Master Thesis

A holistic machining line behavior modeling using Finite State Machines

Shiva Abdoli
Superviser: Dr.DanielTesfamariamSemere

December 2013

Contents

Abstract

Energy consumption of turning and milling operations are analyzed in this thesis profoundly. It is aimed to be able to employ analysis results in real production floors. So it is tried to investigate on most effective parameters on machine tool energy consumptions which are changeable in production floors. Due to existing limitations in predesigned manufacturing lines, cutting depth, feed rate and spindle speed, are chosen to analyze their effects on machine tools energy consumption. All other influencing parameters are presumed constant during research. Evaluating machine tools energy consumption shows increasing machining factors values reduces energy consumption in machining operation. Scrutinizing machining factors effect on energy consumption revealed that, increasing one machining factor when two other factors have constant low or constant high values has different effect on energy consumption.

The main contribution of this research is proposing a mathematical model, based on material removal rate and machining time for estimating machine tools energy consumption. In addition, a methodology to find machine energy consumption profile based on MRR in a particular operation is proposed too. This enables to find critical breakpoints of MRR for energy consumption in machining operations.

Subsidiary effect of increasing Machining factors, on machine energy consumption is analyzed too. To obtain integrative conclusion regarding the effect of machining factors on energy consumption, their influence should be studied in a production system, for long term. In addition machines experience different states with different profile of energy consumption. So energy consumption of machine tools in all states is considered as product associated energy consumption. These targets are achieved by modeling a production line and simulate it for long time. The results indicates for system energy efficiency, it worth's to increase machining factors even if tool life and consequently machine utilization reduce.

Effect of production planning such as batch mode production from energy consumption perspective in production system is evaluated. The results exhibit consistency between tool life, machine idle energy consumption and optimum batch size. The accomplishments can greatly help process planner to achieve optimum production system configuration to enhance energy efficiency.

Acknowledgement

Foremost, I would like to express my sincere gratitude to my supervisor **Dr.Daniel Tesfamariam Semere** for continuous support during my thesis research, his patience, motivation, enthusiasm, and immense knowledge.

I would like to express especial thank and love to my beloved mom and dad, you are all the reasons for me.

Last but not the least, I would like to thank my husband, your patience and love always encourage me in this way.

List of figures, tables and acronyms

Figures

Tables

Acronyms

MRR: Material removal rate

FSM: finite state machine

C_{pk}: Process capability index

MTEC: Machine Tools Energy Consumption

Chapter One: Introduction

Need for research

Global warming dilemma is intensifying nowadays, and corresponding motifs like reducing carbon dioxide emissions has been arising attention in industry. There is an unequivocal link between carbon dioxide and energy consumption. So, manufacturing industries, as huge consumer of electrical energy, should contemplate modern techniques to boost energy efficiency (1), (2).

Common goals in most production systems are increasing output, cost reduction and quality improvement (3). Energy efficient production not only helps sustainability but also benefits on cost reduction. This increases energy efficiency attitudes in manufacturing systems (4).

Realizing machines energy usage profiles and influencing parameters are the preliminary steps toward reducing machine tools energy consumption (4) (5).

Research problem

The existing models for estimating machine energy consumption mainly cover metal cutting state of one single unit. In the other hand, they mostly depend to a lot of variables. This increase model complexity and reduce their applicability. This research is motivated to accomplish an accurate mathematical model to appraise machine tools energy consumption (MTEC) based on machining factors which is also simple enough to be applicable in production floors.

In most research publications about machining operation energy consumption, the effect of various input variables has been studied on one machining operation on one single unit. But usually production plans are long term, and the effect of selecting process parameters should be investigated on mass production. In order to get comprehensive perspective in energy consumption in real production process, it is better to investigate on sequence of several types of machining operations. In this thesis a real machining process to attain a product has been studied and modeled. Each simulation run on proposed model, with different input variables is called experiment. So, it can help production planner to find the best process and system parameters configuration by comparing different experiments results.

Each particular system has its own goals which may differ from similar system. This framework can be adjusted to provide appropriate outputs. This helps to select best input parameter variables for production system, based on their prioritized goals.

Basic terms and concepts

Considering that production lines have input, output and some policies, they can be considered as system. Systems demonstrated different behavior patterns which categorizes them in different classifications. There are two typical types of systems, discrete event systems and continuous systems. *Discrete event system; 'they have discrete-states and are event-driven and the transitions between states depend on the occurrence of asynchronous discrete events over time'* (6). *Continuous systems;* inputs and outputs of these systems are capable of changing at any time instant (7). There are some systems which have both characters of discrete event systems and continuous systems, and it is hard to associate one specific type of systems to them. These systems are called hybrid systems. Production lines have some specific event and hierarchy of events like machining time. This illustrates the discrete event systems properties. But in the meantime, some event can take place in any time like failures during machining based on some stochastic functions. So it expresses some characters of continuous systems too. As a result we should consider the production line as a hybrid system, and model it with this assumption.

Different landscape of production goals, makes finding the best production parameters and configuration, so complicated. It is also not reasonable to find the optimum production specification by try and error in real production process. So modeling the production process and simulate its dynamic behavior can overcome this problem. Models show an abstraction of real world. So it may cover some limited aspects of real object, but it should contain main specifications. This enables to map model result in real world. Thus modeling production system benefits to reduce real experiments cost and time span to get the result. In complex problems like here with multiple objects, simulation modeling instead of analytical modeling is needed.

Finite State Machine (FSM) models the demeanor of an object. It can confront different states by transitions between states. State transition happens by satisfying some conditions like receiving some input variables or can be triggered by an event. Set of states and transitions should be finite (8), (9).

State flow is widely used in hybrid systems modeling to define a discrete controller in the model. The continuous dynamics of the model are specified by using stochastic or differential equations (10). A State flow diagram is a graphical representation of a finite state machine (11). State flow can perfectly provide the interaction of continuous and discrete characteristics of hybrid systems. It mostly uses flowcharts to control the logic of a reactive system. Each state in a flowchart represents a mode of an event-driven system. Making the decision about stay in or leave from a state, investigated through the conditions. The continuous part of hybrid system can be the evaluation criteria for transition conditions. The flow charts can contain hierarchy and parallelism of states. Appropriate actions (like updating variables, calling functions, triggering events), can be defined based on the current active state of flow chart or just satisfied transition condition to move to another state. Transition actions can happen just after condition satisfaction, or after reaching the new state. The state flow in Matlab, provides plenty of options which makes the designing of flowcharts more easier and efficient, which is utilized in this thesis.

Limitation and delimitation of the study

Several parameters can affect machine tools energy consumption. Some parameters like work piece material and machine type are not changeable in production floors. They mostly depend on process limitations and customer order and the selection decision regularly are made before production planning. In the other hand in modeling perspective, increasing input variables not only increase model complexity but also reduce output accuracy. So to get better result; it is recommended to keep constant some parameters. In this thesis, it is tried to study deeply on possible changeable and measurable process parameters in production floors. As a result the investigating parameters are limited to machining factors which are spindle speed, feed and cutting depth. It is aimed to not only achieve accurate academic analysis and models, but also provide an applicable decision making framework for production floors.

Utilizing outcomes of real production line like machine energy consumption, failures and repairs times for modeling, increase model reliability. Due to all limitations to access real production line, we have to take relevant data from previous research which had been collected data from real production floors. The important issue is the research approach which won’t be affected with different input variables.

Different machining operations have their own characteristics and influencing parameters but it is not possible to study all types here. Milling and turning operations which are most popular machining operations in industry, have been selected to be deeply investigated in this thesis.

Chapter two: Machining process characteristics

Discuss and formulate machining process parameters

Before analyzing machining factors effect on energy consumption, it is better to review some basic machining process concepts and formulas. These terms directly influence on machining operation and process parameters (12), (13), (14) (15). To avoid confusion they are described for turning and milling separately.

Turning

Machining time =$\frac{l}{f\times N}$

l: Cutting length

f: feed

N : Spindle rotational speed

Material removal rate= $\pi\times D_{avg}\times f \times N$

D_{avg}: Average diameter of work piece

$D_{avg} = (D_{initia} + D_{final})/2$

Material removal rate (MRR) is the volume of material which will be removed per minute from work piece with the cutting tool, it mostly expresses with mm^3/min.

Milling

Machining time= $\frac{(l+2l_c)}{v}$

L: length of cut

ν :Linear speed of the work piece or feed rate

l_c=extent of the cutter's first contact with work piece which is mostly considered as half of cutter diameter

Material removal rate = $wd\,v$

W: width of cut

d: cutting depth

Tool life

Tool life is considered as the useful and possible time to perform machining operation with a cutting tool. Tool wearing is the most popular failure in cutting tools (14) (16). If worn tool is used in machining operation, it not only reduces surface quality, but also increases other failures occurrence probability like tool breakage. So recognizing tool useful life and employing it just in its specified interval is a critical issue in production systems (14), (15) (16). A lot of parameters can influence on tool life, but generally it can be calculated based on Taylor formula (16), (4).

Tool life=$\mathrm{V_c} * \mathrm{T^n} * \mathrm{D^x} * \mathrm{f^y} = \mathrm{C}$

f: feed rate

V: cutting speed,

d: cutting depth

C and *n*: work piece and cutting tool material constant

x, *y* depends on machining operation type and should be determined experimentally

In production floors, tool life mostly represented as the number of possible cutting operations with same tool. This term, can be regulated by dividing total tool life time to operation time (14). Number of used tools in production is another aspect of production cost associated to product.

Machine states

Machine tool faces 2 basic states; idle and busy, while there are several sub states in busy state. Considering the importance of machine busy state, and to obtain better understanding of machine tool demeanor, all sub states will be explained briefly (17) (18) (5).

Clamping: represents machine mode when clamping device start holding work piece. It refers to the changing clamping device position from un-hold to hold.

Rapid move: represents machine mode when cutting tool has a rapid motion from home position to the work piece proximity.

Spindle acceleration: represents machine mode when spindle starts rotation until reaches determined speed.

Air cutting from home position to work piece: represents machine mode when cutting tool is approaching to work piece until contact it, while spindle is rotating. This state is totally the same with machining state (cutting), but here cutting tool has not contacted the work piece yet.

Metal Machining or machining state: represents machine mode when is performing material removal operation on work piece. As soon as tool advanced to work piece, machining operation starts.

Air cutting from work piece to home position: represents machine mode when material removing operation is finished and tool slowly recedes the work piece and returns to home position. It is similar to first air cutting. When cutting tool reaches home position, the spindle motor will be turned off.

De-clamping: represents machine mode when clamping device changes its position from hold to un-hold.

Failures can occur during each state of machining in different machine sections. Machine transits from busy to fail state until repair action finishes, and then transits to idle state again. It is assumed that machine will be turned off when failure happened, and switched on as soon as repair action is finished.

Process capability

Production process achieve final product by utilizing, machines, materials, staff and all other entities involved in process. The process should produce product within predefined specifications limits. Process ability to meet product specifications mostly is evaluated by statistical methods and the most popular statistical index in this area is called process capabilityC_{pk}. It clarifies the ability of under control process to perform within specification limits, by considering the inherit variability of the process. This index predicts how many parts may be produced out of specification limits (19), (20), (21).

Chapter Three: machine energy consumption characteristics

Analysis machine energy consumption profile

Variable & fix energy consumption

Machining energy usage profiles are mostly divided to fix and variable energy consumption. Fix energy provides machine energy requirements for operational readiness and running basic equipment features like power to start up the computer or fans. It basically depends to machine design and not affected by machining factors (17).

Variable energy consists of operational and machining energy consumption which is the required power to assure specific machining operation. This energy enables axis movement or spindle rotation, to perform air cutting operation. Machining (tool tip) energy is the power demand through material removal operation during machining process and changes by different machining factors (17).

Machine states energy consumption

As explained earlier, machine tools encounter different states in production process with different energy consumption profile. To get better realization of machine energy consumption, energy characterization of all states will be explained here briefly (17) (18) (22) (23) (4) (24) (25) (26) (5) (27)

Idle state: machine tool consumes specific and fix amount of energy during standby state, since some peripheral units are on. This energy consumption depends on machine type. The most influencing solution to reduce machine fix energy consumption is changing machine design. For instance by inactivating unnecessary features, fix energy consumption will be reduced very considerably. But it is mostly the designer job, and is not practically easy to do it in production floors. Reducing machine unproductive time contributes to reduce machine total idle energy usage and can be achieved by production plan optimization. It should be mentioned that, fix value of energy consumption will exists in all other machine states too.

Start clamping: Machine demands the peak value of power due to momentum effect since sudden change of clamping device happens to hold work piece.

Rapid move: Energy consumption of this state depends on machine energy consumption for moving each axis and the traveling path between home position and work piece place. Energy consumption can be reduced by shortening tool travelling path in this state.

Spindle acceleration: machine consumes a peak value of energy by sudden change of spindle motor due to momentum effect. Spindle motor type is the major influencing parameter on this mutation of energy consumption.

Air cutting: machine demands power in this state to move axis and rotate spindle. Spindle motor type and energy consumption due to axis movement are most influencing parameters in energy consumption in this state. Reducing tool traveling path in air cutting states can also help to reduce this state energy consumption.

Machining or metal cutting state: Machine demands power to remove material from work piece. This contributes the biggest share of energy consumption in whole machining cycle. Influencing parameters on energy consumption in this state will be investigated and analyzed deeply in next section.

De clamping: the energy profile of this state is totally similar with clamping action. It refers to sudden change of clamping device to release work piece. Clamping system, consumes energy in a moderate and steady mode from clamping to de clamping action for holding work piece. The practical solution to reduce this energy in production floor can be minimizing machine cycle time.

Dry and wet cutting energy concerns

Machine energy consumption in dry or wet conditions differs even all other influencing parameters are the same. Cooling operations is a considerable source of energy consumption in different forms, like cooling pump electricity consumption or cooling material usage (24) (28) (4). According to main concern of this research which is energy efficiency, dry machining operations are studied in this paper.

Analysis process parameter effect on machine energy consumption

As mentioned earlier, some influencing parameters on machine energy consumption are not changeable when dealing with existing production lines like, machine type. Cutting depth, feed rate and spindle speed are the machining variable factors which can be adjusted easier in production floors. They also determine main process characteristics like, machining time and tool life. In the other hand, by changing process parameters like tool life, they can permute machine utilization and alter machine total idle energy consumption. So, machining factors have dual effect in fix and variable energy consumption. So, in this research, the effect of machining factors will be investigated on machine energy consumption.

As explained earlier, due to all limitation it was hard to measure energy consumption in this research directly from machine tools. By studying pervious research, S.Kara presented models to estimate MTEC in turning and milling operations. The models appraise energy consumption with a good approximation (2). They are based on cutting volume and MRR in machining operation. Cutting volume is independent from cutting conditions. So it is not easy to analyze MTEC based on introduced formula. In addition, finding cutting volume may be hard in some cases, especially if there is sequence of operation in one unit. This research tries to develop an accurate model which not only gives better understanding of MTEC, but also be more applicable in production floors. So the basic model from S.kara paper is utilized to find machine energy consumption values, with different possible machining factors. Obtained energy values have been used for machine tool energy consumption analysis.

Hypothetical milling and turning operations are designed to be studied. Milling and turning are the basic operations of gear manufacturing process. Work piece material assumed to be a cold-rolled steel bar. Turning operation performs machining on work piece to reach required diameter. Process continues by milling operation to grove it. Cutting tool in milling and turning are respectively, solid-uncoated cobalt alloyed HSS and TiN coated tungsten carbide triangular inserts. Mori Seiki Dura Vertical5100 is considered for milling and Colchester Tornado A50 for turning. All tested machining factors are selected from practically possible spectrum for each factor, considering cutting tool and work piece material.

Since it is aimed to find the effect of machining factor on machine energy consumption, other parameters like, tool path in air cuttings and rapid move actions are kept constant in different machining scenarios. Milling and turning are analyzed separately in next following sections.

Turning

Machining factors effect analysis on machining state energy consumption

Energy consumption formulation

Table 1 shows several turning scenarios with different combination of cutting depth, feed and spindle speed. Explained formulas and relations in chapter two are used to find tool life, MRR and machining time. Taylor equation is adjusted by considering hypothetical tool and work piece material for this operation as below (15):

$$V_c * T^{0.2} D^{0.2} S^{0.15} = 273$$

In table 1, Cutting depth scale is in mm, feed rate in mm per revolution, spindle speed in revolution per minute, MRR in cm^3/min, tool life and machining time in minute and Energy is in Mega Joule.

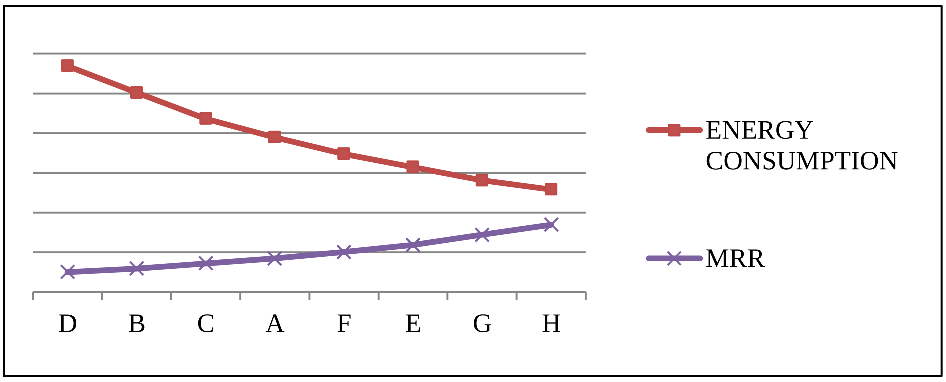
Since number of operation with same tool is more popular in production floors, from now where ever tool life is mentioned, it refers to number of machining operations with same tool.

Table 1. Different turning machining factors combination and result on energy consumption

Scenarios	Cutting depth	Feed rate	Spindle speed	MRR	Tool life	Machining Time	Number of operation with one tool(tool life)	Energy consumption
A	1	0.2	500	42.4	11.9	2	5	194.76
B	1	0.14	500	29.7	15.5	2.86	5	250.96
C	1	0.2	425	36	26.7	2.35	11	218.12
D	1	0.14	425	25.2	34.9	3.36	10	284.47
E	2	0.14	500	59.3	7.74	1.43	5	157.3
F	2	0.14	425	50.4	17.5	1.68	10	173.9
G	2	0.2	425	72.1	13.4	1.18	11	140.62
H	2	0.2	500	84.8	5.93	1	5	129.05

MRR represents all machining factors simultaneously, which are the only variables in these cutting scenarios. So, MRR is utilized to analysis machine energy consumption. Figure 1 show MRR and related energy consumption in different scenarios.

Figure1.MRR effect on turning operation energy consumption



The figure exhibits increasing MRR, decreases machine energy consumption. In graph outset, small increase in MRR leads to considerable reduction in energy consumption. Energy consumption data seri expresses a reduction trend. But the slope of energy reduction by increasing MRR increases continually. In supreme section, considerable increase in MRR does not reduce energy consumption noticeably. It expresses if MRR has a substantial value, reducing one of its constitutive factor won't have a considerable impact on energy consumption, while other remaining factors have high values. In the other hand if two of 3 factors of MRR should have low values, increasing the remaining pillar to a high value, reduces energy consumption considerably.

As mentioned earlier, considering the complexity of existing models for estimating MTEC, this research is motivated to propose an accurate but practical formula for MTEC estimation. In first chapter, it is explained that, machining time depends on MRR or machining factors. So this research, propose a formula depends on machining time and MRR which are quiet intelligible in production floors.

Table 2. Regression equations for turning operation energy consumption using MRR and machining time

Regression Function	Correlation Coefficient
Y=(0,743x+65,903)*t	0,999961
Y=(-0.00001x^2+0,7446x+65,865)*t	0.999963
Y=(0.00002x^3-0,0028x^2+0,8867x+63,691)*t	0.999937
Y=(0.000001x^4-0.0003x^3+0.0203x^2 +0,1338x+ 72,28)*t	0.9925
Y=(-0.000000005x^5+0.000003x^4-0.0004x^3 +0.0271x^2 – 0.0278x+73.74)×t	0,9899
Y=(27.08$x^{0.346}$)×t	0.9966

Using regression analysis, several estimating functions for energy consumption have developed and are showed in table 2.

Correlation coefficient (R square) in regression analysis determines goodness of linear regression model in fitting data set. This can be employed just in linear relation. But it wrongly has been used in widespread nonlinear relation regressions. (29)

$$r^2 = 1 - \frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

y_i refers to experimental data and $\hat{y}_i$ is estimated data with regression model.

To be able to analysis other nonlinear regression models, modified r square for nonlinear relations has been used.

$$r_{nl}^2 = \frac{\left[\sum_{i=1}^{n} (y_i - \bar{y})(z_i - \bar{z})\right]^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2 \sum_{i=1}^{n} (z_i - \bar{z})^2},$$

This factor assumes if the nonlinear model is a good fit, the experimental data and estimated data with model (z_i) should show a good linear relation. This will be measured with above equation.

Second degree equation correlation coefficient is bigger than others. So it indicates better ability for estimation. The final equation for machine tool energy consumption can be express as below:

Machine tool energy consumption= (-0.00001MRR^2+0.7446MRR+65.865) × Machining time

If MTEC should have been appraised out of experimented range, depending high degree equations to more coefficients, may reduce estimation accuracy. In addition since linear equation R square has a small difference with second degree; it can be employed for MTEC estimation.

The proposed equation just depends on process parameters which are comprehensible in production floors and no need to calculate or find any extra complicated variables or constants. It also enables monitoring energy consumption during machining time, not at the end of machining operation since its time dependent. So this equation can easily be used in production floors to estimate MTEC.

Methodology for analyzing energy consumption profile

To obtain a deep understanding of MTEC relation with machining factors, equations just based on MRR are needed. This paper introduces an academic methodology to answer this requirement. It also can be easily employed in industry. The methodology includes five main steps. This methodology is applicable when, particular operation is investigated.

Step1: MTEC with different MRR should be estimated. Final equations based on MRR and time in pervious section will be used for MTEC estimation.

Step2: Several regression functions based on MRR should be regulated. These equations can be used for MTEC when the cutting volume is determined. But proposed equations in pervious section are general and can estimate MTEC for and turning operation in different cases. Table 3 proposes, estimating functions for these specific turning operations.

Table 3. Regression equations for turning operation energy consumption

Regression Function	Correlation Coefficient
Y=-2.4561x+316,42	0.88
Y=0.048x^2-7,7555x+440,94	0.98
Y=-0,001x^3+0.2078x^2-15,778x+563,7	0.99
Y=0.00002x^4-0.0053x^3+0,547x^2-26,802x+ 689,45	0.99
Y=-0.0000007x^5+0. 0001x^4-0.0182x^3+ 1,1844x^2 – 41,921x+826,06	0.03846
Y=2281,8$x^{-0,652}$	0,996

Step3: the best estimating function should be selected. Comparing correlation coefficients, the best fit functions for energy consumption in these specific turning operations is power equation.

Step4: second derivative equation of selected equation (power equation here) should be regulated and solved to find its roots. Recall from figure 1, energy consumption slope was negative but was increasing continuously. The acceptable root refers the point, which graph may have positive slope. In this point, by increasing MRR, energy consumption will augment. Obviously derivative equation of this power equation doesn't have an acceptable root. All equations in table 3 are estimator functions with some estimation error. Forth degree equation has biggest R square after power equation. So, to make sure about MTEC profile, forth degree equation is analyzed here too.

Y''=0, 000024$\boldsymbol{x^2}$-0, 0318$\boldsymbol{x}$ +1,094=0

This equation doesn't have an acceptable root too. It apparently means MRR increment never increases MTEC in this specific operation.

Step4: first derivative equations of selected equations (power equations) should be regulated. Root represents a breakpoint value for MRR. By increasing MRR to this value, energy consumption won’t decrease and remains constant.

Y'=0, 00008$\boldsymbol{x}^{3}$ -0, 0159$\boldsymbol{x}^{2}$ +1,094 $\boldsymbol{x}$ - 26,802=0

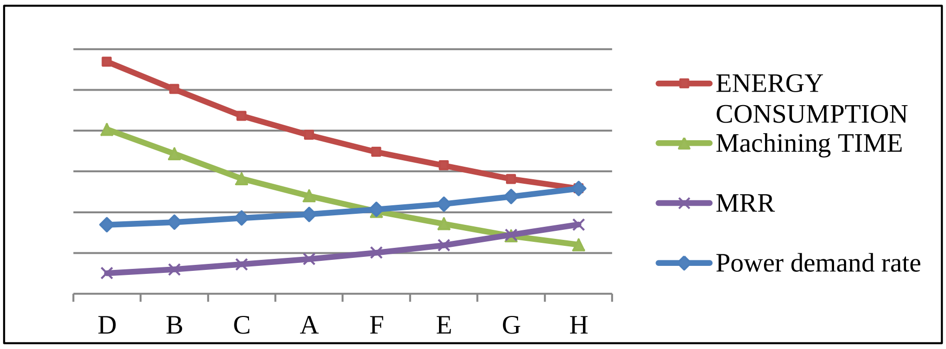
In turning case, first derivative equation does not have an acceptable root. It means increasing MRR always reduces MTEC. Recall, proposed methodology here is applicable only when analyzing a particular case.

Utilizing estimating MTEC models (based on MRR and time), integrated with proposed methodology, can greatly help process planner to realize MTEC. They not only can estimate MTEC in different cases, but also will achieve a deep understanding of MTEC profile in a specific operation.

Machine power demand profile

Figure 2 displays machining time and related energy consumption in different scenarios.

Figure2.MRR interaction with machining time and energy consumption in turning operation



Cutting depth, feed rate and spindle speed constitute MRR and directly affect machining time. Figure 2 illustrate clear trends in MRR, machining time, power demand rate and MTEC. They indicate that, by increasing machining parameters, power demand rate increases in machining operation. But reduction in machining time, not only covers excessive power demand, but also reduces energy consumption in machining state. In other word to reduce energy consumption, machining parameter combination should decrease machining time.

Machining factor subsidiary effect on machine energy consumption

Based on analysis so far, due to reducing energy consumption in machining operation, MRR should be increased. But the subsidiary effect of MRR increment on machine energy consumption should be investigated too. As mentioned earlier, tool life is an important parameter in machining operation, since it determines number of tool replacement actions in production line. Figure 3 shows the relation between MRR and tool life. Reducing number of used tools in machining operations is another considerable target in production system. Beside tool costs, reducing number of used tools will reduce tool replacement time, which consequently increases machine available time for production.

Figure3.MRR interaction with tool life in turning operation

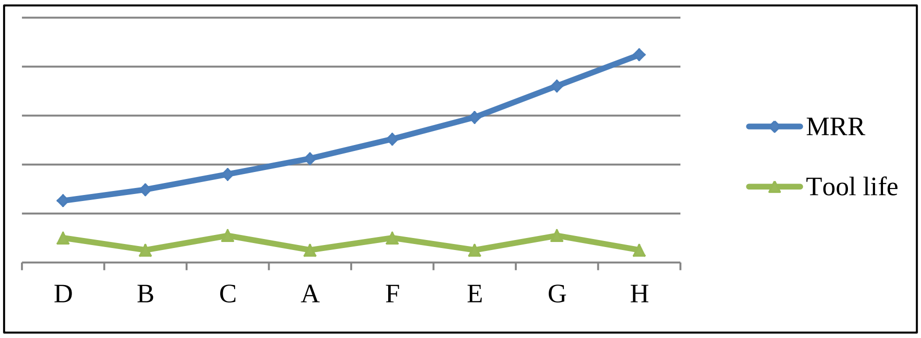


Figure3 shows that, tool life fluctuates by increasing MRR. No explicit relation between MRR or energy consumption and tool life can be determined. MRR represents 3 machining factors simultaneously and each has different contribution on tool life Taylor equation. This clearly means same value for MRR with different values for each factor, generate different tool life. To get a deep understanding of machining factors influence on machine energy consumption, they are studied separately in next section.

Machining factors extreme analysis

To find singular effect of each machining factor on tool life, two categories of analytical experiment are arranged. They are called low extreme and high extreme experiments. Each machining factor studied in low and high extreme experiments independently. Both Low extreme and high extreme experiments include two separate internal experiments too. For example, in low extreme experiment, for each factor, two independent internal experiments are arranged. In both internal experiments no experimental factors have their lowest values. In first internal experiment, studying factor has its lowest value, and in second one has its highest value. The results of each internal experiment are related tool life and energy consumption. Correspondingly, in high extreme experiment category for each factor, two isolated internal experiments are organized. In high extreme category, no experimental factors have their highest values in both internal experiments. Experimental factor has its lowest value in first internal experiment and highest value in second one. An example makes it clearer. Spindle speed effect is studied in low extreme and high extreme experiments category independently. In low extreme experiments, feed and cutting depth have their lowest values. In first internal experiment spindle speed has its lowest value. But in second one, it has its highest value. Related tool life and energy consumption in both internal experiments are recorded. Spindle speed also is studied in two internal high extreme experiments too. Table4 displays low and high levels for each machining factor with same increment steps for all factors.

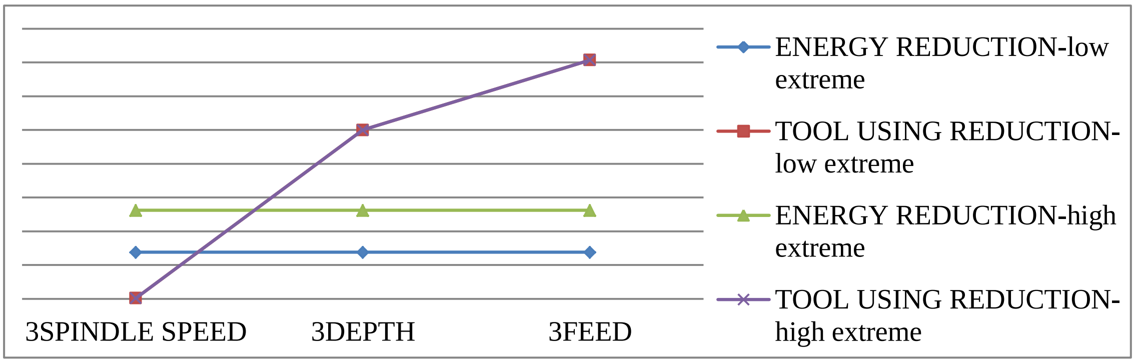
Table 4. Turning machining factors level on high and low extreme experiments

Depth of cut	Feed rate	Spindle speed
0.5	0.07	212,5
1	0.14	425

In high extreme category for each factor, the results of second internal experiment are divided to results in first internal experiment. Similarly, in low extreme category for each factor, results of second internal experiment are divided to results in first internal experiment. Each point in figure5 represents one of these dividing ratios. For example, in high extreme category of feed rate, energy consumption result when feed rate had the highest value (second internal experiment) divided to result when feed rate had its lowest value (first internal experiment).

There are four separate data series in figure5; two series for tool life and two series for energy consumption. Each line consists of 3 points. Each point represents the ratio value (explained above) for each factor.

Figure 4. Tool life and energy consumption ratio in high and low experiment category in turning



Energy ratios are the same for all factors in high and low extreme categories separately. It means same increase in cutting depth, spindle speed or feed rate in high or low extreme categories independently; reduces energy consumption with same ratio.

Energy consumption ratio line in low extreme is below of high extreme category. This difference means, increasing one machining factor when two other factors have their lowest values, reduces energy consumption more remarkably, comparing to situation when two other factors have their highest values.

Tool life has shown an interesting behavior in illustrated experiments in figure4. Tool life ratio from second to first internal experiment for each factor is the same for both low and high extreme categories. As mentioned earlier, tool life regulated by dividing tool operational life time (Taylor equation value), to machining time. All factors have same effect on machining time; clearly same increment in all factors, affect machining time similarly.

It is better to mention that, tool life should be an absolute value in production floors; a cutter can perform machining operation in explicit number of products, like 3 not 3.5. Due to all set up and tolerance limitations it is not acceptable practically, to stop operation in half for tool changing. So if tool life divided to machining time returns 5 or 5.2 or 5.9 tool life is 5. Since in this section, it is aimed to investigate on machining parameters interaction with tool life and energy consumption, raw value of tool life is used for analyzes.

By increasing cutting depth 2 times from 0.5 to 1 mm, tool life won't change; tool life ratio is 1. This increment reduces tool life time 0.25 times. In the other hand it also decreases the machining time 0.25 times. So increasing cutting depth won't have negative impact on practical tool life, while it positively reduces energy consumption in machining state.

In spindle speed the result is completely different; increasing spindle speed 2 times in both extreme categories reduces tool life 0.0625, which is tremendous reduction in practical perspective. Two reasons can justify this enormous reduction; first, the influence of spindle speed in tool life equation is considerably more than cutting depth and feed rate. Second, spindle speed value is much bigger than feed rate and cutting depth. Spindle speed and cutting depth increase 2 times similarly. Spindle speed increase from 212.5 to 425, and the difference is 212.5. But cutting depth increase from 0.5 to 1 and the difference is 0.5. This considerable difference is also aggravated while powered in Taylor equation. So when increasing spindle speed positively reduces MTEC, reduces tool life extensively. Same results about the drastic negative effect of spindle speed on tool life (comparing to other parameters) have been achieved in other research too (26), (30), (3).

Feed rate exhibits inverse behavior on tool life comparing to spindle speed. By doubling feed rate in both categories, tool life increases 1.1 times. Feed rate has the lowest contribution effect in tool life. Feed rate from low to high value has very small difference (which is 0.07). These justify improving in tool life by increasing feed rate.

Selecting each machining factor narrows down, possible selection range for other factors. So the selection order of machining factors is a momentous task. Especially while multi targets, like energy efficiency and tool cost are pointed. All machining factors increment has same positive effect in reducing MTEC. As a result, different effect of each machining factor on tool life can determine best selection order for them.

Cutting depth and feed rate increment don't have negative effect on tool life, while, spindle speed increment, reduces it considerably. As a result, it is better to put cutting depth and feed rate in first positions for selection and spindle speed at the end. By this order, feed rate and cutting depth can be selected as high as possible and consequently spindle speed selection range will be limited.

Feed rate increment, positively increases tool life when cutting depth won't change it. So the first machining selection factor should be feed rate. Feed rate directly effect on Surface quality which limits its selection spectrum to meat surface quality specifications (31) (32) (33). Considering this important limitation, the biggest feed rate should be selected in machining process to take advantage of its positive effect in both energy consumption and tool cost reduction. By determining highest value for feed rate, the biggest possible cutting depth with respect to selected feed rate should be employed. Since its high value reduces energy consumption, and doesn't affect tool life.

Selected cutting depth and feed rate restrict spindle speed selection range. This final selection depends on energy cost and tool cost preferences. In first look, if energy consumption is more important than tool cost, the highest possible spindle speed should be selected. Because it reduces MTEC positively and it's negative effect on tool life can be neglected. But by looking deeper, new tool replacement time may have some secondary effect on MTEC in system. In this case, increasing cutting speed reduces tool life, which increases tool replacement actions. Machine is in idle state during new tool replacement actions. By increasing this action, machine total idle energy consumption increases. In the other hand, higher spindle speed reduces machining state energy consumption. So it seems by increasing spindle speed there are several tradeoffs in system dynamic perspective. This interaction will be analyzed in system analysis section and final conclusion for spindle speed value will be suggested there.

Milling

Machining factors effect analysis on machining state energy consumption

Energy consumption formulation

Table 5 shows different milling scenarios with different combination of cutting depth, feed and spindle speed. Explained formulas and relations in chapter two have been used to find tool life, MRR and machining time. Taylor equation is adjusted by considering hypothetical tool and work piece material for this operation as below [15]:

$$T \quad = 2210/(v^{1.2} \times f_z^{0.25} \times a_a^{0.15})$$

In table 5, Cutting depth scale is in mm, feed per tooth in mm, spindle speed in revolution per minute, Machining Time in minute and Energy is in Mega Joule.

Table 5. Different milling machining factors combination and result on energy consumption

Scenarios	Cutting depth	Spindle speed	Feed per tooth	MRR	Machining Time	Number of operation with one tool(tool life)	Energy
A	2	450	0.2	9	3.68	11	282.96
B	1	450	0.2	4.5	7.02	6	498
C	2	390	0.2	7.8	4.2	11	316.04
D	1	390	0.2	3.9	8.04	6	564.1
E	1	450	0.14	3.15	9.87	5	682.3
F	2	450	0.14	6.3	5.11	8	375.12
G	1	450	0.06	1.35	22.6	2	1 501.52
H	2	450	0.06	2.7	11.5	4	784.72
I	1	390	0.14	2.73	11.3	5	776.8
J	2	390	0.14	5.46	5.84	9	422.3
K	1	390	0.06	1.17	26	2	1 722.02
L	2	390	0.06	2.34	13.2	5	894.9
M	1	331	0.2	3.31	9.41	7	652.6
N	2	331	0.2	6.62	4.88	12	360.2
P	1	331	0.14	2.31	13.3	5	905.7
Q	2	331	0.14	4.62	6.84	9	486.8
R	1	331	0.06	0.993	30.6	2	2 016.9
S	2	331	0.06	1.99	15.8	5	1 040.4

Like turning section, MRR is utilized to analysis machine energy consumption. Figure 6 shows MRR and milling energy consumption in different scenarios.

Figure5.MRR effect on milling operation energy consumption

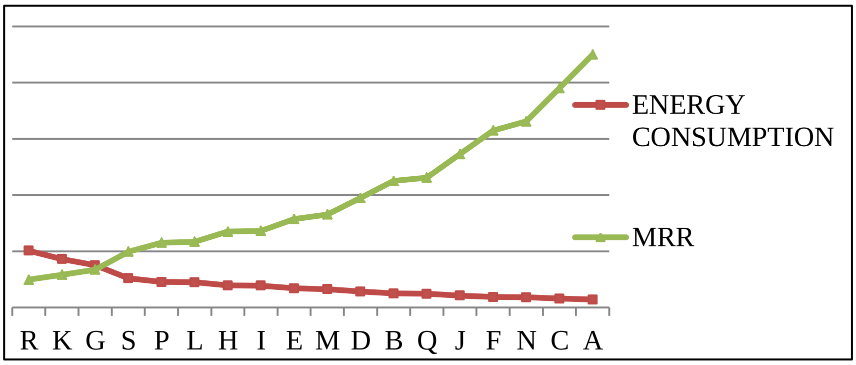


Figure 5 displays MRR and energy consumption per product in different scenarios. It demonstrates reduction in energy consumption by increasing MRR.

When MRR has small value, slight increment in it, reduces energy consumption considerably. Like turning, the slope of energy reduction by increasing MRR increases continually. In graph terminal, same increase in MRR does not reduce energy consumption explicitly. It means, when all machining factors have small values, increasing one factor reduces energy consumption noticeably. But, if MRR has a high value, reducing one factor won't increase energy consumption remarkably.

Using regression analysis, several functions for estimating energy consumption in machining state based on MRR and machining time are regulated and shown in table 6. Like turning appropriate correlation coefficient(r) are used to evaluate their estimation ability. X represents MRR and t is machining time which are independent variables and Y is energy consumption as dependent variable.

Table 6. Regression equations for milling operation energy consumption using MRR and machining time

Regression Function	CorrelationCoefficient
Y=(1.4x+64.5)×t	0.985
Y=(-0.023x^2-1.619x+64.14)×t	0.99986
Y=(-0.001x^3-0.002x^2+1.534x+64.23))×t	0.9999
Y=(0.004x^4-0.088x^3+0.560x^2-0.157x+ 65.25)×t	0.9998
Y=(-0.002x^5+0.061x^4-0.598x^3 + 2.616x^2 – 3.455x+67.38)×t	0.999
Y=(0.001x^6-0.042x^5+0.528x^4-3.252x^3+ 10.32x^2-14.10x+72.7)×t	0.952
Y=(64.42$x^{0.069}$)×t	0.8915

Comparing correlation coefficients for different equations, third degree equation shows better estimation ability here. So the final formula to calculate energy consumption of milling machine can be expressed as below:

Machine tool energy consumption= (-0.001MRR^3-0.002MRR^2+1.534MRR+64.23) ×Machining time

If MTEC should have been appraised out of experimented range, depending high degree equations to more coefficients, may reduce equation accuracy for estimation. In addition since linear equation R square has a small difference with second degree; it can be employed for MTEC estimation.

Like turning, this equation depends on parameters which are comprehensible and no need to find sophisticated variables or constants. It also enables monitoring energy consumption during machining time. So this equation can easily be used in production floors to estimate MTEC.

Methodology for analyzing energy consumption profile

The explained methodology in turning is applied here too. It follows described steps in turning completely. So to avoid repetition, it is containment up to explain the results.

Several estimating functions just based on MRR are regulated and shown in table 7. Recall, these equations can just estimate energy consumption in this particular case which milling operation will remove constant volume of material. However, above equation is more general and can appraise energy consumption for milling operations with different cutting volume

Table 7. Regression equations for milling operation energy consumption

Regression Function	Correlation Coefficient
Y=-174.5x+1474	0.67
Y=45.42x^2-602.5x+2219	0.9083
Y=-11.12x^3 +206.7x^2-1259x+2919	0.9804
Y=2.8x^4-62.8x^3+541.95x^2-2079x+ 3529	0.9954
Y=-0.675x^5+19.11x^4-210.7x^3 + 1137x^2 - 3126x+4146	0.9882
Y=0.206x^6-6.762x^5+89.22x^4-608.6x^3+ 2293x^2-4723.4x+4944	0.991
Y=1938$x^{-0,89}$	0.999

Power equation, has the biggest correlation coefficients, so it shows the best ability to estimate energy consumption in this case.

Obviously derivative equations of power equation don't have acceptable roots. Like turning to make sure about energy consumption profile, forth degree equation is analyzed here too. The roots of second derivative equation of energy consumption should be found. They represent the possible point which graph may start positive slope. In this point by increasing MRR, energy consumption will increase.

Y''=33.6x^2-376.8x +1083.9=0

This equation doesn't have an acceptable root. It clarifies MRR increment never increases energy consumption in this case (in experimented spectrum).

First derivative equation of estimation function is regulated as below.

Y'=11.2x^3-188.4x^2+1083.9x -2079=0

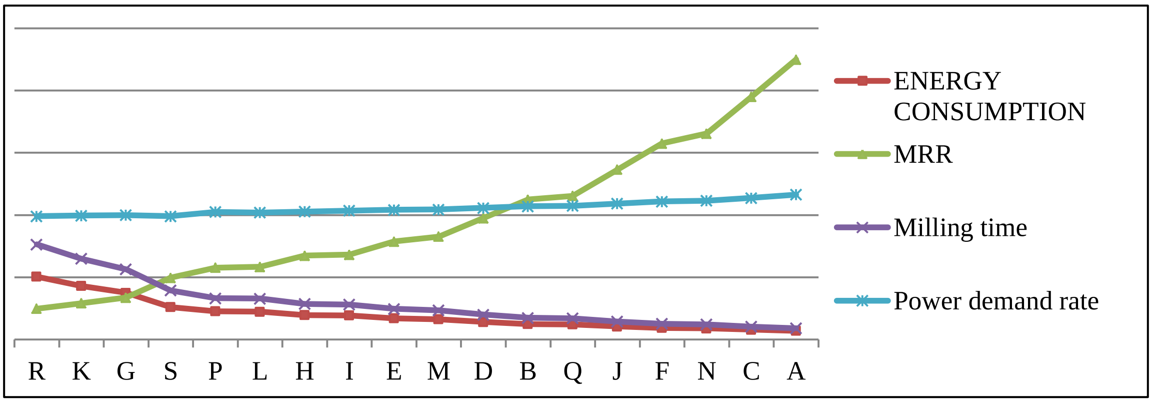
4.44 is the acceptable root of this equation. By increasing MRR from pervious value to this value, energy consumption remains constant. In other word, in this point even MRR is bigger than the previous value, but the energy consumption is the same.

Realizing basic interaction between MRR and energy consumption, and integrated it with proposed methodology, conduces to find the best machining factors, to satisfy energy efficiency targets.

Machine power demand profile

Machining time and related energy consumption in different experiments is shown in figure6.

Figure6.MRR interaction with machining time and energy consumption in milling operation

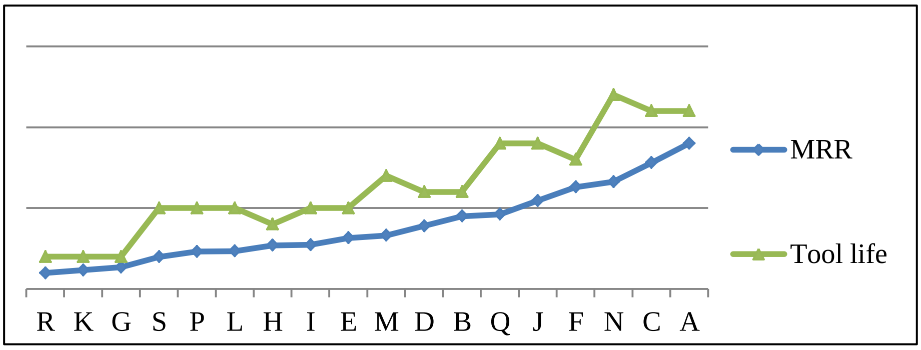


Like turning, by increasing MRR, machine power demand rate increases but total energy consumption reduces. This similar behavior in turning and milling can be justified through direct effect of machining factors on machining time and MRR; increasing machining factors reduce machining time, and increases MRR. Reducing machining time, not only compensate excessive power demands but also reduces total energy consumption.

Machining factor subsidiary effect on machine energy consumption

As explained earlier, increasing tool life, not only benefits directly on tool costs, but also reduces number of tool replacing actions. This reduces machine idle time and consequently machine idle energy consumption. In addition, it increases machine availability for machining operation and increases final outputs. In figure7, MRR and related tool life in different scenarios are displayed.

Figure7.MRR interaction with tool life in milling operation



The graph shows considerable fluctuations by increasing MRR. This makes it hard to associate an unequivocal relation between MRR and tool life. So like turning each machining factor effect on energy consumptions and tool life is investigated in extreme experiments separately.

Machining factors extreme analysis

Like turning two levels of experiment are set, low extreme category and high extreme category. The pattern of experiments is completely similar with turning. Detailed explanation of experiments setup, is given in turning section, so to avoid repetition it is sufficient to it.

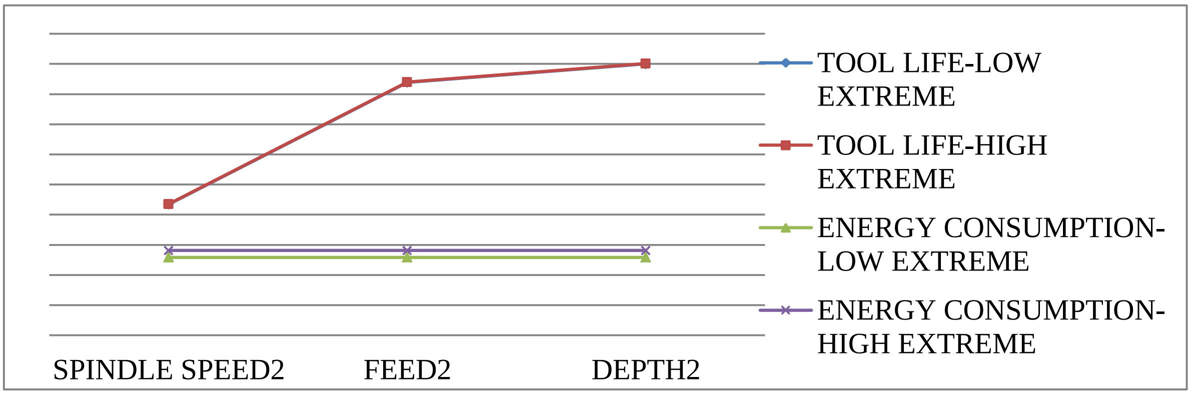
2 levels for each machining factor with same increment steps are shown in table 8. First level is called low and second level high extreme for each factor.

Table 8. Milling machining factors level on high and low extreme experiments

Depth of cut	Feed Per Tooth	Spindle Speed
1	0,06	331
2	0,12	662

Tool life ratio and energy consumption from high to low extreme experiments categories is displayed in figure8.

Figure 8. Tool life and energy consumption ratio in high and low experiment category in turning



Energy consumption ratio in low extreme category is below of high extreme for all 3 factors. This means increasing one factor to its highest value while two other factors have low values reduces energy consumption more remarkably, comparing to the situation when other factors have high values. Like turning, same increase in cutting depth, spindle speed or feed rate in high extreme category independently; reduces energy consumption with same ratio. This pattern exists in low extreme experiment category too.

Tool life ratio from second to first internal experiment for each factor is the same for both low and high extreme categories.

As mentioned in turning section, machining factors have same effect on machining time. By increasing cutting depth two times, tool life time reduces 0.9 times, while machining time reduces to half. So, tool life (number of possible operations with same tool) increases 1.8 times. Feed per tooth also has same behavior. By doubling feed per tooth; tool operational life time reduces 0.84 times, which increases tool life 1.67 times. By increasing spindle speed two times, machining time decreases to half, but tool operational life time decreases 0.4 times. As a result, tool life negatively 0.87. Different effect of each factor in tool life Taylor equation, leads to considerable difference in tool life ratios between spindle speed and two other factors.

As explained earlier, selecting each machining factor, narrows down possible range for other factors. So finding the best order to select machining factors values is a challenging task especially, when dealing with multi objective systems.

Integrating tool life analysis with energy consumption helps to find best selection sequence for machining factors in production system with energy consumption approach. All machining factors increment has same positive effect in reducing MTEC. As a result, different effect of each machining factor on tool life can determine best selection order for them.

Since positive effect of increasing cutting depth on tool life is more than feed per tooth and spindle speed, cutting depth should be selected first. In order to take advantage of its positive effect on energy consumption and tool life, it should be valued as high as possible.

Determining cutting depth value, limits feed per tooth possible selection spectrum. Since feed increment has dual positive effect on both on tool life and energy consumption, it should take the highest possible value too.

Feed per tooth and cutting depth values limits spindle speed selection range. Since increasing spindle speed positively reduces energy consumption and negatively reduces tool life, if energy concern was more important than tool costs, the highest possible value for it should be selected. But as mentioned in turning, new tool replacing action is affected by this parameter value. Higher spindle speed increases tool replacement actions. This increment increase machine idle energy consumption too, which may increase total energy consumption. So to find spindle speed best selection approach, its effect should be studied in a system for long time. This will be analyzed in system analysis section.

Chapter Four: Analytical system modeling

All above results and analysis considered machining operations on one unit in machining state, but practically production plans are long term and machine consumes energy in all states. Beside direct effect of machining factors on energy consumption, they also determine some process parameters like tool life. This can have great influence on system critical indexes like machine utilization which can probably affect energy consumptions too. Modeling production system and simulating its behavior for long time, helps to assess machining factors influence on whole system. In next section, model configuration to simulate production system is explained. It is employed for running different experiments with different input variables.

Hypothetical production system outline

To be able to analysis machining factors effect on system for long time, a hypothetical product and production line is considered and modeled.

The studying production process in this research includes turning and milling operations which are the basic sections of gear manufacturing process. Turning operation performs machining on work piece to reach required cylinder diameter. Process continues by milling to grove it and making the flutes. Beside new tool replacement and machine setup, work piece loading, unloading and machining operations, performs automatically.

Modeled production line, aimed to produce 3 product variants. Raw material comes from warehouse in first conveyor. A Robot in turning station, loads work piece form conveyor to lathe and unload it from lathe to next conveyor. In second station, there are one robot and two milling machines. Released parts from first station come in second conveyor. Robot will load work pieces to an empty machine and unload it in third conveyor. It navigates released parts to final storage. Machining time, machining setup and cutting tool life differ for each variant. So if the variant of pervious served part in machine was different from variant of coming part, machining setup should be changed.

In order to increase model consistency with real world, random failure events, random repair time and product quality rejections based on process capability are superposed in model. These features improve model resemblance with real production lines. So the results and analysis are not limited exclusively to academic research, but also is applicable for real manufacturing systems. Matlab 2013a was used to make explained production line model.

Model configuration

Each subsystem represents one main part of the explained production line. Subsystems contain variables, equations, flowcharts and internal subsystem to facilitate the logical flow of entities in system. These all together, contribute to provide appropriate simulation outputs. Comparing the results of different simulations (experiments) enables to analysis the effect of input variables on system.

T

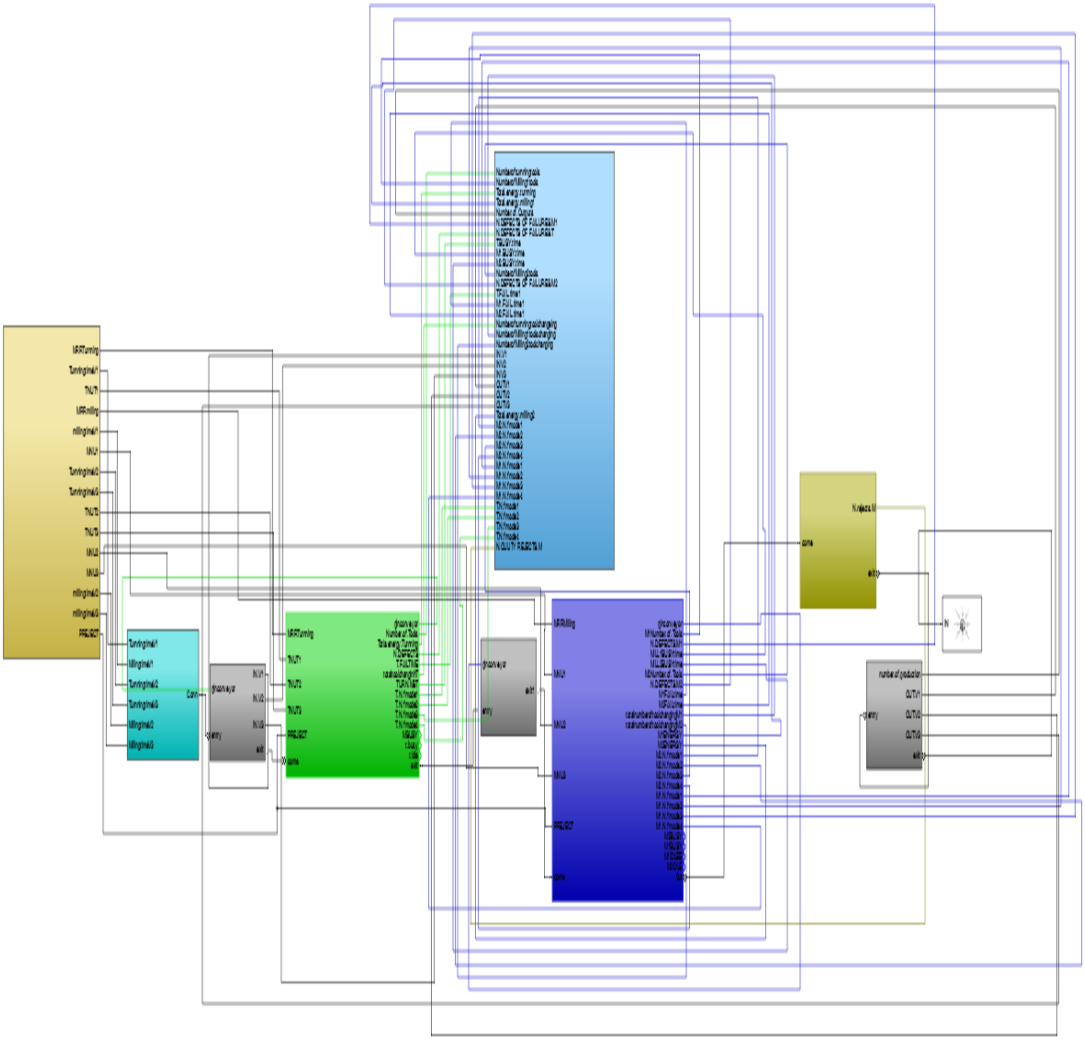


Figure9. Production line model overall configuration

The model includes several main subsystems; 1-Set Process parameters subsystem 2-order generating subsystem 3-first conveyor subsystem4-turnning station subsystem4-second conveyor subsystem5-milling station subsystem 6-quality inspection subsystem 7-third conveyor subsystem8-storage block.

To get better understanding of the system configuration, each subsystem will be explained separately.

1-Set process parameters subsystem:

This subsystem consists of three independent subsystems; *turning process parameter determination*, *milling process parameter determination* and *process capability index determination* subsystem.

In *turning* and *milling process parameters determination* subsystem, depth of cut, feed and cutting speed can be set. By defining machining factors in these two subsystems and using appropriate functions, machining time, material removal rate and tool life for each variant will be calculated. These values will be sent to entity generator, milling station and turning station subsystems with connecting output ports. By changing machining factors, machining time, tool life and material removal rate will change. Machining factors are changed through different experiments in order to find and analysis their effect on system targets. It helps to find the best process parameters selection to achieve production goals.

TNU1, *TNU2* and *TNU3* are calculated tool life for each variant in turning operation based on determined machining factors. *MNU1*, *MNU2* and *MNU3* are related tool life for each variant in milling operation. *MRR Turning* and *MRR milling* variables correspondingly are MRR for turning and milling operation. *Turning TimeV1*, *V2*, *V3* and *Milling timeV1*, *V2* and *V3* are respectively calculated turning and milling time for each variant.

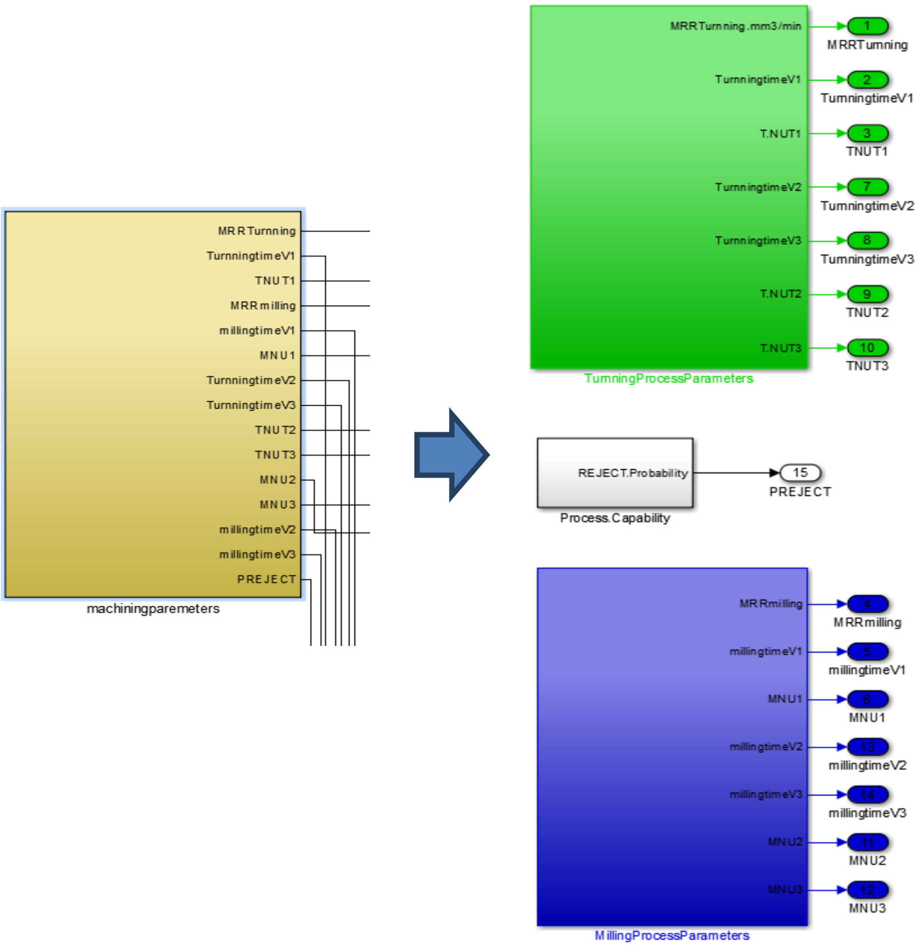


Figure10. Set process parameters subsystem

Turning process parameter determination and *milling process parameter determinations* subsystems are illustrated in figures 11and 12.

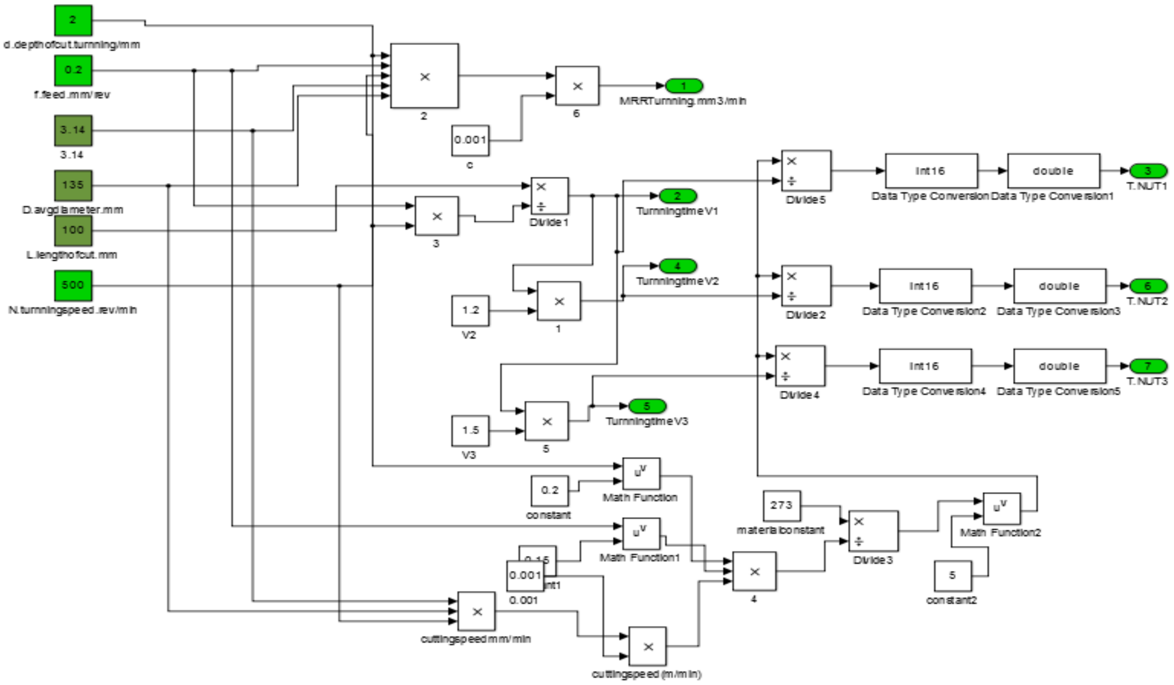


Figure11. Turning process parameter determination subsystem

The light green squares represent machining factors blocks. They can be changed to run different scenarios of production with different machining factors. The ovals are output variables, turning time, tool life and MRR for each variant, which will be sent to appropriate subsystems. The same structure exists in milling subsystem as well, which is illustrated in figure14, in blue theme.

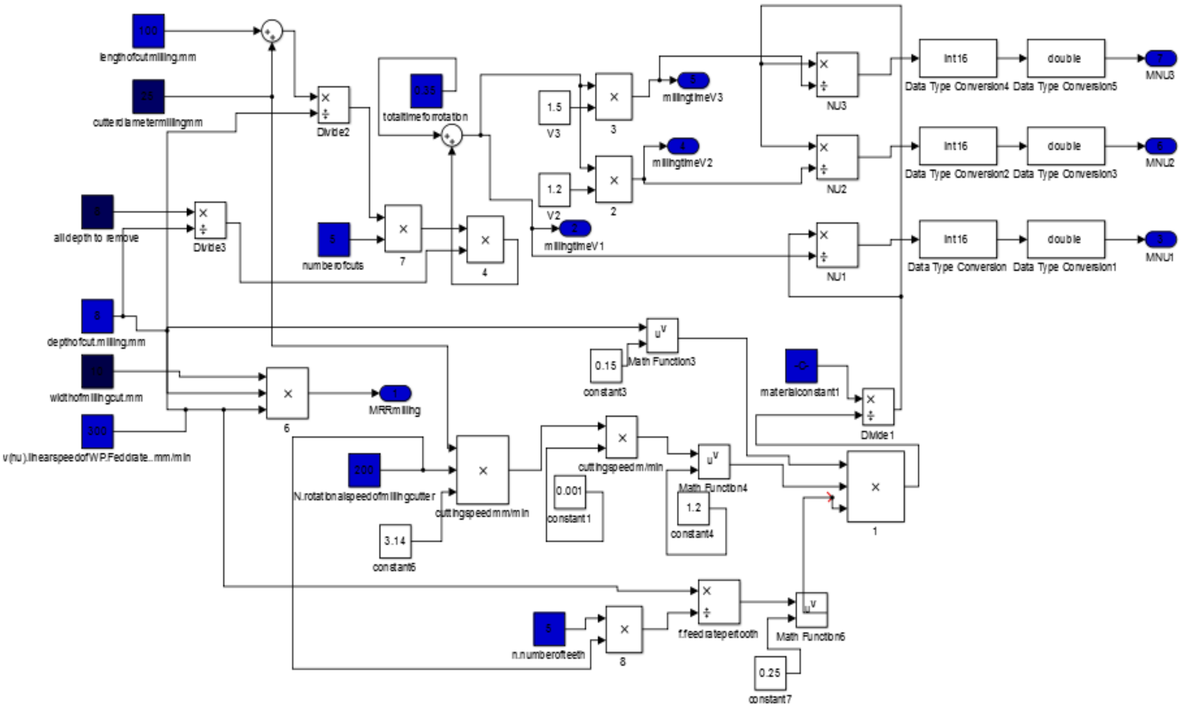


Figure12. Milling process parameter determination subsystem

In *process capability index determination* subsystem we can define the process capability index (CPK).This determines the rejection probability of each part due to quality specifications (*P.REJECT* variable). There are plenty of options for setting CPK index. By choosing each value, the flowchart, returns appropriate rejection probability and send it to milling and turning station subsystems.

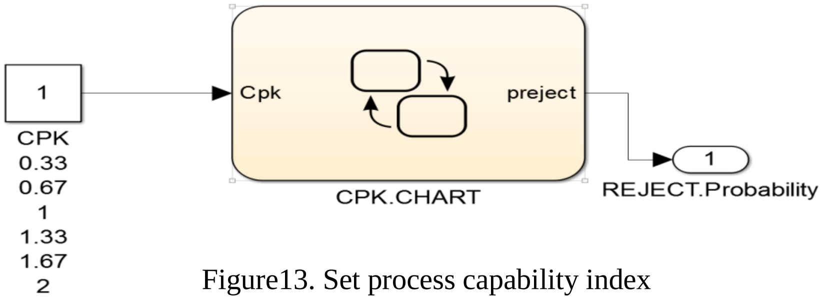


Figure13. Set process capability index

2-Entity generator subsystem:

This subsystem represents the raw material warehouse. It releases parts to production line based on customer orders. The entity generator block generates entities based on an empirical time interval too. The set attribute blocks, set variant type from1-3randomly. After setting variant type for each entity, the appropriate milling, turning and set up time for each variant should be set as an attribute. An output switch separates entities based on their variants. Milling and turning time for each variant is sent from input port of *Set process parameters subsystem* (*Milling timeV1, V2, V3* and *Turning time V1, V2, V3* variables). Machine set up time for each variant will be set as an attribute too. When all variant related attributes are set, a path combiner block gathers all variants together and sends them to the next subsystem, which is conveyor1.

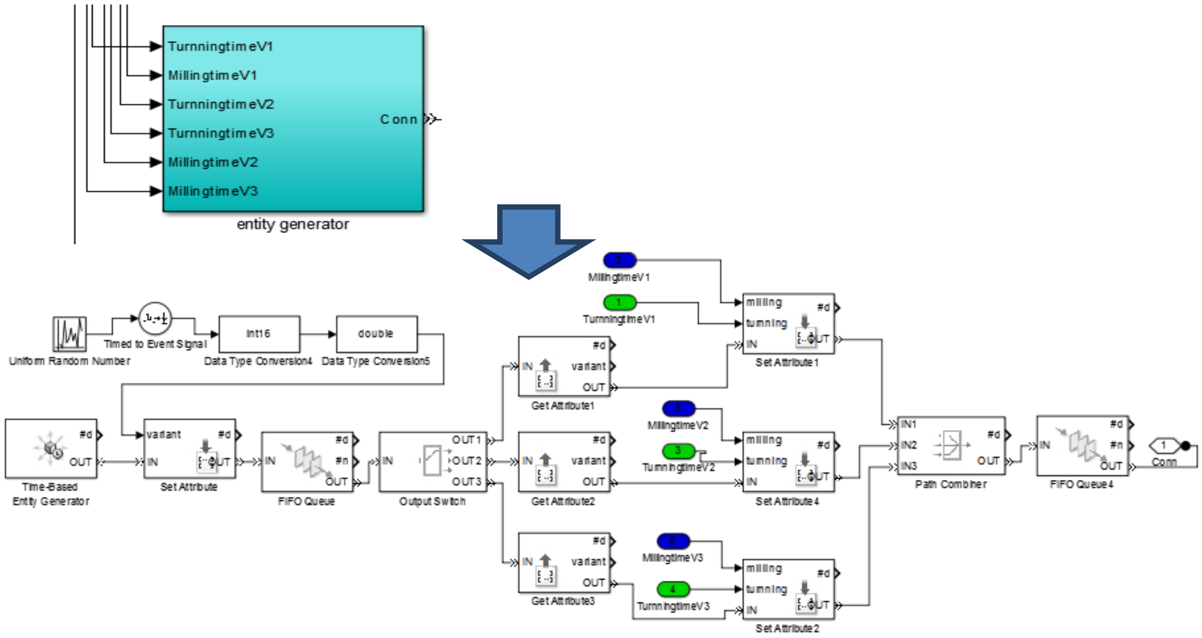


Figure14. Order generating subsystem

3-Conveyor1:

There are several conveyors between main subsystems. The first conveyor is between warehouse and turning station. It passes raw materials from warehouse to turning station. This subsystem contains a flowchart which checks for the availability of conveyor motor. If the motor was on, it sends a positive signal to conveyor gate (part feeder subsystem). Motor ON or OFF state is defined with a signal builder block.

.

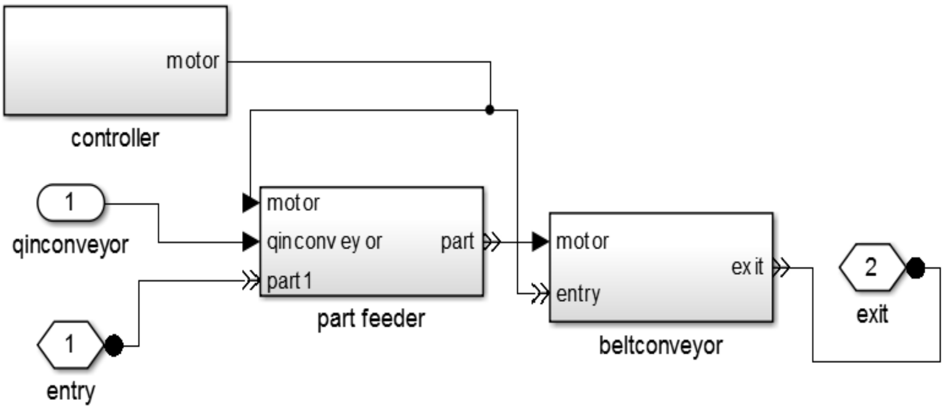


Figure15. conveyor overall configuration

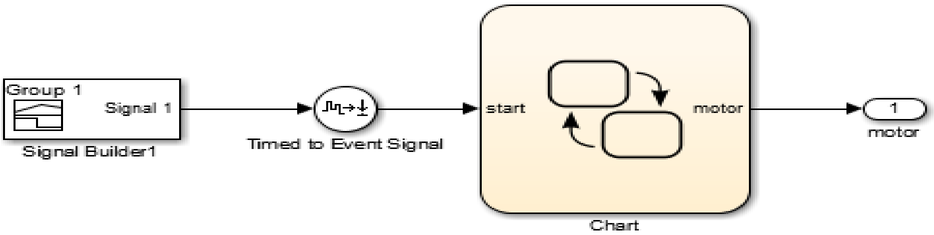


Figure.16Conveyor motor controller subsystem

If number of entities in buffer block (waiting parts), between machining station and conveyor was less than buffer capacity (*qi conveyor* variable), the block also sends a positive signal to conveyor gate (part feeder subsystem

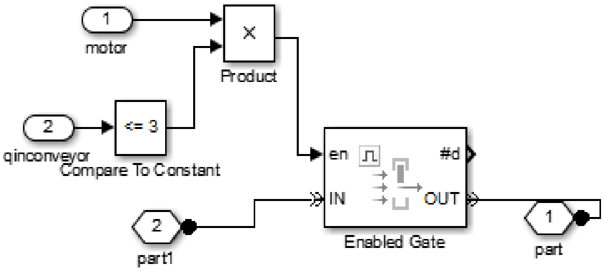


Figure17. conveyor, part feeder subsystem

If the conveyor belt was not full and both signals were positive, the conveyor gate opens and passes new entities (the logical operator checks the positivity of both signals). An N-server block in this subsystem represents the conveyor which can pass multiple parts in the same time. If all the capacity of N-server block is full, it means that, the conveyor is full and cannot serve new part.

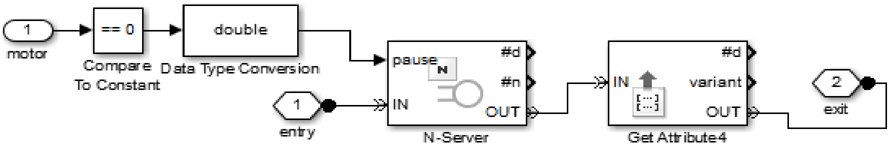


Figure18.conveyor belt subsystem

4-Turning station subsystem:

This subsystem represents the turning station in production line. A FIFO queue in turning subsystem represents buffer in before turning station.

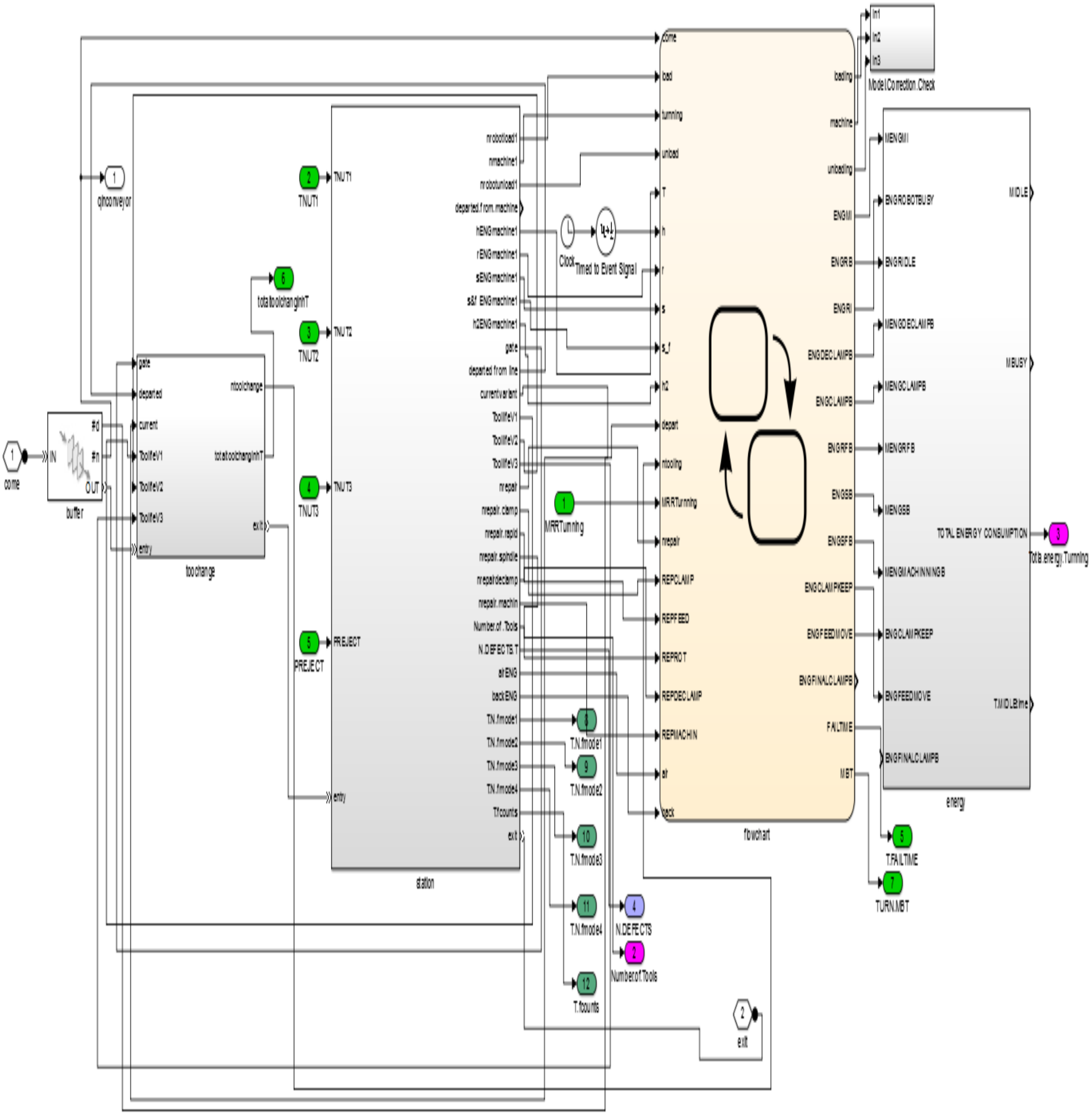


Figure19.Turnning station subsystem

This subsystem includes 4 main interrelating subsystems; 1- set up subsystem, 2-machining subsystem, 3- turning flowchart, 4- energy calculation subsystem. Due to existing a lot of interrelating variables, input and output ports in this subsystem, they are explained separately.

1-machine Setup subsystem

As mentioned earlier, a coming part can get the service, if machine and robot are free. There is also setup condition which needs to be satisfied. Setup action includes machining setup and tool changing. If the variant of coming part was different from the variant of pervious served part, machine needs new set up. Machine also may need a new tool, if tool life is finished. Set up action is presented in this subsystem as a single server block, and gets service time from setup time attribute of coming entity. As explained, setup attribute was defined in entity generator subsystem for each variant separately.

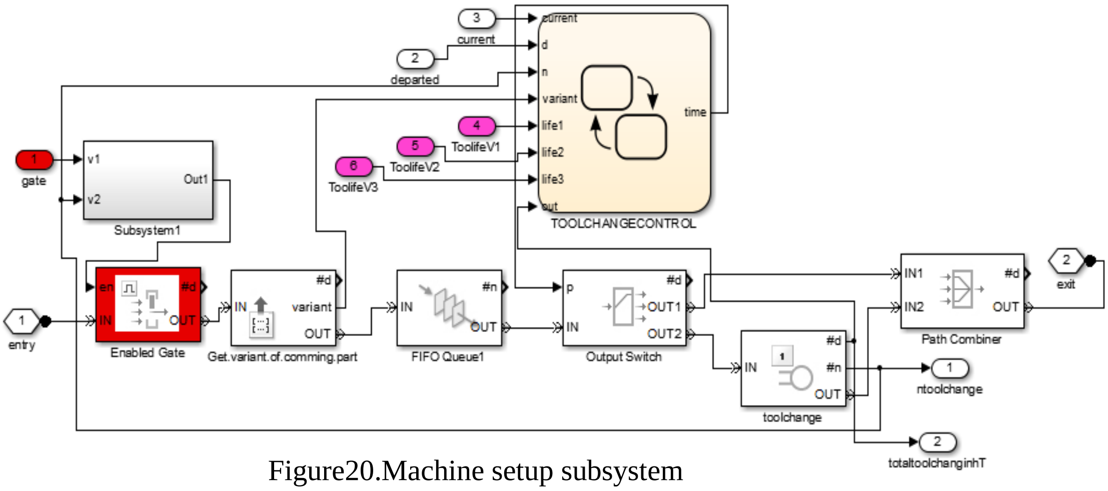


Figure20.Machine setup subsystem

Machining subsystem sends signals about machine and robot availability (*gate* variable) by an output port to this subsystem. The setup server also sends a signal about its state (V2 variable).If setup action is performing in machine the sending signal is 1, otherwise is 0.

Logical operator (compare to constant) in this subsystem, checks the coming signals from machining subsystem and setup server. If machine, robot and set up server were available (the signals should be zero), the logical operator generates a positive signal (*out* variable). It sends the signal to an enable gate block to let the entity enter to turning station.

The need for new tool replacement or new set up is checked with a flow chart. As soon as the gate opened, the robot checks the variant type (before loading). It sends variant type to the flow chart. Get attribute block in figure 20, represents this action. The *machining* subsystem sends tool life signal separately for each variant to this flow chart. If the coming signal about tool life was zero (*life1, life2* and *life* 3 represents tool life for each variant), it shows that, machine can operate with existing tool; otherwise (if the signal was 1) it means, tool life is finished, and machine needs new tool to operate.

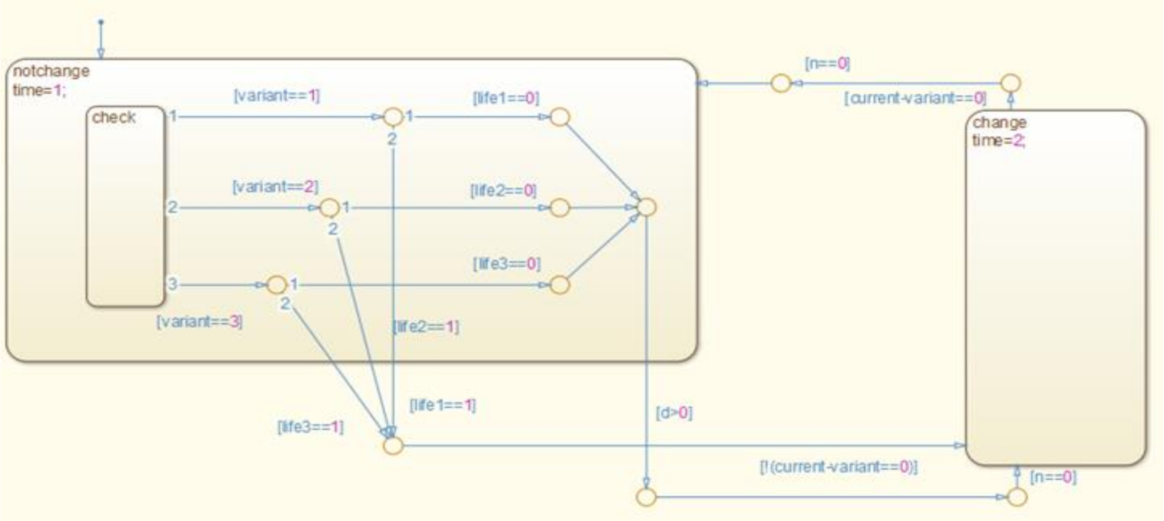


Figure21. Machine setup flowchart subsystem

Machining subsystem sends another signal about the variant of pervious served part in the machine to this flow chart (*current* variable in flowchart). The chart starts with robot signal, and check the coming variant type (*variant* variable). If the coming variant type was the same with the pervious variant (*current* variable) and tool life signal for that variant was 0, chart output will be 1 (*Time* variable). If variant type of coming entity was not the same with pervious served entity, or if tool life signal *(life1, life2, life3*) of coming variant was 1 the chart returns 2.

The output of the chart (*time* variable) is used as an input signal for an output switch block. This output switch has two paths. First path is directly connected to *machining* subsystem, but the second one is connected to the setup server block. The output of the chart defines the appropriate path of the coming entity. If *time* was 1, the coming entity, will be immediately loaded (*time* 1 means, the coming variant is the same with the previous one and the tool life is not finished yet). So it will be navigated to the machining subsystem directly without any extra action. If *time* was 2, the setup action should be performed. This may be needed for changing machine setup because of changing variant or tool replacement due to finishing tool life. Setup action performs in second path of output switch in a single server block. It receives its operation time from setup time attribute of the entity. When setup time in single server elapsed, it releases entity to the machining subsystem.

2-Machining subsystem

As soon as the entity, reaches *machining* subsystem, a get attribute reads variant type. It sends variant type to setup subsystem as *current* variable. This will be used for comparing variant of next coming entity, with this one (explained above). After setup, robot loads the part on machine. So robot loading block starts servicing coming entity. While robot loading block is busy, the signal about robot status is 1. Logical operator in setup subsystem (explained above), receives this signal and the station gate, will be closed. When loading block service time finished (loading action on robot is finished), the machining operation will start in machining subsystem.

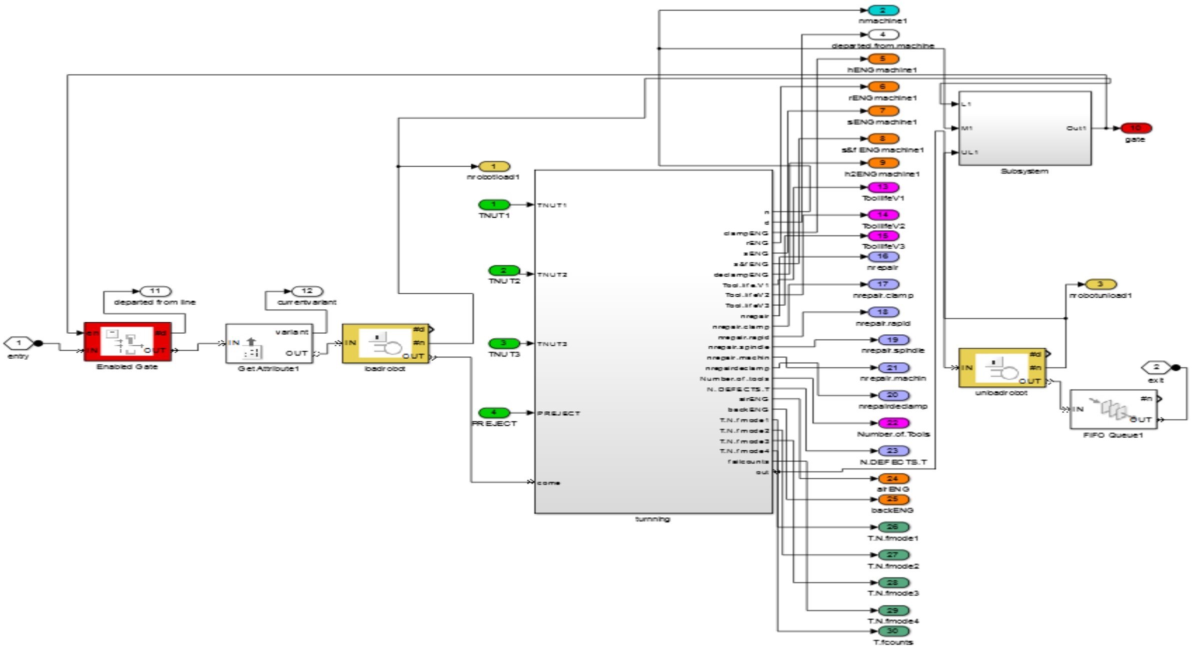


Figure22. Machining subsystem in turning station

As explained earlier, machine experiences different states in its operational cycle. Several subsystems represent different machining states. To obtain a better understanding of logical flow of entities in system and model configuration, they will be explained separately.

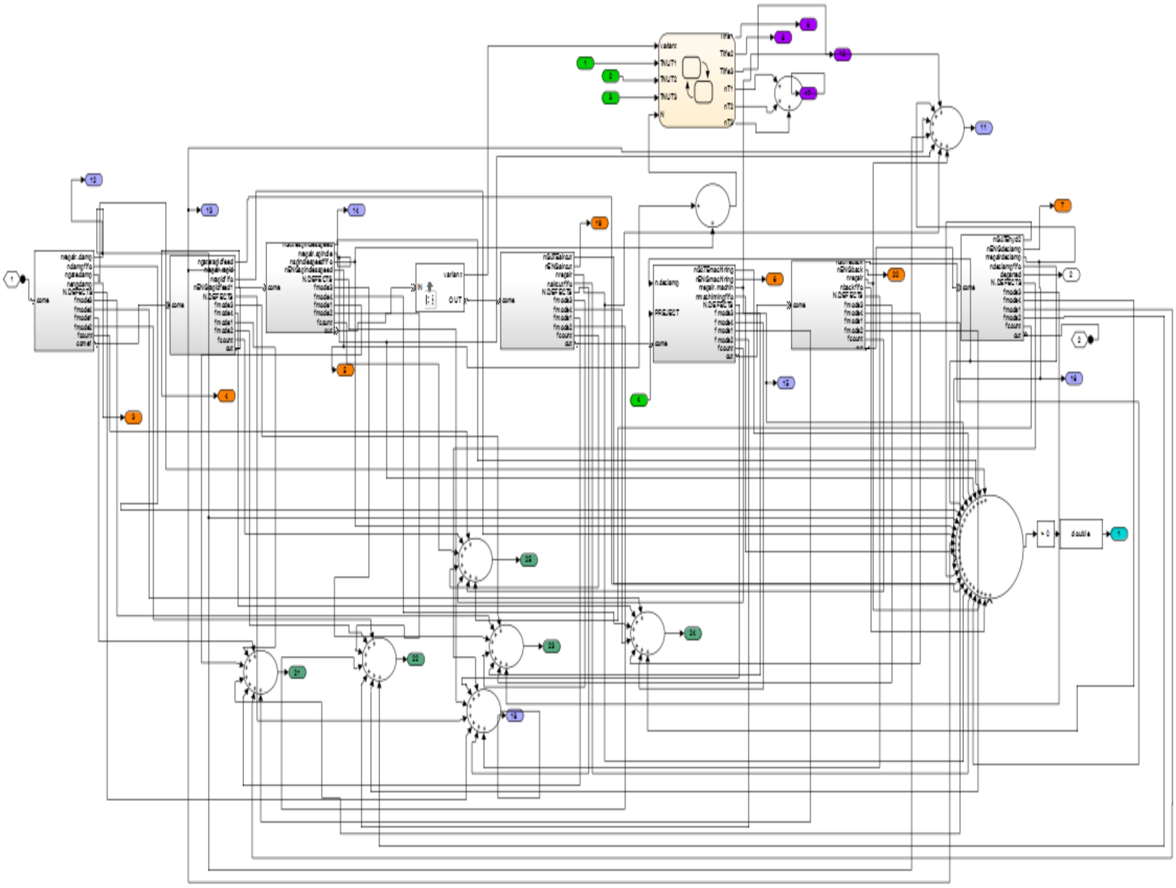


Figure23. Machining states subsystems in turning

Clamping subsystem

This subsystem represents clamping the work piece in machine area. The first block in this subsystem is a single server and elapses determined time which represents clamping action time. While this server is busy it sends a signal (*clamping machine state* signal) to the main machining subsystem to show that the machine is busy with this action.

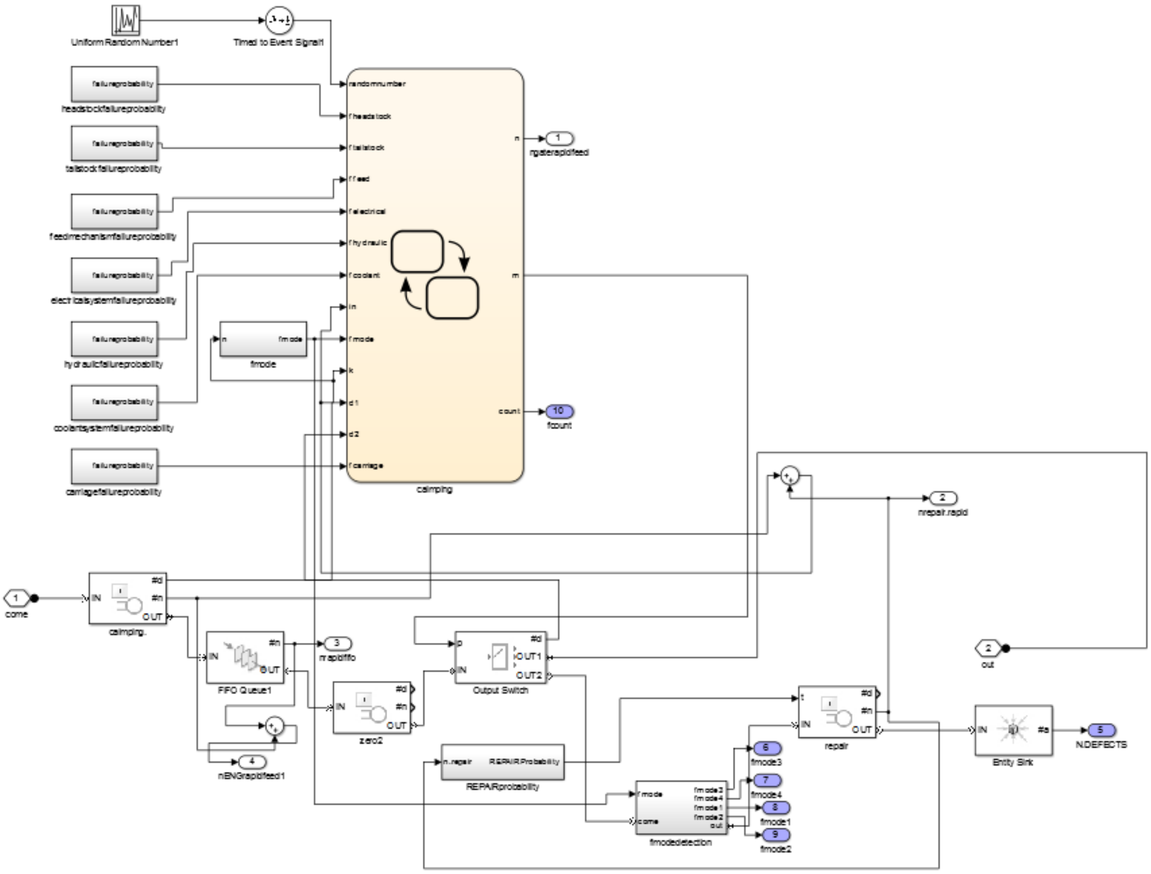


Figure24. Clamping state subsystem

Failures can occur during each state of operation cycle. Turning machine is divided to 7 main sections; head stock, tail stock, feed mechanism, electrical system, hydraulic system, coolant and carriage (34). Each section has its own failure rate. There are 4 typical failure modes for each failure; fuse burnt, component damage, looseness and circuit fault (34). Some probability distributions express acceptable capability to model, failure events, like Gama, Wibuland and Exponential distributions (35) (36) (37). Exponential distribution is used here to generate failure events, during operational cycle. It should be mentioned that, more than one failure can occur in time instant.

Different subsystems generate failure probabilities for each section of the turning machine based on exponential distribution, with different failure rates (different λ).

Table 9 shows each turning machine related failure rates (34) (38).

Table9. Turning machine sections and related failure rates

position	λ
Headstock	0.000228
tailstock	0.000045697
feed mechanism	0.000075
electrical system	0.000544
hydraulic	0.00002937
coolant	0.00002937
carriage	0.0001363

Each failure mode can cause failure in each machine section with a specific occurrence probability. For example component damage can conduct failure in carriage of a turning machine.

Table 8 shows each failure mode occurrence probability in turning machine (34).

Table10. failure modes occurrence probability in turning machine

failure mode	occurrence probability
Fuseburnt	0.1447
component damage	0.3684
looseness	0.1842
Circuit fault	0.3026

Failure mode subsystem produces different failure modes based on their probability to occur.

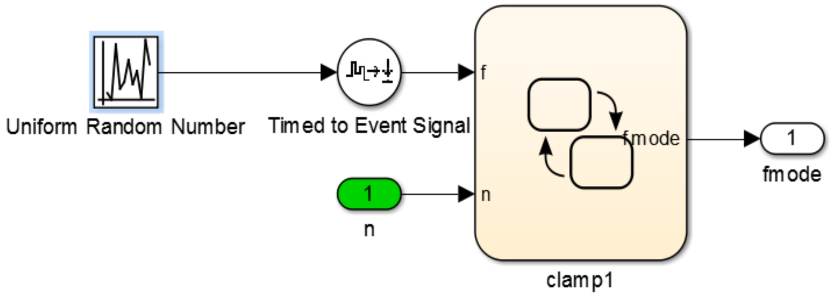


Figure25. Failure mode generation subsystem

The illustrated flowchart in figure25 gets a random value from random generator block and compares it to each failure mode occurrence probability. Based on the received value and each failure mode probability, it returns a failure mode, for occurred failure (will be explained later).

Machine state control chart in this subsystem continuously checks machine general states. This chart consists of 3main basic machine states; 1-idle, 2- busy, 3- fail.

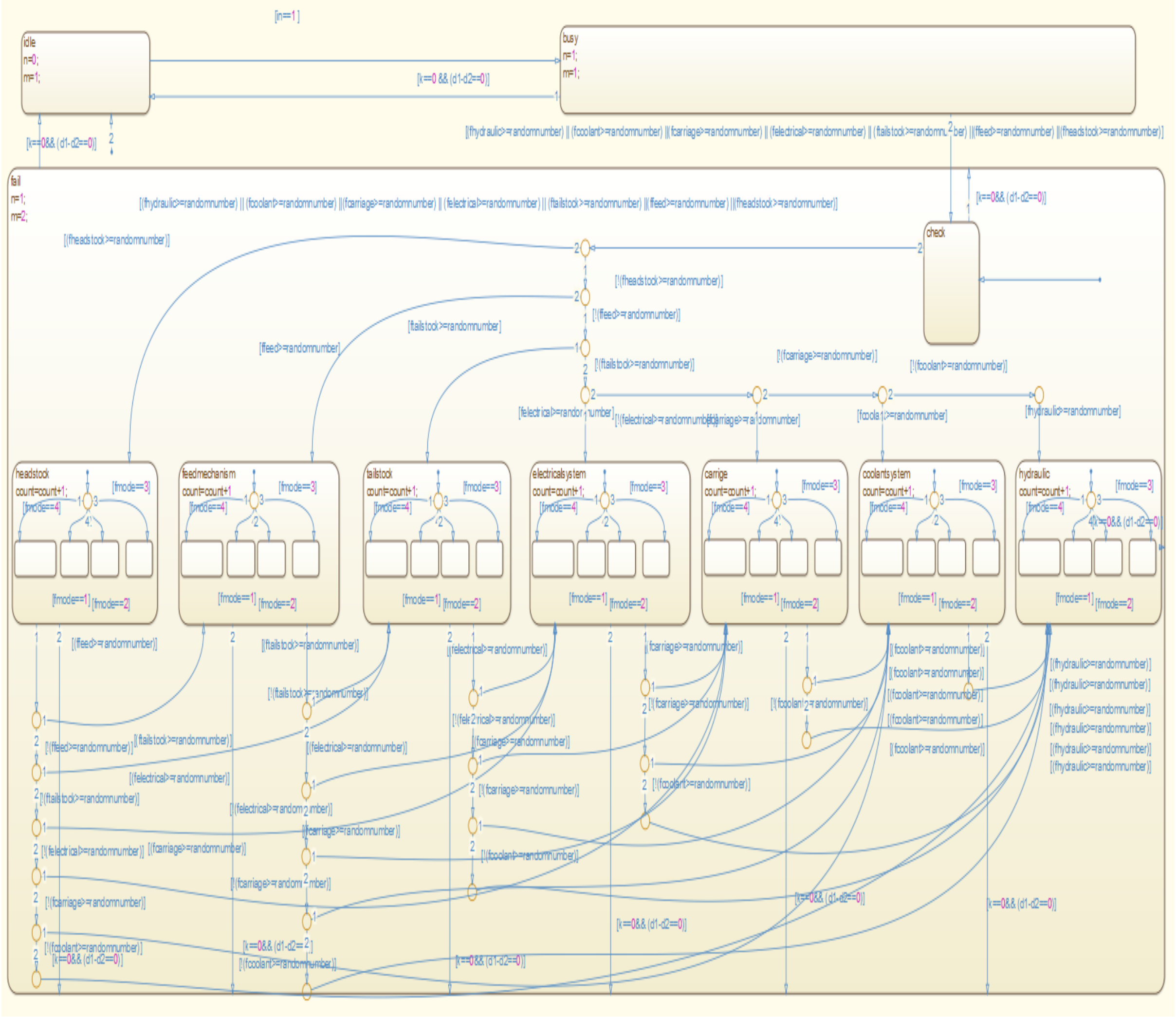


Figure26. Machine general state control chart

When entity enters to this subsystem, it firstly involves with the single server block, which demonstrates clamping action. This server continuously sends a signal about its state to this subsystem control chart (*clamping machine state)*. When the signal changed from 0 to 1, it means that, the machine is busy, so chart transits from idle to busy state.

While flowchart is in busy state, it keeps controlling for failure(s) occurrence and machine state. Chart returns two basic outputs, *machine busy state* and *machine fail state.* As long as clamping block is busy, *machine busy state* variable gets 1. This shows that the machine is not available because it is busy and *machine fail state* value is 1 too (will be explained later).

Several failure subsystems generate failure probabilities for different machine sections. These probabilities are compared with a random number between 0-1 in control chart. If at least one of failure probabilities was more than, given random number, chart transits from busy to fail state.

In fail state based on the given random number, and produced failure probabilities for each machine section, failure position(s) will be detected (head stock, tail stock, feed, electrical, hydraulic, coolant, carriage).

As explained above, failure mode subsystem also sends continually different failure modes to this chart. When the chart is in fail state, it takes the given failure mode from failure mode subsystem and activate the appropriate failure mode state on detected failure position state.

An example makes it clearer. When the control chart is in busy state, the head stock subsystem produces failure probability of 0.05(as an example). If this number was more than random number like 0.001, the chart transits to head stock fail state. This is a sub state of the general failure state. The failure mode subsystem also sends a signal that clarifies, which failure mode (circuit fault, component damage, fuse burn, looseness) caused this failure. As a result, the chart activates for example circuit fault state of headstock failure state.

When the chart is in fail state, *machine busy state* value is 1 which shows that machine is not available because of failure occurrence. *Machine fail state* value is 2, which shows machine is in fail state and need to be repaired.

If no failure happens, chart stays in busy state during its operation action. In this case, by elapsing clamping time on server block, it releases the entity to continue its operation cycle in remaining states. As soon as server licenses the entity, *clamping machine state* signal changes to 0. Chart receives the signal and transits from busy to idle state.

After clamping server block there is an output switch with two paths. The entity path selection is based on *machine fail state* output of control chart. If this variable value is 1, it means that no failure occurred. In this case the entity chooses the first path. It navigates the entity to leave this subsystem and directly continues its operation in following subsystem (which is rapid move here after clamping).

If *machine fail state* value is 2, it means failure(s) occurred, so machine is in fail state, and needs to be repaired. Repair action takes time, but it is not always a deterministic constant value. Some literatures declared that, repair time conforms to some probability distributions, like Gama or exponential (39). A flow chart (repair time chart) produces different repair times with different probabilities. In second path of mentioned output switch, there is single server block which represents repair action. This block takes the repair time from repair time chart. Repair server block also sends a signal to the control chart of this subsystem. While the block is busy, it means the repair action is performing; repair signal is 1 and the chart remains in fail state. As soon as repair time elapsed, the signal changes to zero, and chart transits to idle state. Repair block passes the entity to sink block, which is be considered as defect.

When, chart returns to idle state, the *machine busy state* output is 0. It shows clamping action is finished.

This subsystem provides different outputs for controlling the logical flow of parts in system or to prepare appropriate simulation outputs. *Machine busy state* output: illustrates machine state for this action. When this output is 0, machine is not doing this action and not in fail state either. *Clamping machine state*: when machine is busy this output signal is 1 and it is 0 when machine is in fail or idle state. This signal will be used for energy consumption calculation. *Repair state output*; this output is 1 when the repairing action is take place otherwise it is 0. *Number of defects* output; this output shows the total number of defects in this machining state. *Failure mode* outputs; they show occurrence number of each failure modes.

Rapid move subsystem
This sub system represents the rapid move of cutting tool to the work piece. The structure, logical flow and outputs of this subsystem are totally the same with clamping subsystem.

Spindle start rotation subsystem

This subsystem represents start rotating of the spindle. The structure, logical flow and outputs of this subsystem are totally the same with clamping subsystem.

Air cutting from home position to work piece subsystem

This subsystem represents machine air cutting state from tool home position to work piece vicinity. The structure, logical flow and outputs of this subsystem are totally the same with clamping subsystem.

There is a get attribute block between air cutting and metal cutting state. Before entity enters to metal cutting state, get attribute block reads its variant type. The block sends variant type to a control chart which evaluates tool life condition (tool life control chart).

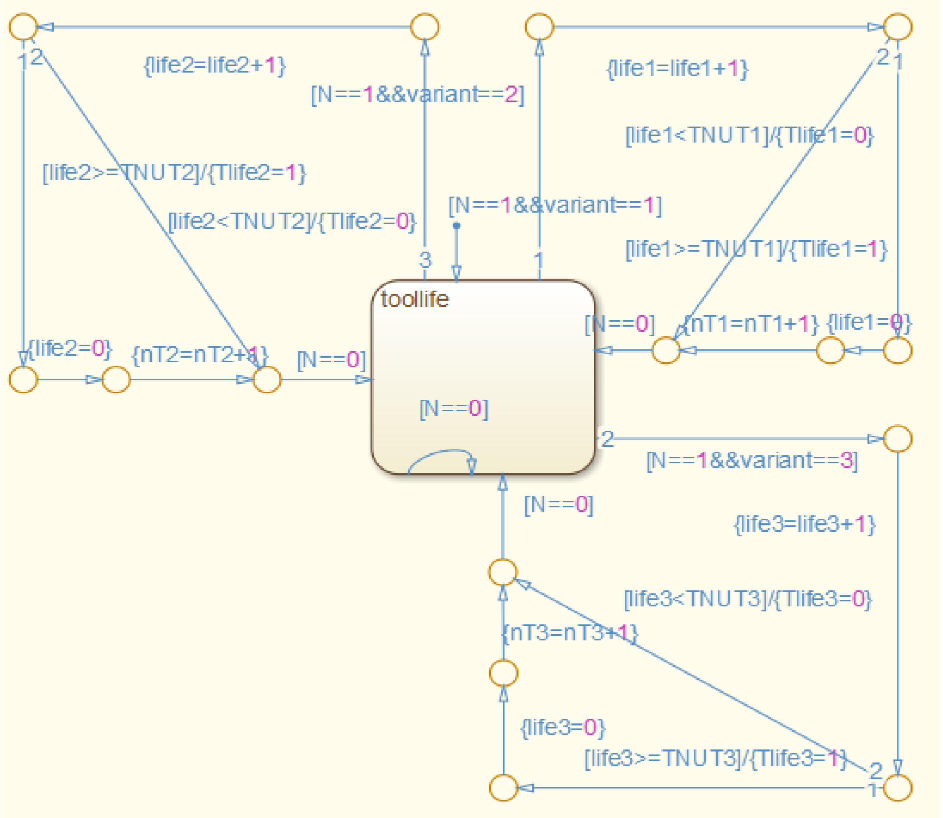


Figure27. Tool life control chart

Tool life chart gets tool life (maximum number of operations with same tool) for each variant separately from *Set process parameters* subsystem (*TNU1* or *TNU2* or *TNU3*). It also gets the variant type of coming part from above mentioned get attribute block. The chart increases the number of performed operation with the same tool for that variant, by 1(because one following machining operation will be executed with this tool for coming entity). If the updated local variable (*life1 or life2 or life3*) was less than tool life for that variant, the chart updates an output variable to 0 (*Tlife1*, *Tlife2* or *Tlife3*).This means this tool still can continue to operate. But if the local variable (*life1 or life2 or life3*) was equal or bigger than tool life for that variant, *Tlife1* or *Tlife2* or *Tlife3* value will be 1. It means this tool cannot perform any more operation. So for next coming same variant, the tool should be replaced with new one. Chart also increases the number of used tools for this variant by 1 to demonstrate number of used tools in simulation interval. *TLife1 or Tlife2 or Tlife3* will be sent to the *Setup* Subsystem to clarify tool condition (explained in *Setup* Subsystem).

Metal cutting subsystem
This subsystem represents metal cutting or material removal operation. This subsystem also has similar structure with other subsystems. The single server which demonstrates machining operation gets machining time from turning time attribute of coming entity (*turning time v1, turning time v2, turning time v3*). As explained earlier, this attribute was valued and set in *Set process parameters* and *entity generator* subsystems.

Process capability influences product quality only in this state and not in other states. As explained earlier, *set process parameters* subsystem provides quality rejection probability based on process capability index. *Set process parameters* subsystem sends this value (*preject*) to machining subsystem.

Roundness, surface roughness and straightness are general quality specifications in turning operations (33). 3control charts are designed to tag some products out of acceptance interval for each quality parameters. Each chart gets *preject* value and compares it with a random number. Chart returns2, if the random number was more than *preject*, otherwise returns 1. After server block (machining), there is a set attribute block. It gets charts outputs and set them as quality attributes for each entities. If an entity has at least one quality attribute with value 2, it will be considered as out of acceptance interval and should be rejected.

The remaining configuration and outputs of this subsystem is the same with other subsystems.

Air cutting from work piece to home position
This subsystem represents cutting tool recedes from the work piece, since the machining operation is finished in pervious subsystem. This subsystem includes similar structure with explained clamping subsystem.

De-clamping subsystem
This subsystem represents de clamping action of work piece from the turning machine. This subsystem has similar structure with other subsystems.

The machine is not available during busy time and repair action (fail state).If any state of operation cycle like clamping, air cutting and others are in busy or repair state, obviously machine is not available. In the other word, machine is available, if all operation states are idle.

As explained in clamping subsystem, the *machine busy state* output represents machine state for that operation. So the machine is available if the *machine busy state* outputs of all subsystems are zero. In order to find the machine availability, all *machine busy state* outputs from all states subsystems will be summed up together and compared to zero. If summation was more than 0(one operational state is not idle), the overall *machine busy state* output will be 1. It means that, machine is not available. This output will be sent as a signal (*gate* variable) to the *Setup* subsystem in order to check the machine availability for accepting new entity in turning station subsystem.

If failure event happens in each operation, the machine is in repair state. *Repair state* outputs of all states subsystems, will be summed up and represents machine repair state. This output will be used to find machine total repair time (will be explained later).

In order to find the total number of defects in turning station, *Number of defects* outputs from all states subsystems will be summed up together. *Failure mode outputs* for all subsystems also summed up together to find total occurrence number of each failure mode.

When all operations finished on work piece, robot unloads the work piece to the second conveyor. The coming entity from de clamping subsystem gets service from robot unloading block. Robot unloading block keeps sending a signal to the *setup* subsystem about its state. While the robot is busy with loading or unloading the signal is 1 otherwise it is 0. Conveyor and buffer capacity is limited. As explained earlier, FIFO block represents buffer between stations. If conveyor server and FIFO blocks were full, robot keeps the work piece on hold, until the conveyor block can accept a new entity. A FIFO block just after the robot block represents this waiting situation. Released entity from turning stations enters to second conveyor to fulfil its cycle on milling station

Turning flowchart

As mentioned earlier, the main goal of production system modeling in this research was to compare machine tools energy consumption in different situations. The main task of turning flowchart in this subsystem is to monitor and calculate machine energy consumption with different input variables.
This chart is the state flow diagram of work piece progresses through different states in turning station. It contains all the states mentioned above and some extra auxiliary states. There are a lot of variables, which control transition action from one state to another in flowchart. Simulation clock is a fundamental input for this flowchart, which returns simulation time in any time instant of simulation. Local variables and functions are defined in order to provide appropriate chart outputs. There are some transition actions, which should be performed, while the transition condition for leaving a state is satisfied. The logical transmission in this chart is the same with the logical flow of entities through explained blocks and subsystems.

There are two main states in this chart, station busy and station idle states, which demonstrate overall state of the station.

Station idle state is the default state in this diagram. The condition of leaving this state is availability of machine and robot to serve a new entity. These conditions are investigated through the signals which robot server blocks, setup server block and machining subsystem send to this chart continually. The precondition to this transition is existence of an entity to get turning service. As explained earlier, FIFO block represents buffer before station. It also keeps sending a signal to the chart which shows the entities quantity in buffer block. If this input was more than zero, and robot and machine signals were zero (which means they are available), the chart transits to station busy state.

Station busy state consists of 4 main sub states; 1-setup 2- loading 3-machining 4-unloading.

When transition condition of leaving idle state to busy state satisfied, it enters setup sub state of busy state first. As explained before, setup block (which represent the setup action) keeps sending signal about its state to this chart. The chart stays in this state while the block is busy (the signal is 1).When the signal changes to 0, the chart goes to robot load sub state. While the robot load block is busy, its sending signal to chart is 1. As soon as loading time in server block elapsed, the signal changes to 0 and the transition condition to leave robot load state is satisfied. The next state after robot load is machining sub state.

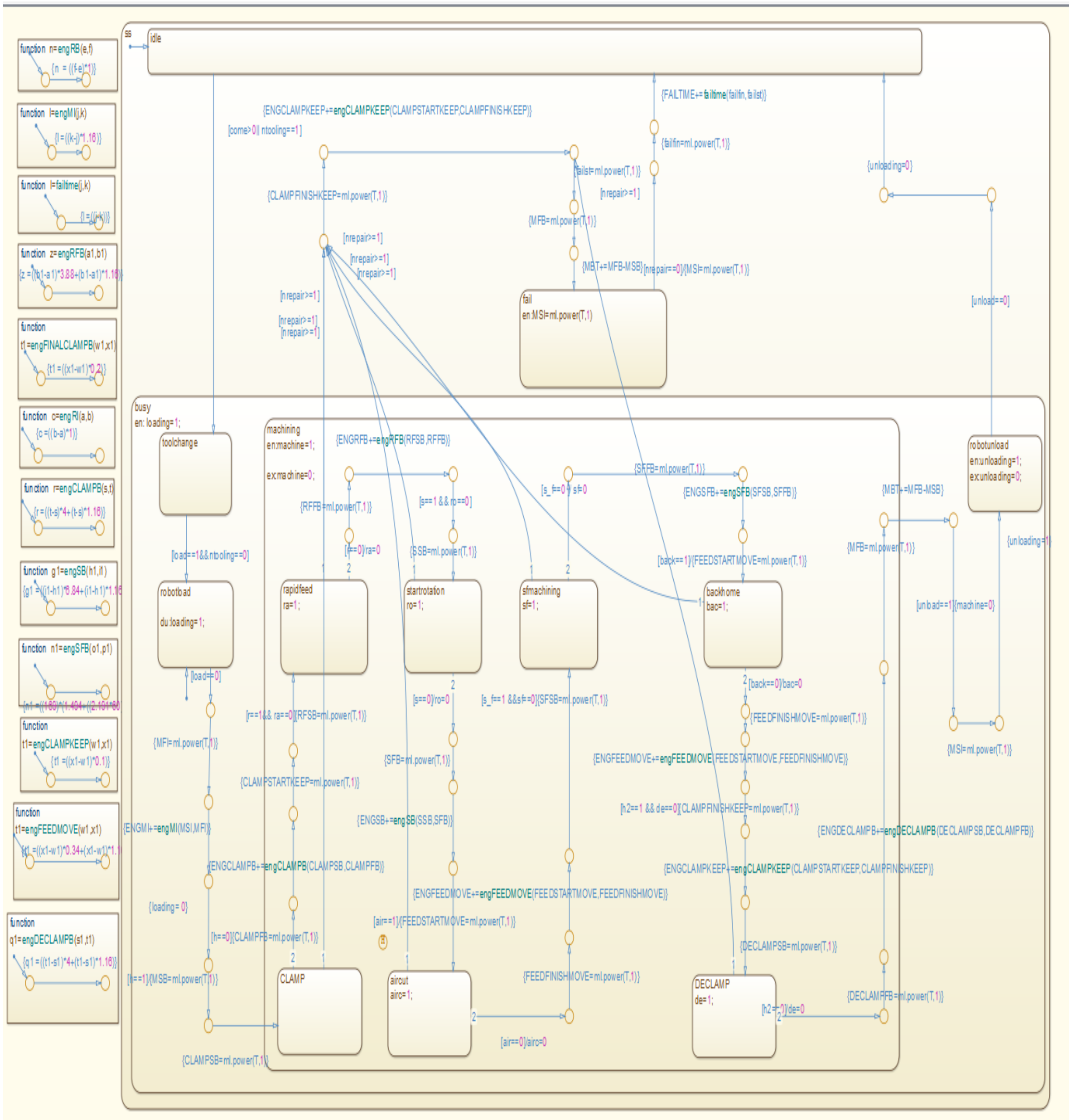


Figure28. Turning station flowchart

There are some transitions actions of leaving robot loading to machining state;

Energy consumption rates used in energy functions in all states beside metal cutting states are educed in previous research and publications in this filed (mostly figures) (17) (2).

Updating *Machine finish idle* local variable: As soon as loading is finished, machine starts its operations and its idle state finishes. So mentioned variable gets current time of simulation. It shows the time that machine finished its idle state.

Calling function *of turning machine idle energy*: this function subtracts the *machine start idle* variable (will be explained soon) from *machine finish idle* and multiply it to machine idle energy consumption rate in idle state, and sums it to the pervious value. This function helps to calculate total machine energy consumption in its idle state in simulation interval.

Turning machine idle energy= (Machine finish idle-machine start idle) ×1.16+turning machine idle energy

Machine idle or fix energy consumption, represent the energy that machine needs for running basic equipment features like power to start up the computer or fans etc.

The chart sends *Turning machine idle energy* value as an output to the turning energy calculation subsystem and simulation result subsystem.

Machine start busy local variable gets current time of simulation and saves it as the time that machine starts operation cycle.

Clamp start busy local variable gets the current time of simulation and saves it as the time that clamping operation started.

Machining sub state consists of different sub states, to represent different actions in operation cycle. Machining cycle starts with clamping, so the first state which will be activated in machining sub sate is clamping sub state.

As mentioned earlier all subsystems of machining subsystem, provides *machine state* signal. It is called *clamping machine state* signal in clamping subsystem, or *rapid move machine state signal* in rapid move subsystem. If the machine is busy with related state action, the ---*machine state* signal value is1, otherwise it is 0.

The chart stays in clamping sub state while the *clamping machine state* signal is 1, and leaves it as soon as *clamping machine state* signal changes to zero (Since clamping time is elapsed in server block). There are some transition actions of leaving clamping sub state;

Updating *Clamp finish busy* local variable: it gets e current time of simulation and saves it as the time that machine finished clamping action.

Calling function *of clamping energy*: this function subtracts *clamp start busy* variable from *clamp finish busy* and multiply it to energy consumption rate in clamping action state, and sum it to pervious value.

Clamping energy= (clamp finish busy - clamp start busy) × (4+1.16) +clamping energy.

Energy consumption in clamping state consists of two basic sections; first: fix energy consumption which machine needs to run its basic features during clamping, which is 1.16KW. Second: machine needed energy to clamp the work piece. Value *4* in above formula points clamping energy consumption when clamping device suddenly changes from un-hold to hold. This output represents the total clamping energy consumption, and will be sent to the energy calculation subsystem.

The clamping state and its related energy function represent peak value of energy consumption due to momentum effect.

As long as clamp device holds work piece, machine consumes moderate and steady value of energy too. This starts just after peak energy consumption in clamping state. To consider this energy consumption, *clamp start keep* variable gets current time of simulation when clamping state is finished. It represent the time machine starts its steady energy consumption due to part holding.

Before activating rapid move sub state, the chart checks failure event occurrence. As explained before, machining subsystem sends *repair state* signal to the chart. If the signal is 1, it means machine is in failure (repair) state, and the condition for activating repair state is satisfied. There are some transition actions related to satisfying the transition condition for activating repair state. If failure happens, part holding action will be finished (since the machine is turned off during repair action). So the *clamp finish keep* local variable gets current time of simulation as finishing time of part holding. The *clamp keep energy function* will be called and subtracts the *clamp start keep* variable from *clamp finish keep* and multiplies it to energy consumption rate during part holding action and sums it to pervious value.

Clamp keep energy= Clamp keep energy+ (clamp finish keep-clamp start keep) ×0.1

0.1KW is machine energy consumption due to holding work piece. This output represents total energy consumption in clamping state, and will be sent to the energy calculation subsystem.

Fail start time local variable, gets the current simulation time and save it as the time that, failure and repair actions started.

Machine finish busy local variable gets current simulation time and saves it as the time that, machine busy time is finished.

Machine busy time function will be called and subtracts *machine start busy time* from *machine finish busy time* and sums it up to the pervious value. This output represents total busy time of the machine, and will be sent to the simulation result subsystem.

Machine busy time=machine busy time+ (machine finish busy -machine start busy)

When all transition actions are done, the chart activates repair state. As long as *repair state* signal is 1, chart stays in repair state. When signal changed to zero, leaving condition of the repair state is satisfied.

There are some transition actions of leaving the repair state. *Fail finish time* local variable gets current simulation time and saves it as the time that repair time is finished. *Machine fail time* function will be called, and subtracts *machine finish fail time* from *machine start fail time* and sums it to the pervious value.

Machine fail time= machine fail time+ (fail finish time-fail start time)

Chart sends this value as an output to the simulation result subsystem which shows the total fail time of turning machine.

Machine start idle local variable gets current simulation time and saves it as the time that machine starts its idle state again (as soon as repair action is finished, machine will be turned on, to serve new part).

If no failure had happened during clamping state, machine follows its basic cycle. So the transition condition to next state which is rapid move state here is satisfied. There is a transition action before activating this state too. *Rapid feed start busy* local variable gets the current simulation time and saves it as the time that, rapid move action started.

The main logic of transition conditions and transition actions, for the rest of the machining sub states is the same with the clamping state, so to avoid repetition, the states, transition conditions and actions will be mentioned briefly.

Rapid move sub state

Transition condition to next state; *rapid feed machine state* signal equals to 0.

Transition actions to next state; *rapid feed finish busy* local variable gets current simulation time and saves it as the time that, rapid move action is finished. *Rapid feed energy function* updating: this function subtracts the *rapid feed start busy* variable from *rapid feed finish busy* and multiplies it to the energy consumption rate during rapid move action and sums it to pervious value.

Rapid feed energy= (rapid feed finish busy- rapid feed start busy) × *(3.88+1.16)*

1.16KW points to the machine basic energy consumption and 3.88KW is energy consumption rate during rapid move state.

This output represents total energy consumption in rapid move state, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to repair sub state is *repair signal* equals to one.

Transition actions of leaving this state to repair state are completely the same with the clamping state.

Transition condition of leaving this state to next action (spindle acceleration) in machining cycle is repair signal equals to zero.

Transition action of leaving this state to spindle acceleration state is; *spindle acceleration start busy* variable gets the current simulation time and saves it as the time that spindle has started rotation.

Spindle acceleration sub state

Transition condition to next state; *spindle acceleration machine state* signal equals to 0.

Transition actions to next state; *spindle acceleration finish busy* variable gets current simulation time and saves it as the time that, spindle acceleration action is finished. *Spindle acceleration energy function* updating: this function subtracts *spindle acceleration start busy* variable from *spindle acceleration finish busy* and multiplies it to the energy consumption rate during spindle acceleration action and sums it to pervious value.

Spindle acceleration energy=Spindle acceleration energy+ (spindle acceleration finish busy - spindle acceleration start busy) × (6.84+1.16)

6.84KW is energy consumption rate during spindle acceleration state until it reaches the determined speed.

This output represents the total energy consumption in spindle acceleration state, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to repair sub state is *repair signal* equals to one.

Transition actions of leaving this state to repair sate is completely the same with pervious states.

Transition condition of leaving this state to next action (air cutting) in machining cycle is repair signal equals to zero.

Transition action of leaving this state to air cutting state is, *air cutting start busy* local variable gets the current simulation time and saves it as the time that air cutting action started.

Air cutting from home position to work piece sub state

Transition condition to next state; *air cut machine state* signal equals to 0.

Transition action to next state; *air cutting finish busy* local variable gets the current simulation time and saves it as the time that, air cutting action is finished. *Air cutting energy function* updating: this function subtracts *air cutting start busy* variable from *air cutting finish busy* and multiplies it to the energy consumption rate during air cutting action and sums it to pervious value.

Air cutting energy =Air cutting energy+ (air cutting finish busy- air cutting start busy) × *(0.34+1.16)*

0.34KW is machine energy consumption rate during air cutting action.

This output represents total energy consumption in air cutting from home position to work piece state, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to repair sub state is *repair* signal equals to one.

Transition actions of leaving this state to repair sate is completely the same with the pervious states.

Transition condition of leaving this state to next action (machining) in machining cycle is *repair* signal equals to zero.

Transition action of leaving this state to machining state is, *machining start busy* local variable gets current simulation time and saves it as the time that turning machine started machining action.

Machining sub state

Transition condition to next state; *machining machine state signal* equals to 0.

Transition action to next state; *machining finish busy* local variable gets the current simulation time and saves it as the time that, machining action is finished. *Machining energy function* updating*:* as concluded in chapter 3, energy consumption in metal cutting state is function of material removal rate and machining time. MRR value has been sent from *set process parameters* subsystem to turning chart, which is calculated based on machining factors (feed rate, depth of cut and spindle speed). So the energy consumption in this state is calculating, using this formula;

Machining energy = (-0.00001MRR^2+0.7446MRR+65,865) × (*machining finish busy –machining start busy*) +*Machining energy*

This output represents total energy consumption in machining state, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to repair sub state, is *repair* signal equals to one.

Transition actions of leaving this state to repair state are completely the same with the pervious states.

Transition condition of leaving this state to next action (air cutting from work piece to home position) in machining cycle is *repair* signal equals to zero.

Transition action of leaving this state to air cutting is, *air cutting from work piece to home position start busy* local variable gets the current simulation time and saves it as the time that back home position action started.

Air cutting from work piece to home position sub state

Transition condition to next state; *air cutting from work piece to home position machine state* signal equals to 0.

Transition action to next state; *air cutting from work piece to home position finish busy* local variable gets the current simulation time and save it as the time that, machine air cutting from work piece to home position action is finished. *Air cutting from work piece to home position energy function* updating: this function subtracts *air cutting from work piece to home position start busy* variable from *air cutting from work piece to home* position *finish busy* variable and multiplies it to energy consumption rate during air cutting action and sums it to pervious value.

air cutting from work piece to home position energy = air cutting from work piece to home position energy+(air cutting from work piece to home position finish busy- air cutting from work piece to home position start busy) × (0.34+1.16)

0.34KW is machine energy consumption rate during air cutting.

This output represents total energy consumption in air cutting from work piece to home position, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to repair sub state is *repair* signal equals to one.

Transition actions of leaving this state to repair state are completely the same with the pervious states.

Transition condition of leaving this state to next action (de clamping) in machining cycle is *repair* signal equals to zero.

There are two transition actions of leaving this state to de clamping state; *clamp finish keep local* variable gets the current simulation time and saves it as the time that, machine finishes holding the work piece. *Clamp keep energy function* updating: this function subtracts *clamp start keep* from *clamp finish keep*, and multiplies it to the energy consumption rate during work piece holding, and sums it to pervious value.

Clamp keep energy= Clamp keep energy+ (clamp finish keep-clamp start keep) × 0.1

0.1KW is machine energy consumption rate during holding work piece

De clamping start busy local variable gets the current simulation time and saves it as the time that de clamping action started.

De clamping sub state

Transition condition to next state; *de clamping machine state signal* equals to 0.

Transition actions to next state; *de clamping finish busy* local variable gets the current simulation time and saves it as the time that, machine de clamping action is finished. *De clamping energy function* updating: this function subtracts the *de clamping start busy* variable from *de clamping finish busy* variable and multiplies it to the energy consumption rate during de clamping action and sums it to pervious value. This energy, like clamping state referees to peak value energy consumption, due to sudden change of clamping device position from hold to un-hold.

De Clamping energy= (de clamp finish busy–de clamp start busy)(4+1.16) +clamping energy*

4 KW is machine energy consumption rate during de clamping action.

This output represents total energy consumption in de clamping state, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to repair sub state is *repair* signal equals to one.

The Transition actions of satisfying leaving this sub state to repair state is the same with others, but it does not include updating *clamp finish keep* local variable and calling *clamp keep energy function*, because the action is already finished.

The machining cycle is finished with this operation, and the chart should go to robot unloading state.

Transition condition of leaving this state to robot unloading state is *repair* signal equals to zero.

There are some transition actions of leaving this state to unloading state. *Machine start idle local* variable gets the current simulation time and saves it as the time that machine starts to be idle. *Machine finish busy* local variable gets the current simulation time and saves it as the time that, machine busy time is finished. *Machine busy time function* will be called and subtracts *machine start busy time* from *machine finish busy time* and sums it up to the pervious value.

Turning Machine busy time=turning machine busy time+ (machine finish busy -machine start busy)

This output represents total busy time of the machine, and will be sent to the simulation result subsystem.

While robot unloading block is busy, its sending signal is 1. As soon as unloading is finished, signal changes to 0 and transition condition to leave robot unload state is satisfied. The next state after robot unloading state is station idle state. It was the first activated state in this cycle. This cycle will be repeated by coming new entity to turning station subsystem.

Released entity from unloading server will continue in conveyor 2 to reach milling station subsystem.

Energy calculation subsystem

This subsystem receives turning flowchart outputs. It sums up all states energy outputs during operating cycle and returns the summation as *total turning machine busy state energy consumption* output to simulation result subsystem.

The *total turning machine busy state energy consumption* will be added to *idle energy consumption* (chart output) and will be sent as *total turning energy consumption* to simulation result subsystem.

Conveyor2

The second conveyor is between turning and milling station. The structure of second conveyor subsystem is the same with the first conveyor.

Milling station

This station has two similar milling machines, which make the flutes on coming parts and shape the final gears. There is a robot in this station to load parts from conveyor to machines and unload them to the final conveyor.

The milling station subsystem main configuration is similar with first station. 7 subsystems are defined in milling station subsystem to make the interaction of local blocks and variables easier; *machine1 setup subsystem, machine1 subsystem, machine2 subsystem, machine2 setup subsystem, milling flowchart, energy calculation subsystem* and *gates control* subsystem.

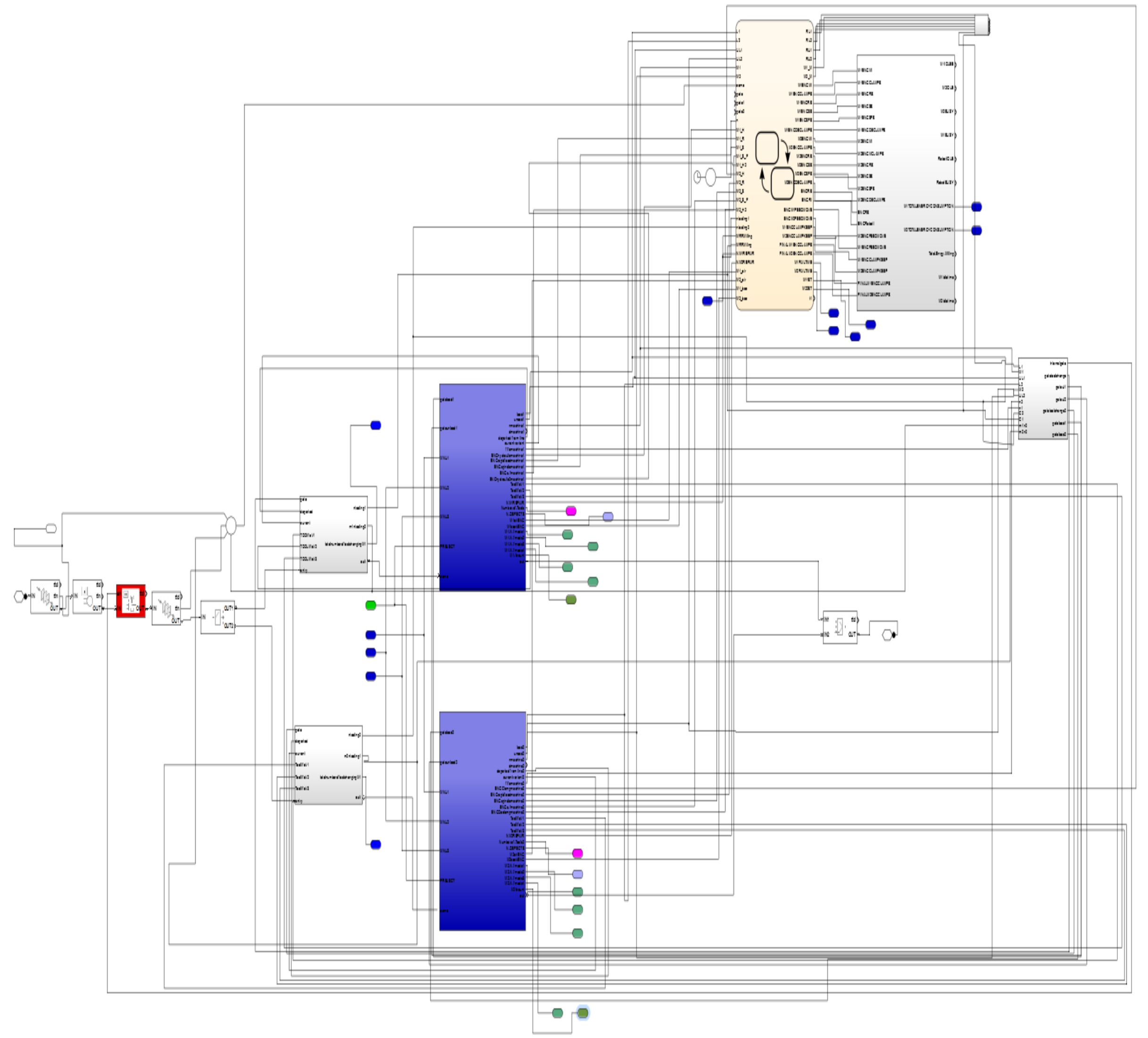


Figure29. Milling station subsystem

Like turning station subsystem, a FIFO block represents the buffer here too. An entity in buffer can get the service, if the robot and at least one machine are idle. If these conditions are satisfied, an enable gate block passes an entity to milling station subsystem. The gate block gets station availability status, from gates control subsystem. If the sending signal from gate control subsystem was positive, gate opens and passes an entity, otherwise remains closed.

Coming part can get the service from machine 1 or 2. There is an output switch after gate block which navigates coming entity to first available machine subsystem.

To achieve a better understanding of milling station subsystem, each mentioned subsystems will be explained separately.

Gates control subsystem

Since one robot serves two machines in this station, there are more logical controllers to check this station availability state, comparing to the first station subsystem. Gates control subsystem, provides controlling signals for gate blocks, based on machines and robot states. The signals control the logical flow of entity in this station.

Like turning station, robot and machines keep sending signal about their states. When machines and robot are idle their sending signal is 0, otherwise it is 1.

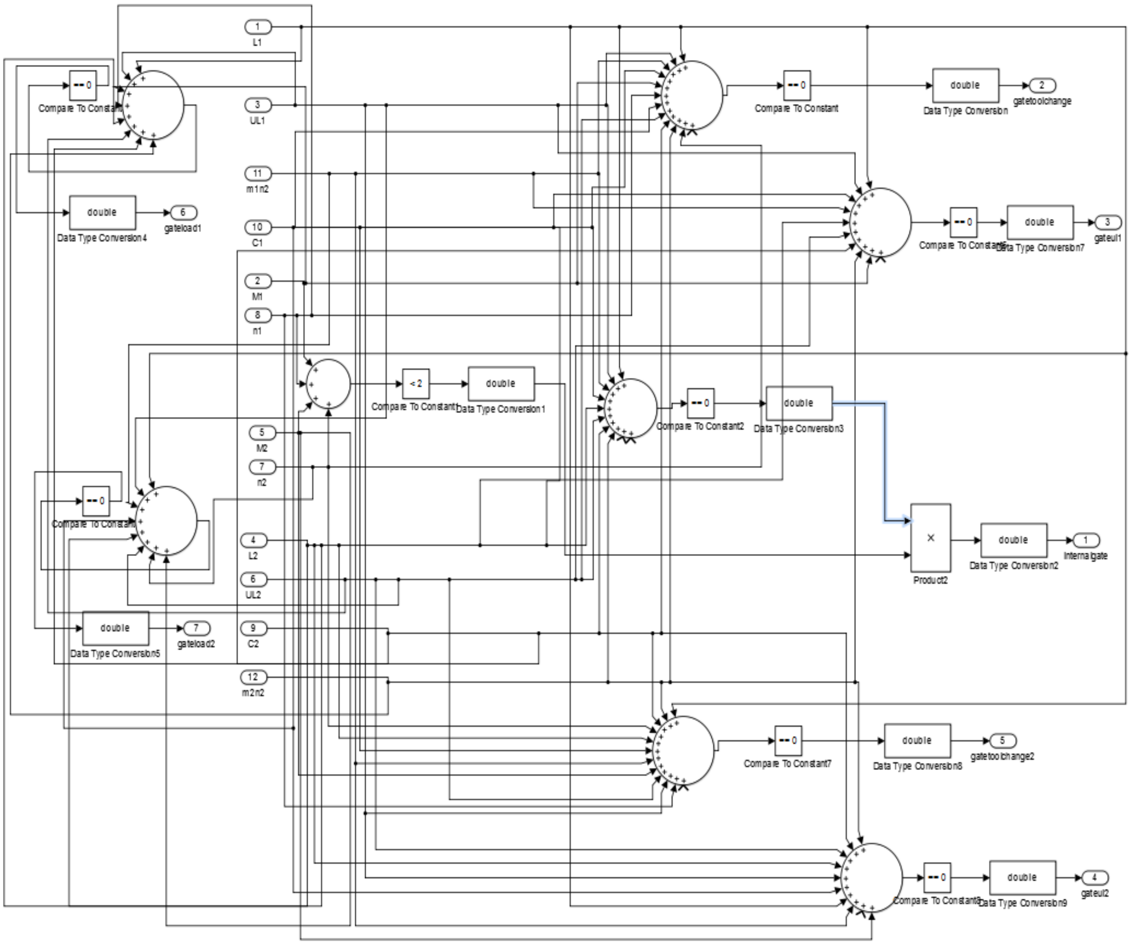


Figure30. Gate Control subsystem

As soon as robot approached to the coming part; it has planned to load it. So, even if setup action is required before loading, robot is not available anymore, since robot plan is already determined. As a result as long as setup action is performing in each machine subsystem, milling station subsystem cannot accept new entity and the gate block should be closed. Setup servers for each machine subsystem, keep sending signals to gates control subsystem about their states. If the server is busy the signal is 1, otherwise it is 0.

Gate control subsystem provides 7 output signals for separate enable gates in different subsystems in milling station subsystem;

1-*inter station gate signal*

A summation block sums these state signals, *machine1 robot loading, machine1 robot unloading,machine2 robot loading, machine 2 robot unloading, machine1 setup,* and*machine2 set up* signals together which are coming from related servers in machining1 subsystem, machining2 subsystem, machine 1 setup subsystem and machine2 setup subsystem. A logical operator compares summation results to zero, if the summation was 0, it returns 1, otherwise 0.

Machine1 state machine signal, machine1 repair signal, machine 1 robot service queue (will be explained)*, machine 2 robot service queue, machine 2 sate machine signal* and *machine2 repair signal* will be summed up together. If the summation was less than, which means at least one machine is free, a logical operator returns1 otherwise 0.

The final outcomes of two above logical operators will be multiplied to each other, if the output was1, the value for *inter station gate signal* is1, otherwise it will be zero.

If the signal was 1; it means that, the robot is free from loading and unloading, the setup action is not performing in any machine, and at least one machine is free. The entity chooses the first available machine subsystem to get milling service.

2-*machine 1 setup gate signal*

A coming part to station can get the service from machine 1, if the robot and machine 1 are available, and setup action is not performing in none of machines.

As explained earlier, there is a switch block after gate block, with two paths connected to each machine subsystem. When a machine is busy, its connecting path should be blocked. The blocking action is performed using an enable gate in first section of each path, and are called; machine1 setup subsystem enable gate and machine2 setup subsystem enable gate.

An operator block sums these state signals; *machine1 repair signal, machine1 machine state, machine 1robot loading, machine2 robot loading, machine 1 robot service queue, machine 2 robot service queue, machine1 robot unloading, machine2 robot unloading, machine 1 setup* and *machine2 setup.* A logical operator compares the summation to zero. If summation result was 0, logical operator returns 1 for *machine1 setup gate* signal, otherwise 0.

This signal will be sent to machine1 setup subsystem enable gate. While this signal is 1, the enable gate is open. If the gate is open, the coming entity to milling station subsystem can choose this machine subsystem to get the service. If this path is blocked the entity should choose second path which is machine2 subsystem to get the service.

3-*machine 1 robot loading gate signal*

When setup time elapsed in setup server, robot loading block should start serving coming entity.

A summation operator sums these state signals; *machine1 repair signal, machine1machine state signal, machine 1 robot service queue, machine 1robot loading, machine2 robot loading, machine1 robot unloading, machine2 robot unloading* and *machine2 setup*. A logical operator compares the summation to zero, if the result was 0, the operator returns 1 for *machine1 robot loading gate signal*, otherwise 0.

This signal will be send to machine1 subsystem robot loading enable gate. While this signal is 1, the enable gate is open and let the robot loading block serves, the coming entity. As soon as machine1 robot loading block becomes busy, *machine 1 robot loading* signal changes to 0, and all described output signals from this station will be zero. So no new entity, cannot inter in to milling station subsystem. When loading time elapsed on loading block, the entity enters to machine1 subsystem and machining cycle starts. As soon as robot loading block released the entity, machine1 robot loading signal changes to 0.

4-machine 1 robot unloading gate signal

When machining operations are finished, if the robot is free, the part can be unloaded from machine to third conveyor. During machine1 operation, the robot can load or unload parts to or from machine 2. If machining operation in machine1 finished, while robot is busy with loading or unloading in machine2, machined part, should wait in machine1, until robot becomes free. A FIFO queue block with 1 size capacity represents this situation. It is placed after final machining operation block to store the entity until the robot becomes free from machine2. This block also keeps sending a signal about its status to gates control subsystem, which is called *machine 1 robot service queue*. It was used in above explained output signals related to enable gates. If *machine 1 robot service queue* signal is 1, it means that there is a machined part in machine1 waiting for robot service. So machine1 cannot serve new part, until this part be unloaded.

An operator block sums these state signals; *machine1 robot loading*, *machine2 robot loading*, *machine1 robot unloading*, *machine2 robot unloading*, *machine1 setup*, *machine 1 robot service queue*, *machine1 repair signal*, *machine1 machine state*, *machine 2 robot service queue*, *machine2 setup*. A logical operator compares the summation to zero. If the summation was 0, the logical operator returns 1 for *machine1 robot unloading* signal, otherwise 0.

This signal will be sent to machine1 robot unloading enable gate. While this signal is 1, the enable gate is open and let the robot unloading block serves the coming entity. As soon as robot unloading block start serving the entity, *machine1 robot unloading* signal changes to 1.

5-machine2 setup gate signal

The logical flow of entity in machine2 subsystem and procedure of processing input signals and making output signals are completely the same with machine 1. So it is containment up to explaining differences in input signals.

A summation operator sums these state signals; *machine2 repair signal,machine2 machine state signal*, *machine 2robot loading*, *machine1 robot loading*, *machine 2 robot service queue*, *machine 2 robot service queue,machine2 robot unloading*, *machine1 robot unloading*, *machine 2 setup* and *machine1 setup signals*. Logical operator compares the summation to zero, if it was0, the logical operator returns 1 for machine2 *setup gate* signal, otherwise 0. This signal will be sent it to machine2 setup subsystem enable gate.

6-machine 2 robot loading

A summation block sums these state signals; *machine2 repair signal,machine2machine state signal*, *machine 2 robot service queue*, *machine 2robot loading*, *machine1 robot loading*, *machine2 robot* *unloading*, *machine1 robot unloading* and *machine1 setup*. A logical operator compares the summation to zero. If the result was 0, the logical operator returns 1 for *machine2 robot loading gate signal*, otherwise 0. This signal will be sent to machine2 subsystem robot loading enable gate.

1-7- machine 2 robot unloading gate signal

An operator block sums these state signals; *machine 2 robot loading*, *machine1 robot loading*, *machine2 robot unloading*, *machine1 robot unloading*, *machine2 setup*, machine1 *setup*, *machine 2 robot service queue, machine1 machine state, machine 2 robot service queue*. A logical compares the summation to zero. If the result was 0, the logical operator returns 1 for *machine2 robot unloading gate signal*, otherwise 0. This signal will be sent to machine2 robot unloading enable gate. While this signal is 1, the enable gate is open and let the robot unloading block serve the coming entity.

Machine1 Setup subsystem

The main structure and logical flow of entities in machine1 setup subsystem is the same with setup sub system in turning station. So main structure and small differences will be explained.

Gate control subsystem sends signal to this subsystem. It clarifies the possibility of performing milling operation in machine1 subsystem (explained above). The setup server also sends a signal to gates control subsystem about its state. If the setup action is performing in *machine1 set up server*, the sending signal is 1, otherwise sends 0.

A logical operator (compare to constant) checks the *machine 1 setup gate signal* and *machine1 setup server* signal. If both signals were zero, the logical operator generates a positive signal and sends it to machine1 setup subsystem enable gate. The positive signal opens the gate and lets the coming entity enters to machine1 setup subsystem.

The need for new tool replacement or new setup is checked with a similar control chart in turning station. The chart uses similar input data from machine1 subsystem.

Life1 or *life2 or life3* like turning station come from machine 1 subsystem and express tool condition for each variant. If each of them equals to zero it means tool life for that variant is not finished, otherwise it should be replaced. If variant type of coming entity was the same with pervious served entity in machine, and tool life signal (*life1*or *life2* or *life3*) was 0, the chart output is 1. If variant type of entity was not the same with pervious served part, or if tool life signal was 1, chart will return 2.

The output of the chart will sent to an output switch block. First path of output switch is directly connected to the machine1 subsystem. Second path is connected to the setup server block. If chart output was 1, the coming entity, will continue its process in machine 1 subsystem directly. If chart output was 2, the setup operation or new tool replacement, should be done first. Like turning there is a set up server block in second path. It gets its service time from *setup time attribute* of the entity.

Machine 1 Machining subsystem

The main structure for machine1, machining subsystem, is the same with machining subsystem in turning station.

As soon as entity enters in machine1 machining subsystem, a get attribute block reads entity variant type. The block sends variant type to machine1 setup subsystem, to compare next variant of next, with this one. Robot loading server block holds the entity, until loading time has elapsed. While the machine 1 robot loading block is busy, robot loading status signal is 1(the used signal in gates control subsystem). When robot loading block service time finished, machining operation will start on machine1, machining subsystem.

To achieve accurate results of energy consumption in machining cycle, this subsystem is broken down to all operational states; machine1 clamping subsystem, machine1rapid move subsystem, machine 1 spindle acceleration, machine 1 air cutting from home position to work piece subsystem, machine1 metal machining subsystem, machine1 air cutting from work piece to home position subsystem, machine 1de clamping subsystem.

These subsystem structures are completely the same with explained subsystem in turning. So, to avoid repetition; the small differences are explained.

Failures can occur during each state of machining in different machine sections. Milling machine is divided to 7 main parts; spindle, table, knee, electrical system, hydraulic system, coolant and saddle. Each above mentioned subsystem generates failure probabilities for each section of milling machine based on exponential distribution. Table11 shows each machine section failure rate (34) (38).

Table11. Turning machine sections and related failure rates

position	λ
Spindle	0.000108
table	0.0003481
knee	0.00008704
electrical system	0.000446
hydraulic	0.0000108
coolant	0.000043522
saddle	0.000043522

Each failure mode can cause failure in each machine section with a specific occurrence probability. Table12 shows each failure mode occurrence probability in milling machine (38) (34).

Table12. failure modes occurrence probability in turning machine

failure mode	Occurrence probability
fuseburnt	0.3085
component damage	0.468
looseness	0.0744
circuit fault	0.1489

In all state subsystems there is a similar flowchart of turning states flowcharts. It checks machine states as being busy, idle or fail state. It provides similar outputs for related subsystems and milling energy flowchart.

Detailed explanation about structure of state subsystems can be found in clamping subsystem of turning station section.

Each subsystem provides different outputs for controlling the flow of entities in system, or to prepare appropriate simulation outputs.

Machine1Machinebusy state output: illustrates machine availability. If the output is 0, machine is available.

Machine1---machine state like *machine1clamping machine state* or *machine1rapid feed machine state*: when machine is busy this output signal is 1 and it is 0 when machine is in fail or idle state. This signal will be used for energy consumption calculation.

Machine1 Repair state output; this output is 1 when repairing action is taking place otherwise it is 0. *Machine1Number of defects* output; this output shows the total number of defects in machine1 in each this machining state.

Failure mode outputs; they show occurrence number of each failure modes.

A tool life control chart similar to turning station determines tool life state in machining subsystem. It receives tool life for each variant from set process parameter subsystem. It also gets variant type of current entity from machining subsystem. The chart sends appropriate output signals like tool life conditions to machine 1 setup subsystem and number of used tools to simulation result subsystem.

Like turning subsystem, process capability influences product quality parameters in metal machining state. Surface roughness and geometry are general quality specifications in milling operations (40). Two control charts (quality chart) are designed, to tag some products out of acceptance interval for each quality parameters (surface roughness, geometry). Quality charts logical structure and procedure of setting quality tags are completely the same with metal cutting in turning station

Milling machine is available, if all operation states are idle. It means *machine1 machine busy state* outputs of all subsystems should be zero. So, all *machine1machine busy state* outputs are summed up together and compared to zero. If summation wasn't 0 the overall *machine1machine busy state* output will be 1. It means machine is not available. If summation was 0 the overall *machine1machine busy state* output will be 0. It means that, machine is available. This output is sent to gates control subsystem. Subsystem checks the signal to open or close gates blocks for entering new entity.

Like turning all *repair state* outputs of all operation subsystems, are summed up and represents the machine repair state. This output will be used to find machine total repair time.

All *Number of defects* outputs from each operation subsystem are summed up and sent to the simulation result subsystem to find total number of defects in milling machine1. *Failure mode outputs* for all subsystems also summed up together to find total occurrence number of each failure mode in milling machine 1.

When all machining operations finished on work piece, and robot is idle, robot unloads the work piece on second conveyor. Robot availability for unloading entities from each machine is investigated in gates control subsystem. If *machine1 robot unloading gate signal* from *gates control subsystem* is 1, an enable gate after de clamping subsystem, opens and passes the entity to machine1 robot unloading server block. Value 1 for *machine1 robot unloading gate signal* means that robot is free and no setup action is performing in this station.

As explained earlier, if operation cycle in machine1 is finished while robot is busy with machine 2, the finished part in machine1 should wait for robot service. A FIFO block after de clamping subsystem with 1 size capacity represents entity waiting state to get the service from robot. The block keeps sending signal, is called *machine 1 robot service queue,* about its holding state to gates control subsystem. If there is a waiting entity there, *machine 1 robot service queue* value is 1. It means, this machine cannot accept new entity until the waiting entity is being unloaded.

When unloading service time is finished, entity leaves the milling subsystem and continues in conveyor 3.

Machine2 Setup subsystem
All components (inputs and outputs) and logical configuration (inter relations subsystem and relation with other subsystems), of machine2 setup subsystem is totally the same with machine1 setup sub system in this station.

Machine 2 Machining subsystem
This subsystem structure is totally the same with machine1, machining subsystem, so to avoid repetition, it containment up with explanation in machine1 machining subsystem.

Path combiner block with two input ports is connected to each machine subsystem output port. It combines released entities from both machines and navigates them to conveyor3 subsystem.

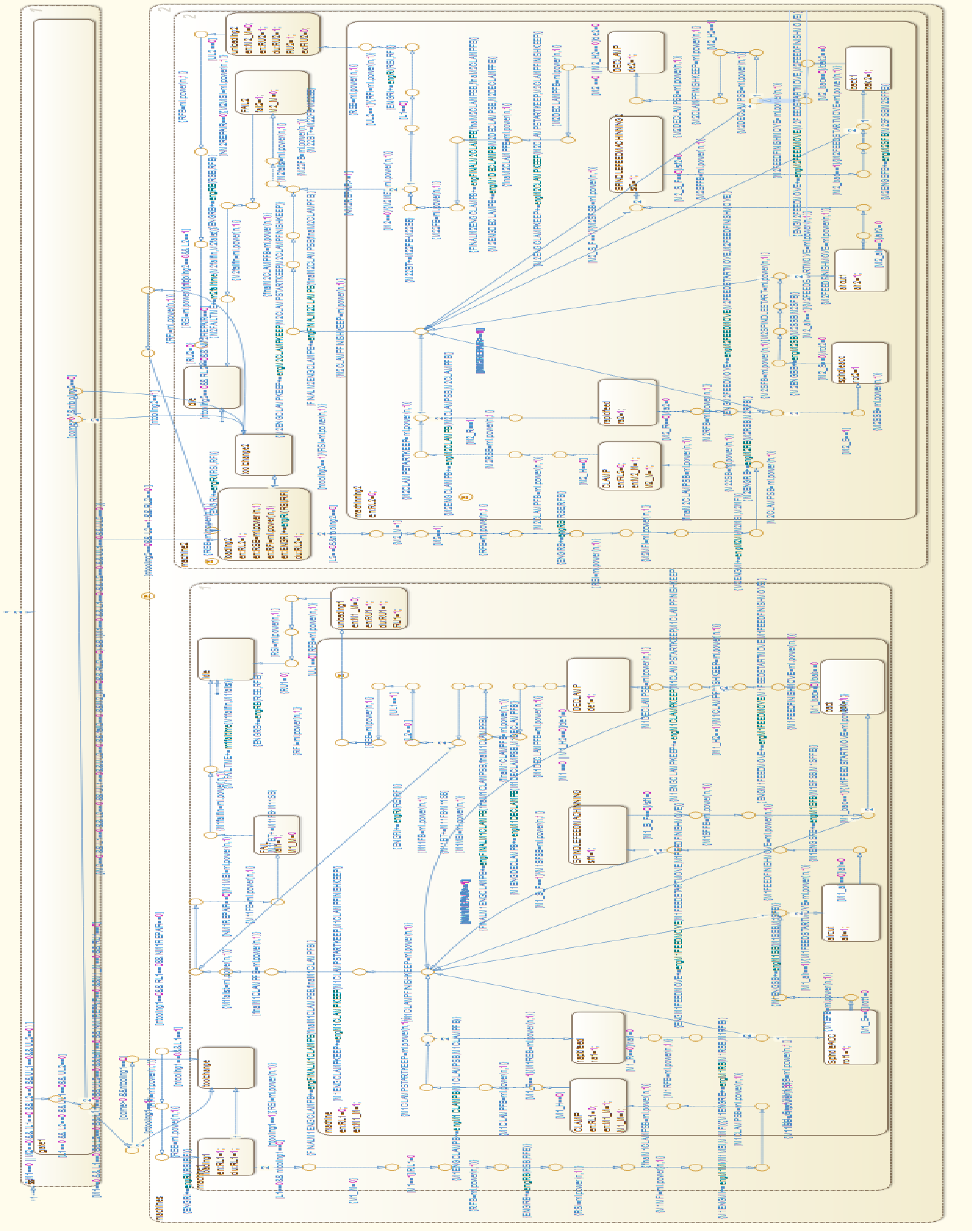
Milling station control Chart
The main task of this chart is to monitor and calculate energy consumption in milling station for two machines separately. It also returns other simulation outputs, which will be explained here.

This chart is the state flow diagram of work piece progress through different states in milling station. Comparing to first station, since there are two machines in this station; this chart includes two main sub states for each machine. Milling chart sub states for each machine also contains all above mentioned states. There are a lot of inputs data, which control transition action from one state to another. One important input is simulation clock, which returns simulation time in any time instant of simulation. Local variables and functions are defined in order to provide appropriate chart outputs. There are some transition actions, which will be executed, while the transition condition from the relevant state is satisfied. The logical transmission in this chart is the same with the logical flow of entities through explained blocks and subsystems.

There are two main states in this chart, station busy and station idle, which represent the overall state of the station.

Station idle state is the default state in this diagram. Leaving condition of this state is availability of robot and at least one machine to serve a new part. These conditions will be investigated by analyzing related sent signals to this chart; *machine1 robot loading*, *machine 1 robot unloading*, *machine2 robot loading*, *machine2 robot unloading, machine1 repair signal, machine2 repair signal,machine1 setup server, machine2 setup server, machine1 busy state* and *machine2 busy state*. The precondition to transition from idle to busy is existence of a part in buffer. As explained before, a FIFO QUEUE represents the buffer in conveyor. It keeps sending a signal to the chart about the number of available entities in buffer block. If this input was more than zero, and *machine1 robot loading*, *machine 1 robot unloading*, *machine2 robot loading*, *machine2 robot unloading*, *machine1 setup server*, *machine2 setup server* signals were zero and summation of *machine1 busy state* and *machine2 busy state* was less than 2, the chart transits to station busy state. Summation results lower than 2 mean that at least one machine is available.

Station busy state includes two main states, machine1 busy and machine 2 busy, with parallel decomposition.



machine1 robot loading, *machine 1 robot unloading*, *machine2 robot loading*, *machine2 robot unloading*, *machine1 setup server* and *machine2 setup server* signals values clarify the chosen machine for running the milling operation on coming entity in subsystem.

Machine 1 and machine 2 busy sub states have completely similar, structures of sub states, transition conditions between states, transition actions, variables and functions, so we sufficient to explain machine1 busy state.

Machine1 busy state consists of 4 main sub states; 1-setup 2- loading 3-machining 4-unloading.

When transition condition of leaving idle state to busy state is satisfied, machine1 setup sub state of machine1 busy state will be activated first. Setup block keeps sending signal about its state to this chart. The chart stays in this state while the block is busy (block sending signal is 1). When the signal changes 0, the charts transits to machine1 robot load sub state. While the machine1 robot loading block is busy, its sending signal is 1. As soon as loading time elapsed in block the signal changes to 0 and transition condition to leave load state is satisfied.

Energy consumption rates used in energy functions in all states beside metal cutting states are educed in previous research and publications in this filed (mostly figures) (18) , (2) .

The next state after machine1 robot load is machine1 machining sub state. Transition actions of leaving machine1 robot loading state to machining state are as below:

Machine1 finish idle local variable updating: As soon as loading is finished, machine starts its operations and finishes its idle state. So mentioned variable gets current time of simulation. It shows the time that machine finished its idle state. Machining sub state includes several sub states, which represent different operational states in machining cycle. Since machining cycle starts with clamping, the first state which will be active in machining sub state should be clamping sub state.

Calling function *of milling machine1 idle energy*: this function subtracts *machine1 start idle* variable from *machine1 finish idle* and multiplies it to energy consumption rate during idle state, and sums it to pervious value. This function helps to calculate machine total energy consumption in its idle states.

*Milling Machine1 idle energy= (Machine1 finish idle-Machine1 start idle)*1.02+ Milling Machine1 idle energy*

The chart sends this value as milling *machine1 idle state energy consumption* output to milling station energy calculation subsystem and simulation result subsystem.

Machine1 start busy variable gets current time of simulation and save it as the time that machine starts operation. *Machine1 Clamp start busy* variable gets the current time of simulation and save it as the time that clamping operation state started.

As mentioned earlier all subsystems of machine1 machining subsystem, provides machine state signals. It is called *machine1 clamp* ubsystem, or *machine1rapid move state signal* in rapid move subsystem. If the machine is busy with related state action, the *machine1---machine state* signal value is1, otherwise it is 0.

Figure31.Milling station flow chart

The chart stays in clamping sub state while the *machine1clamping machine state* signal is 1. When clamping time in clamping server block elapsed, the signal changes to zero and transition condition of leaving this state is satisfied. There are some transition actions of leaving clamping sub state;

Updating *machine1Clamp finish busy* local variable: it gets the current time of simulation and saves it as the time that machine finished clamping action.

Calling function *of machine1clamping energy*: this function subtracts the *clamp start busy* variable from *clamp finish busy* and multiplies it to energy consumption rate during clamping action, and sums it to pervious value.

Machine1 Clamping energy= (machine1 clamp finish busy–machine1 clamp start busy) × (4+1.02) + machine1 clamping energy.

Energy consumption in clamping state consists of two basic sections; first: fix energy consumption which machine needs to run its basic features during clamping, which is 1.02KW. Second: machine needed energy to clamp the work piece. Value *4* in above formula points clamping energy consumption when clamping device suddenly changes from un-hold to hold. This output represents the total clamping energy consumption, and will be sent to the energy calculation subsystem.

The clamping state and its related energy function represent peak value of energy consumption due to momentum effect. As long as clamp device holds work piece, machine consumes moderate value of energy too. This starts just after peak energy consumption in clamping state. To consider this energy consumption, *Machine1clamp start keep* variable gets the simulation current time when clamping state is finished. It represent the time machine starts its steady energy consumption due to part holding.

Before activating rapid move sub state, the chart checks failure event occurrence. As explained before, machine1machining subsystem sends *machine1 repair state* signal to the chart. If the signal is 1, it means machine is in failure (repair) state, and the condition for activating repair state is satisfied. There are some transition actions related to satisfying transition condition to repair state. If failure happens part holding action will be finished (since the machine is turned off during repair action). So the *machine1clamp finish keep* local variable gets the current time of simulation as finishing time of part holding. The *Machine1clamp keep energy function* will be called and subtracts the *clamp start keep* variable from *clamp finish keep* and multiply it to the energy consumption rate during part holding action and sum it to pervious value.

Machine1Clamp keep energy= machine1Clamp keep energy+ (machine1clamp finish keep-machine1clamp start keep) × 0.1

Machine1Fail start time local variable gets the current simulation time and save it as the time that, failure and repair actions started.

Machine1 finish busy local variable gets the current simulation time and save it as the time that, machine busy time is finished.

Milling Machine1 busy time function will be called and subtracts *machine1 start busy time* from *machine1 finish busy* time and sums it up to the pervious value. This output represents total machine busy time, and will be sent to the simulation result subsystem.

Milling Machine1 busy time = milling machine1 busy time + (machine1 finish busy – machine1 start busy)

When all transition actions are performed, the chart transits to repair state. While machine1 repair state signal is 1, the chart stays in repair state. As soon as the signal changed to zero, the leaving condition of repair state is satisfied.

There are some transition actions of leaving the repair state. *Machine1Fail finish time* local variable gets the current simulation time and saves it as the time that repair time is finished. *Milling machine1 fail time* function will be called, and subtracts *Machine1Fail finish time* from *Machine1Fail start time* and sums it to the pervious value. Chart sends this value as an output to the simulation result subsystem which shows total fail time of milling machine1.

Milling Machine1 fail time= milling machine1 fail time+ (Machine1Fail finish time - Machine1Fail start time)

Machine1 start idle local variable gets the current simulation time and saves it as the time that machine starts its idle state.

If no failure had happened during clamping state, machine follows its basic cycle. So the transition condition to next state which is rapid move state here is satisfied. There is a transition action before activating this state too. *Machine1 Rapid feed start busy* local variable gets the current simulation time and saves it as the time that, rapid move action started.

The main logic of transition conditions and transition actions, for the rest of the machining sub states is the same with the clamping state, so to avoid repetition, the states, transition conditions and actions will be mentioned briefly.

Rapid move sub state

Transition condition to next state; *machine1rapid feed machine state signal* equals to 0.

Transition actions to next operation state; *machine1rapid feed finish busy* local variable gets the current simulation time and saves it as the time that, machine1 rapid move action is finished. *machine1rapid feed energy function* updating: this function subtracts *machine1 rapid feed start busy* variable from *machine1 rapid feed finish busy* and multiplies it to energy consumption rate during rapid move state and sum it to pervious value.

Machine1 Rapid feed energy= (machine1 rapid feed finish busy–machine1 rapid feed start busy) × *(5.57+1.02)*

1.02 points to the machine basic energy consumption and 5.57 is milling machine energy consumption rate during rapid move state. This output represents machine1 total energy consumption in rapid move state, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to next operation state (spindle acceleration), is repair signal equals to zero.

Transition actions of leaving this state to repair sate is completely the same with the clamping state.

Transition condition of leaving this state to repair sub state is *machine1repair signal* equals to one.

Transition action of leaving this state to spindle start rotation state is, *machine1spindle acceleration start busy* variable gets the current simulation time and save it as the time that spindle started rotation.

Spindle acceleration sub state

Transition condition to next state; *machine1 spindle acceleration machine state* signal equals to 0.

Transition actions to next state; *machine1spindle acceleration finish busy* variable gets current simulation time and save it as the time that, machine1 spindle acceleration action is finished. *Machine1Spindle acceleration energy function* updating: this function subtracts *machine1spindle acceleration start busy* variable from *machine1 spindle acceleration finish busy* and multiplies it to energy consumption rate during spindle acceleration state and sums it to pervious value.

Machine1 Spindle acceleration energy =machine1 Spindle acceleration energy+ (machine1spindle acceleration finish busy –machine1 spindle acceleration start busy) × (21.6+1.02)

21.6 is milling machine energy consumption rate during spindle acceleration state until it reaches the determined speed. This output represents machine1 total spindle acceleration action energy consumption, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to repair sub state is *machine1 repair signal* equals to one.

Transition actions of leaving this state to repair sate is completely the same with the pervious states.

Transition condition of leaving this state to next operation state (machine1 air cutting) is *machine1 repair signal* equals to zero.

Transition action of leaving this state to machine1 air cutting state is, *machine1air cutting start busy* variable gets current simulation time and saves it as the time that machine1 started air cutting action.

Air cutting from home position to work piece sub state

Transition condition to next state; *machine1air cutting from home position to work piece machine state signal* equals to 0.

Transition action to next state; *machine1 air cutting finish busy* variable gets current simulation time and save it as the time that, milling machine1 air cutting action is finished. *Machine1air cutting energy function* updating: this function subtracts *machine1 air cutting start busy* variable from *machine1 air cutting finish busy* and multiplies it to energy consumption rate during air cutting action and sums it to pervious value.

Machine1 Air cutting energy =Machine1 Air cutting energy+ (machine1 air cutting finish busy–machine1 air cutting start busy) × (0.63+1.02)

0.63 is milling machine1 energy consumption rate during air cutting. This output represents milling machine1 total energy consumption in air cutting action, and will is to energy calculation subsystem.

Transition condition of leaving this state to repair sub state is *machine1 repair* signal equals to one.

Transition actions of g leaving this state to repair sate is completely the same with the pervious states.

Transition condition of leaving this state to next operation state (machining) is *machine1 repair* signal equals to zero.

Transition action of leaving this state to machining state is, machine1 *machining start busy* variable gets current simulation time and save it as the time that milling machine1 started machining action(metal cutting).

Machining sub state

Transition condition to next sub state; *machine1 machining machine state signal* equals to 0.

Transition action to next sub state; *machine1machining finish busy* variable gets current simulation time and save it as the time that, milling machine1, machining action is finished. *Machine1Machining energy function* updating: as concluded in chapter 3, energy consumption in metal cutting state is function of material removal rate and time. MRR value has sent from set process parameters subsystem to milling chart, which is calculated based on machining factors (feed rate, depth of cut and spindle speed). So the energy consumption in this state is calculating, using this formula;

Machine1Machining energy = (-0.001MRR^3 -0.002MRR^2-1.534MRR+64.2 × *(Machine1 machining finish busy- Machine1machining start busy) + Machine1Machining energy*

This output represents milling machine1 total energy consumption in machining action, and will be sent to the energy calculation subsystem.

Transition condition of leaving this state to repair sub state, is *machine1 repair* signal equals to one.

Transition actions of leaving this state to repair sate is completely the same with the pervious states.

Transition condition of leaving this sub state to next operation state (air cutting from work piece to home position) is *machine1 repair signal* equals to zero.

Transition action of leaving this sub state to machining state is, *machine1 back home position start busy* variable gets current simulation time and save it as the time that back home position action started.

Air cutting from work piece to home position sub state

Transition condition to next sub state; *machine1air cutting from work piece to home position machine state* signal equals to 0.

Transition action to next sub state; *machine1 air cutting from work piece to home position finish busy* variable gets current simulation time and save it as the time that, milling machine1 air cutting from work piece to home position action is finished. *Machine1 air cutting from work piece to home position energy function* updating: function subtracts *machine1 air cutting from work piece to home position start busy* variable from *machine1 air cutting from work piece to home position finish busy* and multiplies it to energy consumption rate in back home position action and sums it to pervious value.

Machine1 air cutting from work piece to home position energy = Machine1 air cutting from work piece to home position energy +(machine1 air cutting from work piece to home position finish busy-machine1 air cutting from work piece to home position start busy)×(0.63+1.02)

0.63 is milling machine energy consumption rate during air cutting. This output represents milling machine1 total energy consumption in air cutting from work piece to home position state, and will be sent to energy calculation subsystem.

Transition condition of leaving this state to repair sub state is *machine1 repair signal* equals to one.

Transition actions of leaving this state to repair state are completely the same with the pervious states.

Transition condition of leaving this state to next operation state (de clamping) is *machine1 repair* signal equals to zero.

Transition actions of leaving this sub state to de clamping state are *machine1clamp finish keep* variable gets current simulation time and save it as the time that Machine1 finish holding the work piece (it is finishing the steady state of clamping). *Machine1 clamp keep energy function* updating: this function subtracts *machine1 clamp start keep* from *machine1 clamp finish keep* and multiplies it to energy consumption rate in steady state of work piece holding, and sums it to pervious value.

Machine1 Clamp keep energy= machine1 Clamp keep energy+ (machine1 clamp finish keep-machine1 clamp start keep) × 0.1

0.1KW is milling machine1 steady energy consumption in order to hold work piece. *Machine1 De clamping start busy* local variable gets current simulation time and save it as the time that de clamping action started.

De clamping sub state

Transition condition of leaving de clamp state to next sub state; *machine1 de clamping busy signal* equals to 0.

Transition actions of leaving de clamp state to next sub state; *Machine1de clamping finish busy* variable, gets current simulation time and save it as the time that, machine1 de clamping action is finished. *Machine1de clamping energy function* updating: this function subtracts *machine1 de clamping start busy* variable from *machine1 de clamping finish busy* variable and multiplies it to energy consumption rate during de clamping state, and sums it to pervious value. This energy, like clamping energy, referees to the peak value energy consumption, due to sudden change of clamping device from hold to un-hold.

Machine1 De Clamping energy= (machine1 de clamp finish busy–machine1 de clamp start busy) × (4+1.02) +clamping energy

4 KW points machine energy consumption to de clamp the work piece. This output represents total energy consumption in de clamping state, and will be sent to the energy calculation subsystem.

Machine1 finish busy local variable: gets current simulation time and save it as the time that, machine1 has finished busy state.

MillingMachine1 busy time function updating*:* subtracts *machine1 start busy time* from *machine1 finish busy time* and sums it up to the pervious value. This output represents total milling machine1 busy time, and will is to simulation result subsystem.

Milling Machine1 busy time=Milling machine1 busy time+ (machine1 finish busy –machine1 start busy)

Transition condition of leaving this state to repair sub state is machine1 *repair* signal equals to one.

Transition actions of leaving this sub state to machine1 repair state is the same with others, but it does not include updating *machine1clamp finish keep* variable and calling *machine1 clamp keep energy function*, since the action is already finished.

Transition condition of leaving this state to next state (robot service queue state) is *repair* signal equals to zero.

The transition action of leaving this de clamping sub state to robot service queue state is; *Machine1 start idle* local variable gets the current simulation time and save it as the time that milling machine1 starts its idle state. *Milling Machine1 finish busy* local variable gets the current simulation time and saves it as the time that, machine1 busy time is finished. *Milling Machine1 busy time function* will be called and subtracts milling *machine1 start busy time* from milling *machine1 finish busy time* and sums it up to the pervious value. This output represents total busy time of milling machine1, and is sent to simulation result subsystem.

Milling Machine1 busy time=Milling machine1 busy time+ (Milling machine1 finish busy –Milling machine1 start busy)

If machine2 robot loading, unloading and setup server blocks are idle, the conditions of leaving machine1 robot service queue to machine1 robot unloading state is satisfied.

While machine1 robot unloading block is busy, its state signal is 1. When unloading time elapsed in unloading server, the signal changes to 0 and the transition condition to leave machine1 robot unload state is satisfied. The next state after robot unloading state is machine idle state.

Decomposition, inter sub states, hierarchy, transition conditions and actions ,variables and functions in Machine2 busy sub state, are completely the same with machine1 so we contentment up with machine1 busy state explanation. Released entity from unloading block, continues in conveyor3 subsystem to be stored and finish its production process.

Energy calculation subsystem

This subsystem gets outputs of milling station control chart. It sums up energy outputs for each machine separately. It sends the results as machines busy state energy consumption to simulation result subsystem.

Milling machine1 busy state energy = machine1 clamping energy+machine1 rapid feed+machine1 spindle acceleration energy+machine1 air cutting from home position to work piece energy+machine1 machining energy+machine1 air cutting from work piece to home position energy+machine1 clamp keep energy+machine1 de clamp energy

Milling machine2 busy state energy = machine2 clamping energy+machine2 rapid feed+machine2 spindle acceleration energy+machine2 air cutting from home position to work piece energy+ machine2 machining energy+ machine2 air cutting from work piece to home position energy+machine2 clamp keep energy+machine2 de clamp energy

Total busy state energy consumption for each machine will be added to related idle energy consumption. The result is sent as total energy consumptions of milling machines to simulation result subsystem.

Milling Machine1 total energy = milling machine1 busy state energy + milling machine1 idle energy

Milling Machine2 total energy = milling machine2 busy state energy + milling machine2 idle energy

Inspection subsystem

Inspection process compares products quality parameters with quality specifications, and determines part acceptance or rejection.

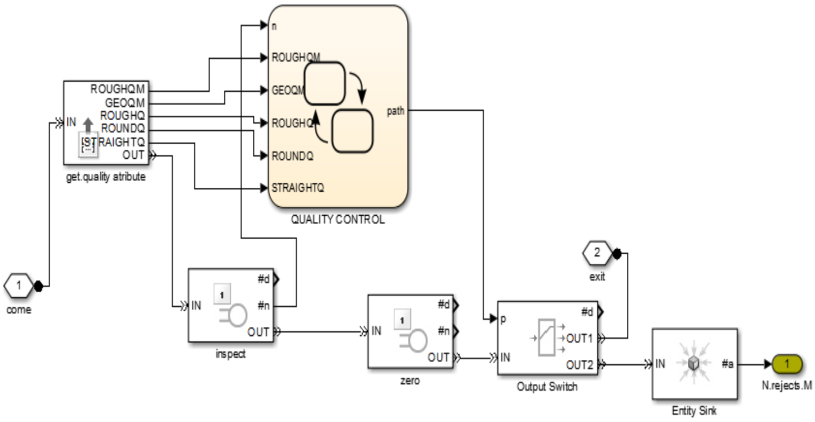


Figure32.Inspection subsystem

It is assumed that inspection is performed in conveyor3, in part transmission from milling station to final product storage. So, it won't impose extra time for inspection action to production process.

In metal cutting subsystems of both milling and turning subsystems, quality parameters have been set as quality attributes for each entity. Surface roughness, roundness and straightness have been defined as turning quality attributes and surface roughness and part geometry for milling (40) (33).Value 1 for each quality attributes means that the part meets that quality parameter specification. A product will be rejected if at least one quality attributes equals to 2.

A get attribute block in inspection subsystem gets all quality attribute values and sends them to Quality control chart. Quality control chart contains two states; accept and reject. Accept state is the default state in quality control chart. As long as the accept state is active in chart, it returns 1(*path* variable). The transition condition of leaving accept state to reject state is at least one quality attributes for coming part equals to 2. By transition from accept state to reject state *path* variable value will change to 2.Transition condition of leaving reject state to accept state is; all quality attributes for coming part equals 1.As soon as accept state is activated again, charts return 1 for path variable.

An output switch with two paths navigates inspected parts to appropriate location. First Output path is connected to conveyor 3 subsystem which passes the entities to storage section. Second path is connected to a sink block which represents defect products storage. If chart output is 1, the inspected entities, passes inspection subsystem and proceed to third conveyor. If the output is 2, the entity considered as defect and will be stored in defect products sink.

Conveyor3

Conveyor3 passes the entities to final product storage block. The structure of third conveyor is the same with the first and second conveyor.

The last block in system, is storage block, which stores all entities. It keeps sending signal about the number of entities which has stored so far. This output total outputs in simulation interval.

Chapter five: Machine Energy consumption analysis in production line

As concluded in pervious sections, machining factors influence on machining state energy profile. They also can effect machine idle times and consequently machine idle state energy consumption amount. In the other hand production planning can effect machine utilization and consequently change machine total idle state energy consumption. In this section effect of machining factors and production planning on system parameters will be investigated.

Machining factor effect analysis

To realize input variables effect on production system parameters, the developed model is simulated for 2working months with different arrangement of machining factors. Simulations results will be compared, to find the effect of input variables on system targets. It helps to find best process parameters combination in production floors. Graph 33 shows results of different experiments by changing factors in both milling and turning station.

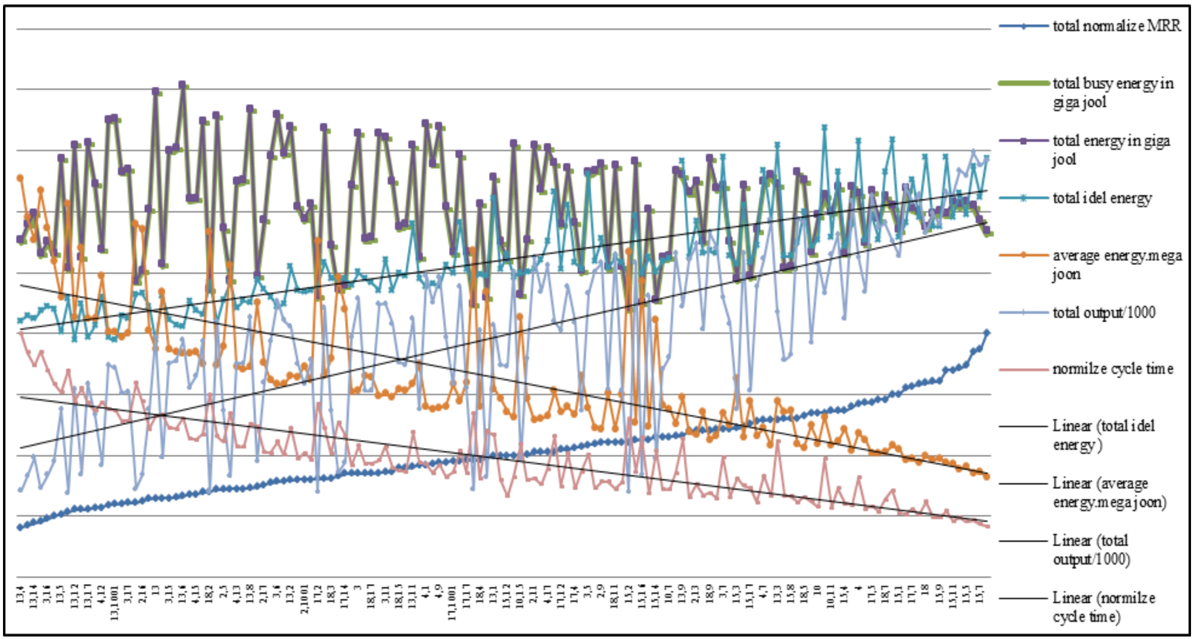


Figure33. Production system outputs with different MRR in milling and turning station

Several indexes are defined to reach more exquisite analysis. Total energy represents total consumed energy in milling and turning station in simulation interval. Machine busy energy consumption, refers to each machine energy usage during its operation time. Idle energy consumption represents machine energy usage when it is in idle state, like while is waiting for loading or setup actions. Average energy consumption per product is calculated by dividing total energy consumption to total output. Total output reflects total produced number in simulation interval.

Observing the graph, by increasing MRR, there is a general reduction trend in average energy consumption per product and cycle time but increasing in total output and idle energy.

In first look there is no obvious tendency in busy or total energy consumption. By dividing the graph to 3 equal sections and finding the average value for each section, the total and busy energy also reduced slowly from first to third section.

Obviously, increasing MRR in whole process, reduces total cycle time, and consequently, increases total outputs. More outputs means more loading and unloading actions. This increases machine idle state, and justifies increasing trend in total idle energy in above graph. In the other hand increasing total output means more machining operation. So it may increase total busy energy consumption. But clearly even it may be hard to associate an obvious reduction trend to total busy energy consumption, it is not possible to call it increasing anyway. Recall chapter3 conclusion, increasing MRR reduces energy consumption in machining state for each per product. By increasing MRR, number of machining operations increases. But its positive effect on machining state energy consumption is so impressive. As a result, by increasing MRR, More operations not only didn't increase total busy energy consumption, but also causes slow reduction tendency. Last three points in graph are good example for this conclusion. By increasing MRR even total output increased, but total busy energy consumption decreased.

Graph clarifies busy state energy as the dominant energy component in total energy consumption. In the other word total energy consumption follows busy energy consumption pattern, and different busy energy profile shifts total energy consumption. As concluded in pervious paragraph, increasing MRR reduces total busy energy consumption. Since busy state energy is influencing pillar of total energy, increasing MRR reduces total energy consumption too. In the other hand MRR increment increases total output too. These all together, leads to definite reduction trend in average energy consumption per product by increasing MRR. If production line aims to produce as much as possible, average energy consumption per product criteria can greatly help to select best machining factors arrangement. But if it has planned to produce specific amount of product, total energy consumption should be considered too.

Different combinations of machining factors in experiments caused fluctuations in graph data series. For example, in experiment EX 2.16, cutting depth, feed and spindle speed for turning and milling are respectively (1, 0.2, 500) and (1, 0.06, 390) for turning and in experiment 4.5 are for (1, 0.2, 425) and (1, 0.06, 331) for milling. Total MRR for these two experiments are proceeding, while more than one input variable have changed. These experiments have different process parameters like tool life, which caused this fluctuation in above figure.

As described earlier, machining factors directly influence tool life.Figure34 demonstrates tool life effect on system parameters.

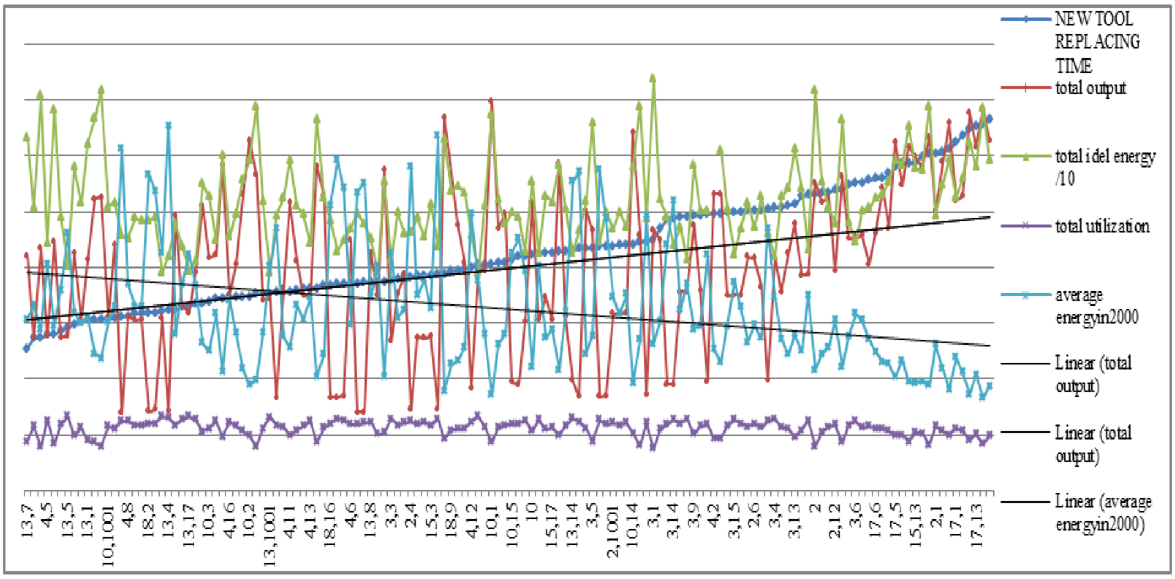


Figure34.Tool life effect on production system

New tool replacement time, represents total spent times for replacing worn tool with fresh one. Total machines utilization refers machines busy time ratio to available time.

By increasing new tool replacement time, average energy consumption shows reduction trend. All data series doesn't show very clear trend. This makes it hard to associate any pattern to them. In order to achieve deterministic conclusion about tool life effect on system parameters, milling and turning are studied separately.

Figures 35 compares experiments where, turning MRR is set to its highest value. Cutting depth, feed rate and spindle speed in turning machine are (2, 0.2, and 500). As a result all differences in results, addresses effect of changing milling process parameters. Similarly figure 36 compares the results when milling had its highest MRR. Cutting depth, feed rate and spindle speed in milling machine are (2, 0.2, and 450). So there is no obstacle from milling process to turning operation.

Figure35.Milling machining factor effect on system energy consumption while turning had highest MRR

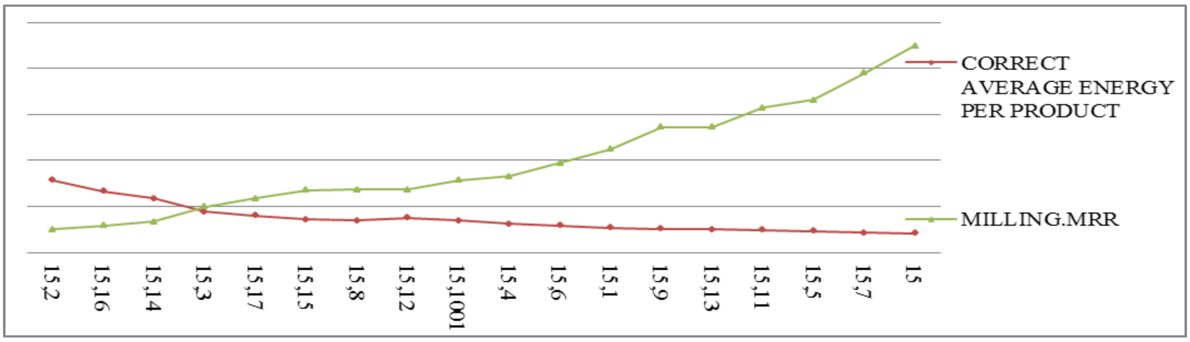


Figure35 displays, Increasing MRR in milling operation decreases average energy per product without any fluctuation which also proves earlier general conclusions.

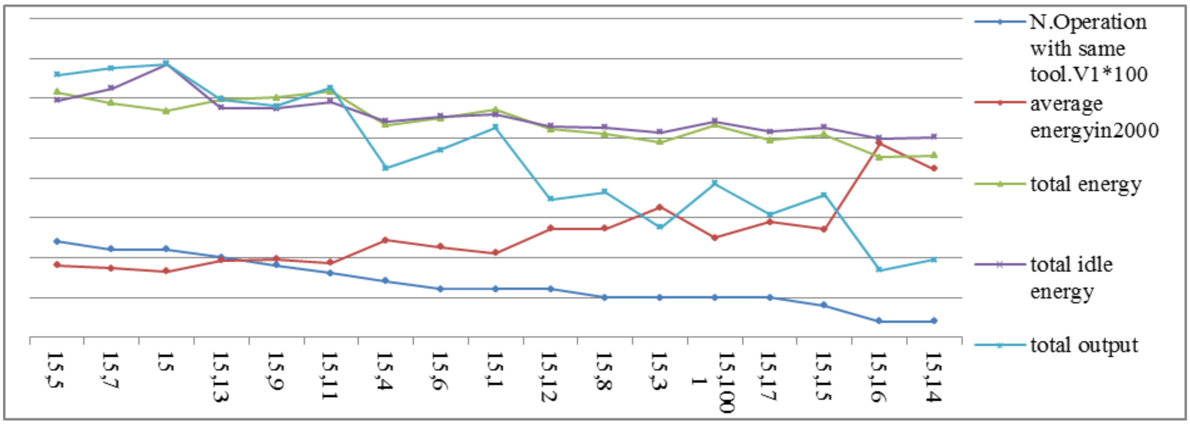


Figure36.Milling Machining factor effect on system outputs while turning had highest MRR

Figure36 draws tool life effect on system parameters. There is no explicit pattern in demonstrated data series by decreasing tool life, and seems this case is too complicated to be analyzed generally, so related data points are compared separately.

For example in EX. 15.5, cutting depth, feed and spindle speed values are (2, 0.2, 331) and in EX.15 are (2, 0.2, 450) and tool life are respectively, 12 and 11. Tool life reduction from EX.15.5 to EX.15, increased total idle energy and total outputs. It also reduced total energy and average energy consumption per product. Same pattern exists in all related points. Tool life reduction causes more new tool replacement actions. This increases machine idle times, and justifies increment in total idle energy. Increment of Machining factors which caused reduction in tool life, simultaneously increases MRR. This also reduces cycle time which increases total output. Recall, MRR increment reduces machining state energy consumption noticeably. As a result, increasing some machining factors reduces tool life, which increases machine idle energy consumption. But positive effect of increasing machining factor, on machining state energy consumption reduction, overcomes it. So increasing machining factors reduce total and average energy consumption per product. Figures 37and 38 compare experiments results where, milling operation had its highest MRR value.

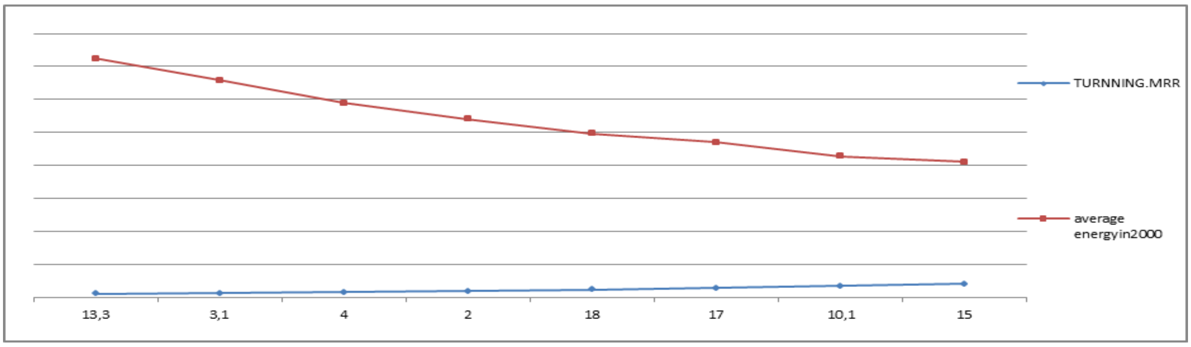


Figure37.Turning machining factor effect on system energy consumption while Milling had highest MRR

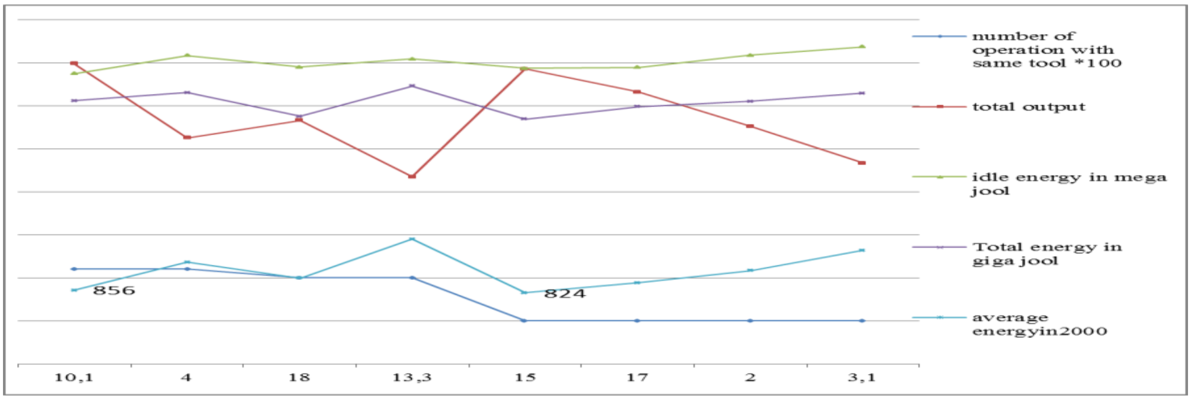


Figure38.Turning Machining factor effect on system outputs while Milling had highest MRR

Demonstrated data series in Figure37 substantiate that increasing MRR, decrease average energy consumption per product.

Like figure38, to achieve accurate analysis, related data points are analyzed separately. In EX. 10.1, turning factors are (2, 0.2, and 425) and in EX.15 are (2, 0.2, and 500) with tool life respectively 11 and 5.

Tool life increment from 5 to 11 from EX.15 to EX.10.1, reduces total idle energy consumption. It also increases total energy and average energy consumption per product. There is same pattern by comparing each two related points in each series. Similar explained logic for figure36 can justify existing pattern here too.

Recall, spindle speed increment reduces tool life. As mentioned in chapter 3, this impact should be studied by considering its interaction with other system parameter. Examples clarify that, by boosting spindle speed, machine idle energy consumption increases. But its remarkable positive effect on reducing energy consumption in machining state reduces total energy and average energy per product. It also reduces cycle time and increases total output. This aggregates its positive effect on reducing average energy consumption per product. So in conclusion, to improve energy efficiency of machine tools in production system, spindle speed also should get its highest possible value.

Production planning effect analysis

In this section effect of batch mode production on energy consumption will be investigated. Graph 39 demonstrates several experiments results with different batch sizes. All experiments had same machining factors and the only difference is in batch size. This difference won't affect machining energy profile. So, machines idle energy consumption and related parameters are compared with different batch sizes.

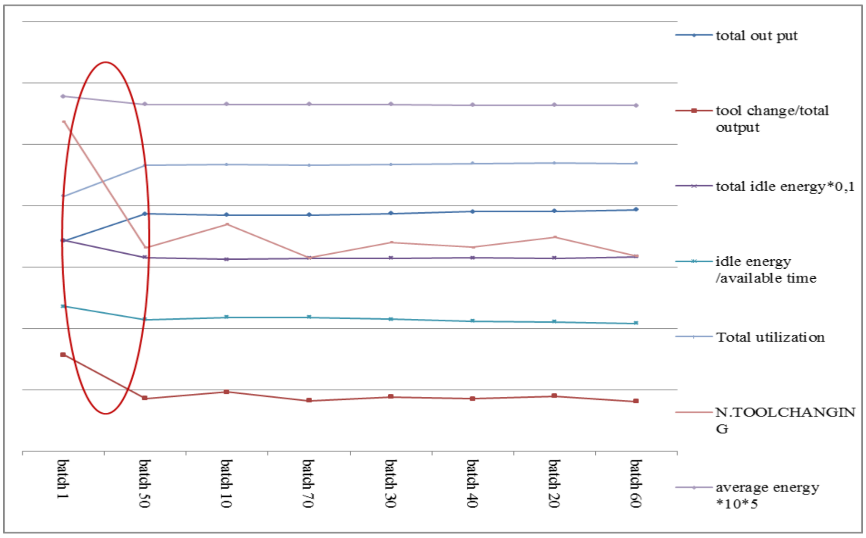


Figure39.Batch mode effect on production system Energy consumption

By increasing batch size from 1 to 10, Considerable decrease in average energy consumption happens. Batch mode production helps reducing set up and tool change actions due to variants difference. This leads to reduction in 'total tool change' and 'total tool change per total output' indexes. This increases machine utilization, and eventuate reduction in machine idle energy consumption. Boosting machine utilization increases total output too. Total output increased with same machining energy consumption profile. These all together, helps to reduce average energy consumption per product.

Maximum tool life among turning and milling operations for 3 variants in investigated experiments is 10 times. All tested batch sizes are absolute multiple of maximum tool life. This is consistent with minimum necessary tool changing actions due to changing worn tool. This caused a considerable positive shift in all system parameters from batch size1 to all other batch sizes. In other word, it helps to take advantage of maximum tool life without changing cutting tool, due to changing variant. If batch size set to a number below of maximum tool life, it causes some tool changing actions due to changing variant while the same tool is still applicable. The small variation in experiments results in different batch sizes can be due to random failure occurrences and repair times, but by deleting these failure events, all follow same pattern. Ultimately it can be concluded that the best number of batch size should be a multiple of maximum tool life in production system.

Chapter Six: Conclusion

Analyzing machine tools energy consumptions with different machining factors, in both milling and turning operations, clarifies Increasing machining factors reduces machining energy consumption.

If two machining factors have high values, reducing third factor, won't increase energy consumption very noticeably. In the other hand if two machining factors have low values, increasing third one, reduces energy consumption very remarkably. This conclusion is quiet helpful, when due to some limitations, some factors should get low values.

All machining factors have same positive effect on machining energy consumption, but they have different effects on tool life.

By increasing machining factors, machine power demand rate increases. Since increasing MRR reduces machining time considerably, it causes reduction in total machining energy consumption for same operation.

In milling operation, cutting depth and feed rate increment has positive effect on tool life but in turning, cutting depth won't change it. In contrast, spindle speed increment has negative effect on tool life in both milling and turning operation. The positive effect of spindle speed increment on machining energy consumption, totally overcome its negative impact on increasing machine idle energy consumptions. So, if energy cost is more important than tool cost spindle speed should be valued as high as possible.

Each factor value, limits another factors selection range. Realizing machining factors same effect on energy consumption but different effect on tool life, helps to find best approach for machining factor selection order. So in turning to increase energy efficiency, the best approach for factors selection is 1-feed rate, 2- cutting depth, 3-spindle speed. Recommended selection order of milling parameters is 1-cuttng depth, 2-feed, 3- spindle speed. All factors should be valued as high as possible.

Machining state energy consumption is the influencing pillar of total energy consumption. As a result, to reduce energy consumption it worth to reduce machining cycle even it increases machine idle stats or equivalently machine total idle energy consumption.

Beside the effect of machining factors on machines energy consumptions, some strategies like batch mode production can significantly reduce total energy consumption. Batch mode production reduces number of tool changing due to any reasons other than, finishing tool life. It increases machine utilization, and decrease machine idle times and idle energy consumption. If batch size has been set to a multiple value of maximum tool life for different product variants, it aggregates its positive effect on energy consumption reduction.

Future research

An important factor on machine tools energy consumption is tool condition. Some research points that, used tools increases machine power demand comparing to fresh ones. So, to achieve more accurate model for energy consumption estimation, it is recommended to consider this factor too. (41)

In some operation, MRR changes in operation cycle when cutting tool have complicated tool path. So in future research in this field it is recommended to adjust model with different MRR value in operation cycle.

Chapter seven

Bibliography

1. *Impact of energy efficiency on computer numerically controlled machining.* **S E Anderberg, S Kara, T Beno.** Proceedings of the Institution of Mechanical Engineers, 2010, Journal of Engineering Manufacture, p. Part B 224: 531.

2. *Unit process energyconsumptionmodels for material removal processes.* **S.Kara, W. Li.** 1, 2011, Manufacturing Technology, Vol. 60, pp. 37-40.

3. *OPTIMIZATION OF CUTTING PARAMETERS FOR CNC TURNED PARTS USING TAGUCHI'S TECHNIQUE.* **DuraiMatinsuresh BABU, MouleeswaranSenthil KUMAR, Jothiprakash VISHNUU. .** FASCICULE3(ISSN1584-2673), 2012, International journal of engineeringTomeX.

4. *Calculation of optimum cutting parameters based on minimum energy footprint.* **P.T. Mativenga *, M.F. Rajemi.** 1, 2011, Manufacturing Technology, Vol. 60, pp. 149-152.

5. *Analysis and Estimation of Energy Consumption for Numerical Control.* **Yan He*, Fei Liu1, Tao Wu, F. P Zhong, B. Peng.** 2012, Proceeding of the Instituation of Mechanical Engineers, Part B:Journal of Mechanical of Engineering Manufacture, pp. 255-266.

6. *http://en.wikipedia.org/wiki/Discrete_event_dynamic_system.*

7. *http://www.answers.com/topic/continuous-system#ixzz2YjKt58Wt.*

8. *http://sakharov.net/fsmtutorial.html.*

9. *http://www.cse.chalmers.se/~coquand/AUTOMATA/book.pdf.*

10. *3 http://en.wikipedia.org/wiki/Stateflow.*

11. *SIMULATION OF HYBRID SYSTEMS.* **Anis SAHBANI, Jean-Claude PASCAL.** 2000.

12. *http://www.scribd.com/doc/37093576/Manufacturing-Process-Material-Removal-Process.*

13. *http://staff.fit.ac.cy/eng.os/AMEM201_LEC9_PROBLEMS.pdf.*

14. *Prediction of tool life in end milling by response surafce methodology.* **M.Alauddin, M.A. El Baradie, M.S.J. Hashmi.** 3, 1997, Journal of Material Processing Technology, Vol. 71, pp. 456-465.

15. *an economic and reliable tool life estimation procedure for turning.* **D.K. Ojha, U.S.Dixit.** 2005, Advance Manufacuring Technology, pp. 726-732.

16. *Tool-wear mechanism in hard turning with polycrystaline cubic boron nitride tools.* **G.Poulachon, A.Moisan, I.S.Jawahir.** 1-12, 2001, Wear, Vol. 250, pp. 576-586.

17. *An Investigation into Fixed Energy Consumption of Machine Tools.* **Wen Li, Andre Zein, Sami Kara, Christoph Herrmann.** Braunschewig : Proceedings of the 18th CIRP International 268, 2011. 18th CRIP International conference on Life Cycle Engineering Technische Universität Braunschewig.

18. *Energy consumption Forecasting and Optimization for Tool Machines.* **A. Dietmair, A. Verl.** 2009, MM science Journal, pp. 62-67.

19. *http://en.wikipedia.org/wiki/Process_capability_index.*

20. *http://en.wikipedia.org/wiki/Process_capability.*

21. *http://www.isixsigma.com/tools-templates/capability-indices-process-capability/process-capability-cp-cpk-and-process-performance-pp-ppk-what-difference/.*

22. **HEIDENHIN.** *Aspects of Energy Efficiency in Machine Tools.* s.l. : HEIDENHAIN, 2010.

23. *Software based tool path evaluation for environmental sustainability.* **Daeyoung Kong, seunghoun choi, Yusui, Sushrut Pavanaskar, David Dornfeld, Paul Wright.** 4, 2011, Journal of Manufacturing systems, Vol. 30, pp. 241-247.

24. *An Environmental Analysis of Machining.* **Jeffery B.Dahmus, Timothy G.Gutowski.** Anhaeim : ASME International Mechanical Engineering Congress and RD&D Expo, 2004. ASME International mechanical Engineering Congress and RD&D Expo.

25. *Automated energy monitoring of machine tools.* **A.Vijayaraghavan, D.Dornfeld.** 2010, Elsevier. Manufacturing Technology, pp. 21-24.

26. *Strategies for Minimum Energy Operation for Precision Machining.* **Nancy Diaz, Moneer Helu, Andrew Jarvis, Stefan Tönissen, David Dornfeld,Ralf Schlosser.** 2009. MTTRF.

27. *Energy Consumption Characterization and Reduction Strategies for Milling Machine Tool Use.* **Nancy Diaz, Elena Redelsheimer, David Dornfeld.** Braunschweig, Germany : s.n., 2011. Proceedings of the 18th CIRP International Conference on Life Cycle Engineering, Technische Universität Braunschweig,.

28. *Predictive Simulation for Model Based Energy Consumption Optimisation in Manufacturing System and Machine Control.* **Anton Diemair,Alexander Verl,Philipp Eberspacher.** 2009, Flexible Automation and Intelligent Manufacturing, pp. 226-233.

29. *Study on Linear Correlation Coefficient and Nonlinear Correlation Coefficient in Mathematical Statistics.* **WANG Ting, ZHANG Shiqiang.** s.l. : Studies in Mathematical Sciences , Vol. 3.

30. *Sustainable machining:selection of optimum turning conditions based on minum energy consideration.* **M.F Rajemi, P.T.Mativenga, A.Aramcharoen.** 10-11, s.l. : Journal of cleaner production, 2010, Vols. 18.PP1059-1065.

31. *Effects of cutting edge geometry, work piece harness, feed rate and cutting speed on surface roughness and forces in finish turning of hardened AISI H3 steel.* **Tugrul Özel, Tsu-Kong Hsu, Erol Zeren.** s.l. : International Journalof Advance Manufactring, 2005.

32. *Analysis of tool wear and surface finish in hard turnning.* **T.Tamizharasan, T.Selvaraj, A.Noorul Haq.** s.l. : International Journal of Advanced Manufacturing Technology, 2005.

33. *The cutting parameters design for product quality improvement in turning operations: Optimization and validation with Taguchi method.* **Chomsamutr. K,Jongprasithporn.S.** Awaji : Computers and Industrial Engineering (CIE), 2010 40th International Conference on, 2010.

34. **S.Saravanan, G.S.Yadava, P.V.Rao.** *Machine Tool Faiure Analysis For Condition Monitoring Application.*

35. *Time between failures model and failure analysis of CNC system.* **ZHANG Hai-bo, J IA Ya-zhou, ZHOU Guang-wen.** s.l. : Journal of Harbin Institute of Technology, 2007, Vol. 14.

36. *Probability distribution of machining center failures.* **Jia Yazhou, Wang Molin, Jia Zhixin.** Northen Ireland : Elsevier Science Limited, 1995.

37. *Field failure database of CNC lathe.* **Yiqiang Wang, Yazhou Jia, Junyi Yu, Shangfeng Yi.** s.l. : International Journal of Quality & Reliability Management, 1999, Vol. 16.

38. *Early Failure Mode Effect and Ciriticality Analysis for CNC Machining Tools.* **JIA Zhicheng, YANG Zhaojun, SHEN Guixiang.** s.l. : Applied Mechanics and Materials, 2013, Vols. 303-306.

39. *Failure probabilistic model of CNC Lthe.* **Yiqiang Wang, Yazhou Jia, Junyi Yu, Yuhua Zheng, Shangfeng Yi.** s.l. : Elsevier, RELIABILITY ENGINEERING & SYSTEM SAFTY, 1998.

40. *Optimization of cutting parameters in machining surface to reduce errors.* **Wu Gang, Zhang Mianhao.** Wuhan : Computing, Control and Industrial Engineering (CCIE), 2011 IEEE 2nd International Conference on, 2011.

41. *Investigation of cutting tool influence on machine tool power consumption.* **Sabahudin Ekinvic, Edin Begovic,Irene Buj Corral, Dzavid Alicic.** Prague, : s.n., 2011. 15th International Research/Experet Conference 'Trends in the development of Machinary and Associated Technology'.