conveyor 17, NODE 1

Model statements for module: ACCESS 17

77$ QUEUE, ACCESS Node10EXIT.Queue
ACCESS: Conveyor 16, 1:
NEXT(80$)

Model statements for module: Exit 16

80$ EXIT: ConST14N10:
NEXT(79$)

Model statements for module: Convey 17

79$ CONVEY: Conveyor 16, PACKING

Model statements for module: Station 28

167$ STATION, EXIT1
292$ DELAY: 0.0, , VA:
NEXT(176$)

Model statements for module: Exit 28

176$ EXIT: ConPACEXIT:
NEXT(168$)

Model statements for module: Dispose 1

168$ ASSIGN: Dispose 1.NumberOut=Dispose 1.NumberOut + 1
293$ DISPOSE: Yes

Model statements for module: Enter 1

179$ STATION, 1ENDSHIELD
294$ DELAY: 0., , VA:
NEXT(297$)
297$ EXIT: Conveyor 7:
NEXT(93$)

Model statements for module: Decide 17

93$ BRANCH, 1:

If, OPERATION==0, 305$, Yes:
Else, 306$, Yes
305$ ASSIGN: Decide 17.NumberOut True=Decide 17.NumberOut True + 1:
NEXT(94$)
306$ ASSIGN: Decide 17.NumberOut False=Decide 17.NumberOut False + 1:
NEXT(180$)

Model statements for module: Process 1

94$ ASSIGN: Process 1.NumberIn=Process 1.NumberIn + 1:
Process 1.WIP=Process 1.WIP+1
310$ QUEUE, Process 1.Queue
309$ SEIZE, 2, VA:
Resource 1, 1:
NEXT(308$)
308$ DELAY: 5, , VA
307$ RELEASE: Resource 1, 1
355$ ASSIGN: Process 1.NumberOut=Process 1.NumberOut + 1:
Process 1.WIP=Process 1.WIP-1:
NEXT(100$)

Model statements for module: Assign 2

100$ ASSIGN: OPERATION=OPERATION+1:
NEXT(180$)

Model statements for module: ACCESS 19

180$ QUEUE, Con ST1ST2.Queue
ACCESS: ConST1ST2, 1:
NEXT(95$)

Model statements for module: Convey 19

95$ CONVEY: ConST1ST2, STATION 02

Model statements for module: Enter 2

182$ STATION, STATION 02
358$ DELAY: 0., , VA:
NEXT(361$)
361$ EXIT: ConST1ST2:
NEXT(96$)

Model statements for module: Decide 19

96$ BRANCH, 1:
If, OPERATION==1, 369$, Yes:
Else, 370$, Yes
369$ ASSIGN: Decide 19.NumberOut True=Decide 19.NumberOut True + 1:
NEXT(97$)
370$ ASSIGN: Decide 19.NumberOut False=Decide 19.NumberOut False + 1:
NEXT(183$)

Model statements for module: Process 3

97$ ASSIGN: Process 2.NumberIn=Process 2.NumberIn + 1:
Process 2.WIP=Process 2.WIP+1
374$ QUEUE, Process 2.Queue
373$ SEIZE, 2, VA:
Resource 2, 1:
NEXT(372$)
372$ DELAY: Triangular(8, 10, 12), , VA
371$ RELEASE: Resource 2, 1
419$ ASSIGN: Process 2.NumberOut=Process 2.NumberOut + 1:
Process 2.WIP=Process 2.WIP-1:
NEXT(99$)

Model statements for module: Decide 20

99$ BRANCH, 1:
With, 95/100, 101$, Yes:
Else, 183$, Yes

Model statements for module: ACCESS 20

183$ QUEUE, Con ST2ST3.Queue
ACCESS: ConST2ST3, 1:
NEXT(98$)

Model statements for module: Convey 20

98$ CONVEY: ConST2ST3, STATION 03

Model statements for module: Assign 3

101$ ASSIGN: OPERATION=OPERATION+1:
NEXT(183$)

Model statements for module: Enter 3

185$ STATION, STATION 03
424$ DELAY: 0., , VA:
NEXT(427$)
427$ EXIT: ConST2ST3:
NEXT(102$)

Model statements for module: Decide 21

102$ BRANCH, 1:
If, OPERATION==2, 435$, Yes:
Else, 436$, Yes
435$ ASSIGN: Decide 21.NumberOut True=Decide 21.NumberOut True + 1:
NEXT(103$)
436$ ASSIGN: Decide 21.NumberOut False=Decide 21.NumberOut False + 1:
NEXT(186$)

Model statements for module: Process 4

103$ ASSIGN: Process 3.NumberIn=Process 3.NumberIn + 1:
Process 3.WIP=Process 3.WIP+1
440$ QUEUE, Process 3.Queue
439$ SEIZE, 2, VA:
Resource 3, 1:
NEXT(438$)
438$ DELAY: Triangular(10, 14, 18), , VA
437$ RELEASE: Resource 3, 1
485$ ASSIGN: Process 3.NumberOut=Process 3.NumberOut + 1:
Process 3.WIP=Process 3.WIP-1:
NEXT(105$)

Model statements for module: Decide 22

105$ BRANCH, 1:
With, 95/100, 106$, Yes:
Else, 186$, Yes

Model statements for module:ACCESS 21

186$ QUEUE, Con ST3ST4.Queue
ACCESS: ConST3ST4, 1:
NEXT(104$)

Model statements for module: Convey 21

104$ CONVEY: ConST3ST4, STATION 04

Model statements for module: Assign 4

106$ ASSIGN: OPERATION=OPERATION+1:
NEXT(186$)

Model statements for module: Enter 4

188$ STATION, STATION 04
490$ DELAY: 0., , VA:
NEXT(493$)
493$ EXIT: ConST3ST4:
NEXT(107$)

Model statements for module: Decide 23

107$ BRANCH, 1:
If, OPERATION==3, 501$, Yes:
Else, 502$, Yes
501$ ASSIGN: Decide 23.NumberOut True=Decide 23.NumberOut True + 1:
NEXT(108$)
502$ ASSIGN: Decide 23.NumberOut False=Decide 23.NumberOut False + 1:
NEXT(189$)

Model statements for module: Process 5

108$ ASSIGN: Process 4.NumberIn=Process 4.NumberIn + 1:
Process 4.WIP=Process 4.WIP+1
506$ QUEUE, Process 4.Queue
505$ SEIZE, 2, VA:
Resource 4, 1:
NEXT(504$)
504$ DELAY: Triangular(13, 15, 17), , VA

503$ RELEASE: Resource 4, 1
551$ ASSIGN: Process 4.NumberOut=Process 4.NumberOut + 1:
Process 4.WIP=Process 4.WIP-1:
NEXT(110$)

Model statements for module: Decide 24

110$ BRANCH, 1:
With, 95/100, 111$, Yes:
Else, 189$, Yes

Model statements for module: ACCESS 22

189$ QUEUE, Con ST4N5.Queue
ACCESS: ConST4N5, 1:
NEXT(109$)

Model statements for module: Convey 22

109$ CONVEY: ConST4N5, NODE 5

Model statements for module: Assign 5

111$ ASSIGN: OPERATION=0.4:
NEXT(189$)

Model statements for module: Enter 5

191$ STATION, 2ENDSHIELD
556$ DELAY: 0., , VA:
NEXT(559$)
559$ EXIT: Conveyor 10:
NEXT(112$)

Model statements for module: Decide 25

112$ BRANCH, 1:
If, OPERATION==4, 567$, Yes:
Else, 568$, Yes
567$ ASSIGN: Decide 25.NumberOut True=Decide 25.NumberOut True + 1:
NEXT(113$)
568$ ASSIGN: Decide 25.NumberOut False=Decide 25.NumberOut False + 1:

NEXT(192$)

Model statements for module: Process 6

113$ ASSIGN: Process 5.NumberIn=Process 5.NumberIn + 1:
Process 5.WIP=Process 5.WIP+1
572$ QUEUE, Process 5.Queue
571$ SEIZE, 2, VA:
Resource 5, 1:
NEXT(570$)
570$ DELAY: Triangular(12, 14, 16), , VA
569$ RELEASE: Resource 5, 1
617$ ASSIGN: Process 5.NumberOut=Process 5.NumberOut + 1:
Process 5.WIP=Process 5.WIP-1:
NEXT(115$)

Model statements for module: Decide 26

115$ BRANCH, 1:
With, 95/100, 116$, Yes:
Else, 192$, Yes

Model statements for module:ACCESS 23

192$ QUEUE, ConST5ST6.Queue
ACCESS: ConST5ST6, 1:
NEXT(114$)

Model statements for module: Convey 23

114$ CONVEY: ConST5ST6, STATION 06

Model statements for module: Assign 6

116$ ASSIGN: OPERATION=OPERATION+1:
NEXT(192$)

Model statements for module: Enter 6

194$ STATION, STATION 06
622$ DELAY: 0., , VA:
NEXT(625$)

625$ EXIT: ConST5ST6:
NEXT(117$)

Model statements for module: Decide 27

117$ BRANCH, 1:
If, OPERATION==5, 633$, Yes:
Else, 634$, Yes
633$ ASSIGN: Decide 27.NumberOut True=Decide 27.NumberOut True + 1:
NEXT(118$)
634$ ASSIGN: Decide 27.NumberOut False=Decide 27.NumberOut False + 1:
NEXT(195$)

Model statements for module: Process 7

118$ ASSIGN: Process 6.NumberIn=Process 6.NumberIn + 1:
Process 6.WIP=Process 6.WIP+1
638$ QUEUE, Process 6.Queue
637$ SEIZE, 2, VA:
Resource 6, 1:
NEXT(636$)
636$ DELAY: Triangular(10, 12, 14), , VA
635$ RELEASE: Resource 6, 1
683$ ASSIGN: Process 6.NumberOut=Process 6.NumberOut + 1:
Process 6.WIP=Process 6.WIP-1:
NEXT(120$)

Model statements for module: Decide 28

120$ BRANCH, 1:
With, 95/100, 121$, Yes:
Else, 195$, Yes

Model statements for module: ACCESS 24

195$ QUEUE, Con ST6ST7.Queue
ACCESS: ConST6ST7, 1:
NEXT(119$)

Model statements for module: Convey 24

119$ CONVEY: ConST6ST7, STATION 07

Model statements for module: Assign 7

121$ ASSIGN: OPERATION=OPERATION+1:
NEXT(195$)

Model statements for module: Enter 7

199$ STATION, STATION 07
688$ DELAY: 0., , VA:
NEXT(691$)
691$ EXIT: ConST6ST7:
NEXT(122$)

Model statements for module: Decide 29

122$ BRANCH, 1:
If, OPERATION==6, 699$, Yes:
Else, 700$, Yes
699$ ASSIGN: Decide 29.NumberOut True=Decide 29.NumberOut True + 1:
NEXT(123$)
700$ ASSIGN: Decide 29.NumberOut False=Decide 29.NumberOut False + 1:
NEXT(197$)

Model statements for module: Process 8

123$ ASSIGN: Process 7.NumberIn=Process 7.NumberIn + 1:
Process 7.WIP=Process 7.WIP+1
704$ QUEUE, Process 7.Queue
703$ SEIZE, 2, VA:
Resource 7, 1:
NEXT(702$)
702$ DELAY: Triangular(11, 15, 19), , VA
701$ RELEASE: Resource 7, 1
749$ ASSIGN: Process 7.NumberOut=Process 7.NumberOut + 1:
Process 7.WIP=Process 7.WIP-1:
NEXT(125$)

Model statements for module: Decide 30

125$ BRANCH, 1:
With, 95/100, 126$, Yes:
Else, 197$, Yes

Model statements for module: ACCESS 25

197$ QUEUE, Con ST7ST8.Queue
ACCESS: ConST7ST8, 1:
NEXT(124$)

Model statements for module: Convey 25

124$ CONVEY: ConST7ST8, STATION 08

Model statements for module: Assign 8

126$ ASSIGN: OPERATION=OPERATION+1:
NEXT(197$)

Model statements for module: Enter 8

200$ STATION, STATION 08
754$ DELAY: 0., , VA:
NEXT(757$)
757$ EXIT: ConST7ST8:
NEXT(127$)

Model statements for module: Decide 31

127$ BRANCH, 1:
If, OPERATION==7, 765$, Yes:
Else, 766$, Yes
765$ ASSIGN: Decide 31.NumberOut True=Decide 31.NumberOut True + 1:
NEXT(128$)
766$ ASSIGN: Decide 31.NumberOut False=Decide 31.NumberOut False + 1:
NEXT(201$)

Model statements for module: Process 9

128$ ASSIGN: Process 8.NumberIn=Process 8.NumberIn + 1:
Process 8.WIP=Process 8.WIP+1
770$ QUEUE, Process 8.Queue
769$ SEIZE, 2, VA:
Resource 8, 1:

NEXT(768$)
768$ DELAY: Triangular(13, 14, 15), , VA
767$ RELEASE: Resource 8, 1
815$ ASSIGN: Process 8.NumberOut=Process 8.NumberOut + 1:
Process 8.WIP=Process 8.WIP-1:
NEXT(130$)

Model statements for module: Decide 32

130$ BRANCH, 1:
With, 95/100, 131$, Yes:
Else, 201$, Yes

Model statements for module: ACCESS 26

201$ QUEUE, Con ST8ST9.Queue
ACCESS: ConST8ST9, 1:
NEXT(129$)

Model statements for module: Convey 26

129$ CONVEY: ConST8ST9, STATION 09

Model statements for module: Assign 9

131$ ASSIGN: OPERATION=OPERATION+1:
NEXT(201$)

Model statements for module: Enter 9

203$ STATION, STATION 09
820$ DELAY: 0., , VA:
NEXT(823$)
823$ EXIT: ConST8ST9:
NEXT(132$)

Model statements for module: Decide 33

132$ BRANCH, 1:
If, OPERATION==8, 831$, Yes:
Else, 832$, Yes
831$ ASSIGN: Decide 33.NumberOut True=Decide 33.NumberOut True + 1:
NEXT(133$)

832$ ASSIGN: Decide 33.NumberOut False=Decide 33.NumberOut False + 1:
NEXT(204$)

Model statements for module: Process 10

133$ ASSIGN: Process 9.NumberIn=Process 9.NumberIn + 1:
Process 9.WIP=Process 9.WIP+1
836$ QUEUE, Process 9.Queue
835$ SEIZE, 2, VA:
Resource 9, 1:
NEXT(834$)
834$ DELAY: Triangular(10, 12, 14), , VA
833$ RELEASE: Resource 9, 1
881$ ASSIGN: Process 9.NumberOut=Process 9.NumberOut + 1:
Process 9.WIP=Process 9.WIP-1:
NEXT(135$)

Model statements for module: Decide 34

135$ BRANCH, 1:
With, 95/100, 136$, Yes:
Else, 204$, Yes

Model statements for module: ACCESS 27

204$ QUEUE, Con ST9N7.Queue
ACCESS: ConST9N7, 1:
NEXT(134$)

Model statements for module: Convey 27

134$ CONVEY: ConST9N7, NODE 7

Model statements for module: Assign 10

136$ ASSIGN: OPERATION=OPERATION+1:
NEXT(204$)

Model statements for module: Enter 10

206$ STATION, STATION 10
886$ DELAY: 0., , VA:

NEXT(889$)
889$ EXIT: Conveyor 14:
NEXT(137$)

Model statements for module: Decide 37

137$ BRANCH, 1:
If, OPERATION==9, 897$, Yes:
Else, 898$, Yes
897$ ASSIGN: Decide 37.NumberOut True=Decide 37.NumberOut True + 1:
NEXT(138$)
898$ ASSIGN: Decide 37.NumberOut False=Decide 37.NumberOut False + 1:
NEXT(208$)

Model statements for module: Process 12

138$ ASSIGN: Process 10.NumberIn=Process 10.NumberIn + 1:
Process 10.WIP=Process 10.WIP+1
902$ QUEUE, Process 10.Queue
901$ SEIZE, 2, VA:
Resource 10, 1:
NEXT(900$)
900$ DELAY: Triangular(13, 14, 15), , VA
899$ RELEASE: Resource 10, 1
947$ ASSIGN: Process 10.NumberOut=Process 10.NumberOut + 1:
Process 10.WIP=Process 10.WIP-1:
NEXT(140$)

Model statements for module: Decide 38

140$ BRANCH, 1:
With, 95/100, 141$, Yes:
Else, 208$, Yes

Model statements for module: ACCESS 28

208$ QUEUE, ACCESS ConST10ST11.Queue
ACCESS: ConST10ST11, 1:
NEXT(139$)

Model statements for module: Convey 29

139$ CONVEY: ConST10ST11, STATION 11

Model statements for module: Assign 12

141$ ASSIGN: OPERATION=OPERATION+1:
NEXT(208$)

Model statements for module: Enter 11

207$ STATION, STATION 11
952$ DELAY: 0., , VA:
NEXT(955$)
955$ EXIT: ConST10ST11:
NEXT(142$)

Model statements for module: Decide 39

142$ BRANCH, 1:
If, OPERATION==10, 963$, Yes:
Else, 964$, Yes
963$ ASSIGN: Decide 39.NumberOut True=Decide 39.NumberOut True + 1:
NEXT(143$)
964$ ASSIGN: Decide 39.NumberOut False=Decide 39.NumberOut False + 1:
NEXT(211$)

Model statements for module: Process 13

143$ ASSIGN: Process 11.NumberIn=Process 11.NumberIn + 1:
Process 11.WIP=Process 11.WIP+1
968$ QUEUE, Process 11.Queue
967$ SEIZE, 2, VA:
Resource 11, 1:
NEXT(966$)
966$ DELAY: Triangular(15, 16, 17), , VA
965$ RELEASE: Resource 11, 1
1013$ ASSIGN: Process 11.NumberOut=Process 11.NumberOut + 1:
Process 11.WIP=Process 11.WIP-1:
NEXT(145$)

Model statements for module: Decide 40

145$ BRANCH, 1:
With, 95/100, 146$, Yes:
Else, 211$, Yes

Model statements for module: ACCESS 29

211$ QUEUE, ACCESS ConST11ST12.Queue
ACCESS: ConST11ST12, 1:
NEXT(144$)

Model statements for module: Convey 30

144$ CONVEY: ConST11ST12, STATION 12

Model statements for module: Assign 13

146$ ASSIGN: OPERATION=OPERATION+1:
NEXT(211$)

Model statements for module: Enter 12

210$ STATION, STATION 12
1018$ DELAY: 0., , VA:
NEXT(1021$)
1021$ EXIT: ConST11ST12:
NEXT(147$)

Model statements for module: Decide 41

147$ BRANCH, 1:
If, OPERATION==11, 1029$, Yes:
Else, 1030$, Yes
1029$ ASSIGN: Decide 41.NumberOut True=Decide 41.NumberOut True + 1:
NEXT(148$)
1030$ ASSIGN: Decide 41.NumberOut False=Decide 41.NumberOut False + 1:
NEXT(213$)

Model statements for module: Process 14

148$ ASSIGN: Process 12.NumberIn=Process 12.NumberIn + 1:
Process 12.WIP=Process 12.WIP+1
1034$ QUEUE, Process 12.Queue
1033$ SEIZE, 2, VA:
Resource 12, 1:
NEXT(1032$)

1032$ DELAY: Triangular(13, 14, 15), , VA
1031$ RELEASE: Resource 12, 1
1079$ ASSIGN: Process 12.NumberOut=Process 12.NumberOut + 1:
Process 12.WIP=Process 12.WIP-1:
NEXT(150$)

Model statements for module: Decide 42

150$ BRANCH, 1:
With, 95/100, 151$, Yes:
Else, 213$, Yes

Model statements for module: ACCESS 30

213$ QUEUE, ACCESS ConST12ST13.Queue
ACCESS: ConST12ST13, 1:
NEXT(149$)

Model statements for module: Convey 31

149$ CONVEY: ConST12ST13, STATION 13

Model statements for module: Assign 14

151$ ASSIGN: OPERATION=OPERATION+1:
NEXT(213$)

Model statements for module: Enter 13

215$ STATION, STATION 13
1084$ DELAY: 0., , VA:
NEXT(1087$)
1087$ EXIT: ConST12ST13:
NEXT(152$)

Model statements for module: Decide 43

152$ BRANCH, 1:
If, OPERATION==12, 1095$, Yes:
Else, 1096$, Yes
1095$ ASSIGN: Decide 43.NumberOut True=Decide 43.NumberOut True + 1:

NEXT(153$)
1096$ ASSIGN: Decide 43.NumberOut False=Decide 43.NumberOut False + 1:
NEXT(216$)

Model statements for module: Process 15

153$ ASSIGN: Process 13.NumberIn=Process 13.NumberIn + 1:
Process 13.WIP=Process 13.WIP+1
1100$ QUEUE, Process 13.Queue
1099$ SEIZE, 2, VA:
Resource 13, 1:
NEXT(1098$)
1098$ DELAY: Triangular(11, 12, 13), , VA
1097$ RELEASE: Resource 13, 1
1145$ ASSIGN: Process 13.NumberOut=Process 13.NumberOut + 1:
Process 13.WIP=Process 13.WIP-1:
NEXT(155$)

Model statements for module: Decide 44

155$ BRANCH, 1:
With, 95/100, 156$, Yes:
Else, 216$, Yes

Model statements for module: ACCESS 31

216$ QUEUE, ACCESS ConST13ST14.Queue
ACCESS: ConST13ST14, 1:
NEXT(154$)

Model statements for module: Convey 32

154$ CONVEY: ConST13ST14, STATION 14

Model statements for module: Assign 15

156$ ASSIGN: OPERATION=OPERATION+1:
NEXT(216$)

Model statements for module: Enter 14

220$ STATION, STATION 14
1150$ DELAY: 0., , VA:
NEXT(1153$)
1153$ EXIT: ConST13ST14:
NEXT(157$)

Model statements for module: Decide 45

157$ BRANCH, 1:
If, OPERATION==13, 1161$, Yes:
Else, 1162$, Yes
1161$ ASSIGN: Decide 45.NumberOut True=Decide 45.NumberOut True + 1:
NEXT(158$)
1162$ ASSIGN: Decide 45.NumberOut False=Decide 45.NumberOut False + 1:
NEXT(218$)

Model statements for module: Process 16

158$ ASSIGN: Process 14.NumberIn=Process 14.NumberIn + 1:
Process 14.WIP=Process 14.WIP+1
1166$ QUEUE, Process 14.Queue
1165$ SEIZE, 2, VA:
Resource 14, 1:
NEXT(1164$)
1164$ DELAY: Triangular(10, 11, 12), , VA
1163$ RELEASE: Resource 14, 1
1211$ ASSIGN: Process 14.NumberOut=Process 14.NumberOut + 1:
Process 14.WIP=Process 14.WIP-1:
NEXT(160$)

Model statements for module: Decide 46

160$ BRANCH, 1:
With, 95/100, 161$, Yes:
Else, 218$, Yes

Model statements for module: ACCESS 32

218$ QUEUE, ACCESS ConST14N10.Queue
ACCESS: ConST14N10, 1:
NEXT(159$)

Model statements for module: Convey 33

159$ CONVEY: ConST14N10, NODE 10

Model statements for module: Assign 16

161$ ASSIGN: OPERATION=OPERATION+1:
NEXT(218$)

Model statements for module: Enter 15

221$ STATION, PACKING
1216$ DELAY: 0., , VA:
NEXT(1219$)
1219$ EXIT: Conveyor 16:
NEXT(162$)

Model statements for module: Decide 47

162$ BRANCH, 1:
If, OPERATION==14, 1227$, Yes:
Else, 1228$, Yes
1227$ ASSIGN: Decide 47.NumberOut True=Decide 47.NumberOut True + 1:
NEXT(163$)
1228$ ASSIGN: Decide 47.NumberOut False=Decide 47.NumberOut False + 1:
NEXT(222$)

Model statements for module: Process 17

163$ ASSIGN: PACKING 1.NumberIn=PACKING 1.NumberIn + 1:
PACKING 1.WIP=PACKING 1.WIP+1
1232$ QUEUE, PACKING 1.Queue
1231$ SEIZE, 2, VA:
Resource 15, 1:
NEXT(1230$)
1230$ DELAY: Triangular(9, 10, 11), , VA
1229$ RELEASE: Resource 15, 1
1277$ ASSIGN: PACKING 1.NumberOut=PACKING 1.NumberOut + 1:
PACKING 1.WIP=PACKING 1.WIP-1:
NEXT(165$)

Model statements for module: Decide 48

165$ BRANCH, 1:

With, 95/100, 166$, Yes:
Else, 222$, Yes

Model statements for module: ACCESS 33

222$ QUEUE, ACCESS ConPACEXIT.Queue
ACCESS: ConPACEXIT, 1:
NEXT(164$)

Model statements for module: Convey 34

164$ CONVEY: ConPACEXIT, EXIT1

Model statements for module: Assign 17

166$ ASSIGN: OPERATION=OPERATION+1:
NEXT(222$)

Model statements for module: Enter 16

224$ STATION, RLine
1282$ DELAY: 0., , VA:
NEXT(1285$)
1285$ EXIT: Conveyor 4:
NEXT(169$)

Model statements for module: Process 18

169$ ASSIGN: REPAIRMAN.NumberIn=REPAIRMAN.NumberIn + 1:
REPAIRMAN.WIP=REPAIRMAN.WIP+1
1296$ QUEUE, REPAIRMAN.Queue
1295$ SEIZE, 2, VA:
Resource 16, 1:
NEXT(1294$)
1294$ DELAY: Triangular(15, 20, 25), , VA
1293$ RELEASE: Resource 16, 1
1341$ ASSIGN: REPAIRMAN.NumberOut=REPAIRMAN.NumberOut + 1:
REPAIRMAN.WIP=REPAIRMAN.WIP-1:
NEXT(170$)

Model statements for module: Assign 18

170$ ASSIGN: OPERATION=OPERATION+1:
NEXT(225$)

Model statements for module: ACCESS 34

225$ QUEUE, ACCESS ConRLN3.Queue
ACCESS: ConRLN3, 1:
NEXT(171$)

Model statements for module: Convey 35

171$ CONVEY: ConRLN3, NODE 3

PROJECT, "ARI","MIRLIND BRUQI",,,No,Yes,Yes,Yes,No,No,No,No,No;

ATTRIBUTES: OPERATION;

VARIABLES: Process 4.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 7.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 14.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Process 12.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 41.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 7.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Decide 39.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 1.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 2.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Decide 17.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Process 11.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 13.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 45.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 23.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 31.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
DECIDE NODE 4.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
PACKING 1.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Decide 47.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Process 11.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):

Decide 27.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 9.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Decide 29.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Process 4.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
DECIDE NODE 4.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 6.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 33.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 9.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 14.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 43.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 25.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
PACKING 1.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 3.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Dispose 1.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 11.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 13.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Process 2.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 4.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 6.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 8.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 21.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
DECIDE NODE 5.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Process 6.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Process 1.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
PACKING 1.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 37.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 37.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 19.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Create 1.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
DECIDE NODE 5.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 43.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 10.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 21.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 12.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):

Process 14.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 8.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 33.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 19.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
DECIDE NODE 5P.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 10.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Process 8.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
DECIDE NODE 5P.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Process 3.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Process 2.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 47.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 25.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Process 5.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 10.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Process 13.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 31.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
REPAIRMAN.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 45.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 29.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 27.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Process 12.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
REPAIRMAN.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Process 5.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
Decide 41.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 23.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
Process 1.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 3.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 5.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 7.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Process 9.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 17.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude"):
Decide 39.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
REPAIRMAN.NumberOut,CLEAR(Statistics),CATEGORY("Exclude");

QUEUES: Access EntryNode1.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node8 Node9.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 9.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node5Long.Queue,FIFO,,AUTOSTATS(Yes,,):
Con ST7ST8.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 11.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node2RLine.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node5Node6.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node10Node1.Queue,FIFO,,AUTOSTATS(Yes,,):
Access ConST13ST14.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node6Long.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 5.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node3Node8.Queue,FIFO,,AUTOSTATS(Yes,,):
Access ConST12ST13.Queue,FIFO,,AUTOSTATS(Yes,,):
PACKING 1.Queue,FIFO,,AUTOSTATS(Yes,,):
Con ST3ST4.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 12.Queue,FIFO,,AUTOSTATS(Yes,,):
Access ConPACEXIT.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 6.Queue,FIFO,,AUTOSTATS(Yes,,):
Con ST6ST7.Queue,FIFO,,AUTOSTATS(Yes,,):
Access ConST11ST12.Queue,FIFO,,AUTOSTATS(Yes,,):
Access ConST14N10.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node9Node10.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 1.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node7Node8.Queue,FIFO,,AUTOSTATS(Yes,,):
Access ConRLN3.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 13.Queue,FIFO,,AUTOSTATS(Yes,,):
Access ConST10ST11.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 7.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node6Node7.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node2Node3.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node9Node2.Queue,FIFO,,AUTOSTATS(Yes,,):
Con ST2ST3.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 2.Queue,FIFO,,AUTOSTATS(Yes,,):
Con ST9N7.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 14.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node10EXIT.Queue,FIFO,,AUTOSTATS(Yes,,):
ConST5ST6.Queue,FIFO,,AUTOSTATS(Yes,,):
Con ST8ST9.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node1Node2.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node5.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 3.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 8.Queue,FIFO,,AUTOSTATS(Yes,,):
Con ST1ST2.Queue,FIFO,,AUTOSTATS(Yes,,):
REPAIRMAN.Queue,FIFO,,AUTOSTATS(Yes,,):
Con ST4N5.Queue,FIFO,,AUTOSTATS(Yes,,):
Access Node3Node4.Queue,FIFO,,AUTOSTATS(Yes,,):
Process 10.Queue,FIFO,,AUTOSTATS(Yes,,):

Process 4.Queue,FIFO,,AUTOSTATS(Yes,,);

PICTURES: Picture.Airplane:
Picture.Green Ball:
Picture.Blue Page:
Picture.Telephone:
Picture.Blue Ball:
Picture.Yellow Page:
Picture.EMail:
Picture.Yellow Ball:

RESOURCES: Resource
1,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
2,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
3,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
4,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
5,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
6,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
7,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
8,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
9,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
10,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
11,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
12,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
13,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
14,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
15,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,):
Resource
16,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(Yes,,);

STATIONS: NODE 1,,,NODE 1,AUTOSTATS(Yes,,):
NODE 2,,,NODE 2,AUTOSTATS(Yes,,):
NODE 3,,,NODE 3,AUTOSTATS(Yes,,):
NODE 4,,,NODE 4,AUTOSTATS(Yes,,):

```
                NODE 5,,,NODE 5,AUTOSTATS(Yes,,):
                NODE 6,,,NODE 6,AUTOSTATS(Yes,,):
                NODE 7,,,NODE 7,AUTOSTATS(Yes,,):
                NODE 8,,,NODE 8,AUTOSTATS(Yes,,):
                NODE 9,,,NODE 9,AUTOSTATS(Yes,,):
                PACKING,,,PACKING,AUTOSTATS(Yes,,):
                ENTRY,,,ENTRY,AUTOSTATS(Yes,,):
                2ENDSHIELD,,,2ENDSHIELD,AUTOSTATS(Yes,,):
                STATION 02,,,STATION 02,AUTOSTATS(Yes,,):
                STATION 03,,,STATION 03,AUTOSTATS(Yes,,):
                STATION 04,,,STATION 04,AUTOSTATS(Yes,,):
                STATION 06,,,STATION 06,AUTOSTATS(Yes,,):
                STATION 07,,,STATION 07,AUTOSTATS(Yes,,):
                STATION 08,,,STATION 08,AUTOSTATS(Yes,,):
                STATION 09,,,STATION 09,AUTOSTATS(Yes,,):
                STATION 10,,,STATION 10,AUTOSTATS(Yes,,):
                STATION 11,,,STATION 11,AUTOSTATS(Yes,,):
                STATION 12,,,STATION 12,AUTOSTATS(Yes,,):
                STATION 13,,,STATION 13,AUTOSTATS(Yes,,):
                STATION 14,,,STATION 14,AUTOSTATS(Yes,,):
                EXIT1,,,EXIT1,AUTOSTATS(Yes,,):
                RLine,,,RLine,AUTOSTATS(Yes,,):
                NODE 10,,,NODE 10,AUTOSTATS(Yes,,):
                1ENDSHIELD,,,1ENDSHIELD,AUTOSTATS(Yes,,);

SEGMENTS:       Conveyor 5.Segment,NODE 3,NODE 4-3:
                Conveyor 4.Segment,NODE 2,RLine-12:
                Conveyor 13.Segment,NODE 8,NODE 9-1:
                ConST1ST2.Segment,1ENDSHIELD,STATION 02-6:
                Conveyor 12.Segment,NODE 7,NODE 8-1:
                Conveyor 3.Segment,NODE 2,NODE 3-1:
                ConST3ST4.Segment,STATION 03,STATION 04-6:
                Conveyor 2.Segment,NODE 1,NODE 2-4:
                ConST5ST6.Segment,2ENDSHIELD,STATION 06-10:
                Conveyor 11.Segment,NODE 6,NODE 7-1:
                ConST7ST8.Segment,STATION 07,STATION 08-10:
                Conveyor 10.Segment,NODE 6,2ENDSHIELD-10:
                ConST13ST14.Segment,STATION 13,STATION 14-9:
                Conveyor 17.Segment,NODE 10,NODE 1-1:
                Conveyor 9.Segment,NODE 5,NODE 6-1:
                ConST12ST13.Segment,STATION 12,STATION 13-9:
                ConST14N10.Segment,STATION 14,NODE 10-9:
                Conveyor 1.Segment,ENTRY,NODE 1-14:
                ConRLN3.Segment,RLine,NODE 3-12:
                ConST11ST12.Segment,STATION 11,STATION 12-9:
                Conveyor 8.Segment,NODE 4,NODE 5-1:
                Conveyor 16.Segment,NODE 10,PACKING-7:
                ConST4N5.Segment,STATION 04,NODE 5-6:
```

ConPACEXIT.Segment,PACKING,EXIT1-7:
ConST10ST11.Segment,STATION 10,STATION 11-9:
ConST9N7.Segment,STATION 09,NODE 7-10:
Conveyor 7.Segment,NODE 4,1ENDSHIELD-6:
ConST2ST3.Segment,STATION 02,STATION 03-6:
Conveyor 6.Segment,NODE 3,NODE 8-3:
Conveyor 15.Segment,NODE 9,NODE 2-1:
Conveyor 14.Segment,NODE 9,STATION 10-9:
ConST6ST7.Segment,STATION 06,STATION 07-10:
ConST8ST9.Segment,STATION 08,STATION 09-10;

CONVEYORS: ConST10ST11,ConST10ST11.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
Conveyor 10,Conveyor 10.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
Conveyor 11,Conveyor 11.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
Conveyor 12,Conveyor 12.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
Conveyor 13,Conveyor 13.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
Conveyor 14,Conveyor 14.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
Conveyor 15,Conveyor 15.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
Conveyor 16,Conveyor 16.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
Conveyor 17,Conveyor 17.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConST9N7,ConST9N7.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConST12ST13,ConST12ST13.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConST4N5,ConST4N5.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConST1ST2,ConST1ST2.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConST2ST3,ConST2ST3.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConST3ST4,ConST3ST4.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConPACEXIT,ConPACEXIT.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConST5ST6,ConST5ST6.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):
ConST6ST7,ConST6ST7.Segment,1,1,Active,1,Non-Accumulating,,AUTOSTATS(Yes,,):

```
                ConST11ST12,ConST11ST12.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                ConST7ST8,ConST7ST8.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                ConST8ST9,ConST8ST9.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                ConRLN3,ConRLN3.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 1,Conveyor 1.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 2,Conveyor 2.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 3,Conveyor 3.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 4,Conveyor 4.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 5,Conveyor 5.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 6,Conveyor 6.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 7,Conveyor 7.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 8,Conveyor 8.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                Conveyor 9,Conveyor 9.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                ConST13ST14,ConST13ST14.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,):
                ConST14N10,ConST14N10.Segment,1,1,Active,1,Non-
Accumulating,,AUTOSTATS(Yes,,);

REPLICATE,
1,,SecondsToBaseTime(57600),Yes,Yes,SecondsToBaseTime(250),,,24,Seconds,No,N
o,,DATETIME("Sep 12, 2003 10:14:01"),
                No;

ENTITIES:     Entity 1,Picture.Blue Ball,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(Yes,,);

ACTIVITYAREAS: NODE 1,0,,AUTOSTATS(Yes,,):
                        NODE 2,0,,AUTOSTATS(Yes,,):
                        NODE 3,0,,AUTOSTATS(Yes,,):
                        NODE 4,0,,AUTOSTATS(Yes,,):
                        NODE 5,0,,AUTOSTATS(Yes,,):
                        NODE 6,0,,AUTOSTATS(Yes,,):
                        NODE 7,0,,AUTOSTATS(Yes,,):
                        NODE 8,0,,AUTOSTATS(Yes,,):
                        NODE 9,0,,AUTOSTATS(Yes,,):
                        PACKING,0,,AUTOSTATS(Yes,,):
```

ENTRY,0,,AUTOSTATS(Yes,,):
2ENDSHIELD,0,,AUTOSTATS(Yes,,):
STATION 02,0,,AUTOSTATS(Yes,,):
STATION 03,0,,AUTOSTATS(Yes,,):
STATION 04,0,,AUTOSTATS(Yes,,):
STATION 06,0,,AUTOSTATS(Yes,,):
STATION 07,0,,AUTOSTATS(Yes,,):
STATION 08,0,,AUTOSTATS(Yes,,):
STATION 09,0,,AUTOSTATS(Yes,,):
STATION 10,0,,AUTOSTATS(Yes,,):
STATION 11,0,,AUTOSTATS(Yes,,):
STATION 12,0,,AUTOSTATS(Yes,,):
STATION 13,0,,AUTOSTATS(Yes,,):
STATION 14,0,,AUTOSTATS(Yes,,):
EXIT1,0,,AUTOSTATS(Yes,,):
RLine,0,,AUTOSTATS(Yes,,):
NODE 10,0,,AUTOSTATS(Yes,,):
1ENDSHIELD,0,,AUTOSTATS(Yes,,);

UNIVERSITETI I PRISHTINËS
FAKULTETI I INXHINIERISË MEKANIKE
PRISHTINË

Mr. Sc. Mirlind Bruçi , inxhi. i dipl.

Optimalizimi i proceseve prodhuese në linjat fleksibile asembluese të reparteve të elektromotorëve në fabrikën ATB-Wiena

-PUNIM I DOKTORATURËS-

Prishtinë, 2005