CHAPTER 4

ORGANIZATION OF MODULES AND FORMS IN THE SYSTEM

4.1 INTRODUCTION TO THE MODULES

In the present work, the User Interface for all the modules in the system has been developed using Windows Forms available in Microsoft Visual Studio 2008. The functionality has been captured and coded using C# language and the database has been created in SQL Server 2005.

The system works with a simple and smooth flow in a user-friendly interactive environment. The different forms/ capabilities & output of the system have been highlighted by taking number of sample components. The system comprises of 3 modules.

1. Product Expert module
2. Intelligent System for Manufacturing Information (ISMI) module
3. Scheduling module

The organization of the various modules and forms in the system has been shown in the Fig 4.1.
Fig 4.1 Organization of the Modules and Forms

The Product Expert module helps to provide technological data to the intelligent system. The forms in the intelligent system have been designed for the prismatic, cylindrical, taper, complex and interacting features. Each form in the intelligent system gets populated with data once the object containing the features gets recognized. The output gets displayed in the form of a cost estimate, part program, process plan and "STEP+" file. A Scheduling Module enables the operations suggested by the intelligent system to be scheduled taking into consideration the capacity utilization of the shop floor. A Help section has been developed to provide the user with basic help regarding the STEP file, steps to get the required output etc.
In the sections to follow, the different modules along with the constituent forms have been explained. Also, the content displayed by the forms, the background functions (coded in C# language) used to get the output, basis of the calculations etc., have been briefly explained taking a sample component.

### 4.2 Programming Concepts of Product Expert Module

Technological data is required by any intelligent system to cater to the requirements of the industry. Technological data consists of data associated with Material properties, Machine and Tool parameters, Cutting speed, feed and associated data, Machine, Tool, Material & Overheads data along with sales & customer data. This data is authenticated & fed into the system using the Product Expert module. The Product Expert has access to all data related to Material/company Overheads, Tool Database & Parameters, Machine Database, Customer Database, Cutting Speed and Feed data based on machine capabilities. This data is collected from user input as well as a reliable & validated Knowledge base of technological data such as data handbooks. The user interface allows the product expert to enter the required data, modify the data and delete the obsolete data as and when required.

The Product Expert module consists of 6 forms/ sub-modules using which data can be entered/ retrieved and manipulated (shown in Fig 4.2). They are:

1. Material Properties
2. Cost Details
3. Cutting Speed and Feed
4. Tools Data
5. Machine Data
6. Customer Information
7. Online Help

Fig 4.2 Product Expert Module

The Login screen helps the user to log into the Product Expert module. The screenshots of the 6 forms along with their functions have been discussed in the ensuing sections.

4.2.1 Material Properties

The Material properties form takes as input the various properties of materials. For the present work few properties have been considered as shown in Fig 4.3.
The material data is entered into the product expert module by the function InsertData(). The ModifyMaterialProperty() function takes Material name as input to modify the properties of the material. The material and its properties are removed from the database by the RemoveMaterial() function.

The Help button activates the Online Help form which takes the input search string and displays the results in www.google.com using a Mozilla browser. This feature has been made available in the Cutting speed and Feed form as well as Tools data form.

### 4.2.2 Cost Details

The tool and material cost can be input along with the company overhead (%) details as shown in Fig 4.4. These will be input for manufacturing cost calculation.
The tool cost can be entered for the tool number which has already been entered in the tools database. The tool number is checked by the function CheckToolNo(). The material cost can be entered for the material name which has already been entered in material properties database. The material name is checked by the function CheckMaterialName(). Similarly the overheads can be entered into the database. All these entered data can be modified accordingly.

### 4.2.3 Cutting Speed and Feed

The Cutting speed, Feed and Depth of Cut (DOC) values for Drilling, Reaming, Turning, Milling and Grinding can be input using the Cutting speed and Feed form as shown in Fig 4.5. The data of Cutting speed, feed and DOC for machining processes that has been taken and compiled from various sources has been shown in Appendix C.
The function `CheckWorkMaterialExists()` checks the presence of the material name in the database to insert the cutting speed values for drilling and reaming. The values are inserted and cleared by the functions `InsertDrillReamCuttingspeed()` and `clearDrillReamCuttingSpeed` respectively. The feed for drilling and reaming is inserted into and cleared from the database using the function `InsertDrillReamFeed()` and `ClearDrillReamFeed()` respectively.

The cutting speed, feed and DOC for turning are inserted by the function `InsertTurningCuttingSpeed()` after checking the existence of the work material and tool material in the database.
Similarly the cutting speed and feed are inserted into database by the function InsertMillingSpeedFeed() after checking the existence of work material, tool material and selection of milling type.

In case of grinding, the peripheral speed, DOC and longitudinal feed ratio are inserted into database by the function InsertGrindingSpeed() after checking the existence of work material, tool number and selection of grinding type.

**4.2.4 Machine Details**

The machine details available on the shop floor along with the individual operation cost on hourly basis can be entered using the Machine details form as shown in Fig 4.6.

![Fig 4.6 Machine data](image)

The function InsertMachineData() inserts the machine details along with the operation cost into the database after checking the machine number. The machine number is checked by the function CheckMachineNo() so as to avoid duplication of machines in the database.
4.2.5 Tool Data

The tool parameters, depending on the operation type can be entered using the Tool data form as shown in Fig 4.7. All the individual tool parameters are marked on the tool image thus making the data entry easier for the product expert.

The tool details are fed into the database depending on the operation type, tool number and machine number. A tool has a unique number and is assigned to a particular machine by functions called CheckToolNo() and InsertDrillToolData() for drilling operation. Any modifications to the data can be effected by the function ModifyDrillToolData() after checking the tool and machine number using functions SelectToolNo() and SelectMachineNo(). The tool details
can be cleared from the database using function RemoveDrillToolData() taking Tool number and machine number as the input.

An entire database of tools supporting the following operations has been created.

a) Drilling  
b) Centre drilling  
c) Reaming  
d) End milling  
e) Dovetail milling  
f) T-slot milling  
g) Turning  
h) Grinding

4.2.6 Customer Information

The customer information along with sales details w.r.t each customer can be entered in the Customer information form as shown in Fig 4.8.

![Customer Information Form](image)

Fig 4.8 Customer Information

The customer data and part data is entered into the database using the functions InsertCustInfo() and InsertPartInfo().
4.2.7 Online Help - Novel Feature

In Material properties, Tool data and Cutting speed and Feed forms an additional feature has been provided which enables the product expert to access help regarding any topic from online resources using the Google search engine and an internet browser. The advantage lies in the fact that the online search can be done from the form itself, without having to exit the application. The search string can be entered in text box as shown in Fig 4.9 and the results of the search are displayed in the browser as shown in Fig 4.10.

![Fig 4.9 Online Help - Entering the search string](image)

After the search string is entered, at the click of the button the search results are displayed by the Google search in the default browser by the function:

Fig 4.10 Mozilla browser displaying the search results
4.3 PROGRAMMING CONCEPTS OF INTELLIGENT SYSTEM FOR MANUFACTURING INFORMATION (ISMI) MODULE

The ISMI is a system that automatically generates and integrates the entire product data on a single platform. It integrates the following sub-modules/forms:

1. STEP File Processing
2. Object Details
3. Time-Cost Calculations
4. Cost estimate
5. Process plan
6. Part program

The input-output model of the ISMI has been shown in Fig 4.11. The screenshots of the 6 sub-modules/forms along with their functions have been discussed in the ensuing sections. A sample component "Plate with a hole" has been taken to explain the functionality of the sub-modules/forms in ISMI.
Fig 4.11 Input Output Model of ISMI
### 4.3.1 STEP file decoding

The STEP file decoding form takes as input the STEP (.stp) file of the CAD model as input. The STEP file is a text file which contains the geometric data of the 3D solid model. The data is in the form of Cartesian coordinates, Edge curves, Control points, Generated cylinder, Conical surface etc. All these strings contain some numerical values which need to be identified and extracted. The actual Feature Recognition process takes place in this sub-module. Numerous functions have been utilized to code the Feature recognition process which gives the feature data as output. The Fig 4.12 shows STEP File processing form for Plate with a hole.

![STEP File Processing - PART1-PLATE WITH HOLE](image)

**Fig 4.12 STEP file decoding**

The function line.contains() checks the presence of the string "CARTESIAN_POINT,('Axis 2P3D Location'," and then by using further substring functions, gets the values of the hash codes, X, Y, Z
coordinates of each Cartesian point. These values are stored in database.

Another function line.contains() checks the presence of the string "CYLINDRICAL_SURFACE('generated cylinder'," and then by using further substring functions, gets the values of the hash code for the centre of the cylinder, the (X, Y, Z) coordinates of the centre of cylinder and the radius of cylinder.

Then, another function line.contains() checks the presence of the string "DIRECTION('Axis2P3D Direction','" and then by using the checkDirection() function, finds the direction vector value to be Z=-1, indicating that the cylinder is a hole (subtracted volume).

Finally the function line.contains() checks the presence of the string "CIRCLE('generated circle',," and then using the count() function the count of circle strings is taken and verified.

The STEP file processing form displays the strings discussed above in separate panels using list boxes.

4.3.2 Object details

The Object details form displays the geometric data of the object and the Feature indentified, including dimensions of individual features along with volume, density, weight and process capabilities. The Object details form for Plate with a hole is shown in Fig 4.13.

The function LoadCoordinates() calculates the Maximum of X, Y and Z coordinates and displays as the Length, Width and Thickness of the plate. The function FindDepth() finds and displays the depth of the
hole. The FormLoad() function populates the value of radius of the hole. Also, the picturebox tool displays the image of the object.

The functions FindVolumeOfHole() and FindVolumeOfObject() calculates the volume of the hole feature and the object i.e. plate with a hole.

The functions GetMaterialName() and GetMaterialDensity() populates the material types (by taking data from Product Expert database) in the combo box and on selection of the material type, calculates and displays the weight of the object.

The details regarding the manufacturing processes and their sequence, needed for the feature is displayed to the user based on the technological information submitted by user in the form of Process Capabilities. Based on the values entered by the user, the function CheckProcessCapabilities() verifies the entered values with the stored values for the capabilities and recommends the operations and their sequence.

The Process Capabilities entered by the user decides the operations necessary to manufacture the feature of the component. Ex: In this object, since the Tolerance value entered by the user is 0.009 mm, Surface Finish is 0.0009 mm, True Position 0.27 mm & Roundness 0.015 mm, Both Drilling & Reaming processes have been recommended.
4.3.3 Time Cost calculations

The Time-Cost data for the recommended processes gets generated in this form. The formulae and reference values used for machining time and cost calculations of different processes have been discussed in detail in Appendix C.

The functions like getMachineName(), getMachineNo(), getToolName(), getToolNo() fetches the machine and tool data from the Product Expert database and populates the form. Also the tool parameters like tool cost, diameter, tip angle gets populated in the
form when the tool number combo box click() event gets activated.

Further depending on the material type and the diameter of the hole the range values for Cutting speed and Feed gets fetched from the Product Expert database and is displayed to the user, thereby enabling the user to enter the correct value for Cutting speed and Feed for machining.

The function calculateTotalDepth() calculates the depth which the drill tool and reamer have to travel to complete the machining. Based on the values of cutting speed entered by the user, the RPM is calculated by the function CalculateRPM().

The Manufacturing time calculation is carried out by the CalculateManufacturingTime() function which takes a input the auxiliary, delay, setup and machining time. The function is executed in the Leave() event of the auxiliary, delay or setup time text boxes.

The Cost of the object along with the entire order is calculated by the functionality written in the Click() event of the Calculate button in the Time-cost form. The cost values w.r.t raw material, machining cost/ hour etc., have been taken from the Cost database in the Product Expert module.

The Time-Cost data based on Drilling and Reaming processes has been shown in Fig 4.14. From this sub-module/ form, the Cost estimate, Part Program and Process Plan for the product can be generated.
4.3.4 Cost estimate

The Cost estimate as shown in Fig 4.15 takes input from the user, Customer Database as well as from the Time-cost calculations form to calculate & display the values. The parameters and formulae used to calculate product cost has been discussed in Appendix C.

The Cost estimate gets generated by the Click() event of the Quotation button provided in the Time-cost calculations form. The customer details are fetched from the Customer information database. The tax rates are entered by the user and the final price is arrived and displayed in the Total price text box.

The Cost estimate provides the format to enter other standard Terms and conditions like quality standards, payment terms, validity of Cost estimate etc. Other statutory details can also be entered. The Cost estimate can be saved to database or as a PDF file, thus enabling it to be attached to an email and sent to prospective customers. A
print option has been provided to take the printout of the Cost estimate.

Fig 4.15 Cost estimate

4.3.5 Process Plan

The Process Plan for Plate with a hole can be generated by taking inputs from Time-Cost & Object Details forms. A standard process plan format has been developed taking into consideration part requirements, raw material selection, manufacturing operations recommendations, machine selection, tool selection, selection of manufacturing conditions i.e. cutting speed, feed and depth of cut and manufacturing time based on machining, delay, setup and auxiliary time. The Process plan is shown in Fig 4.16.
On the Click() event of the Process plan button the Process plan gets generated. The details like part name, number, machine name etc., gets displayed.

The functions corresponding to the processes get executed and the parameters get populated in the Datagridview data holder thereby displaying the process plan. In this case 2 functions namely InsertDrillingProcessPlanner() and InsertReamingProcessPlanner() gets executed and the values of the machining parameters gets populated in the Datagridview data holder. The other text boxes get populated with the addition of the drilling and reaming times for delay, auxiliary, machining etc., and finally displaying the total operation time.

The picture box tool displays the image of the object.

Fig 4.16 Process plan
4.3.6 Part Program

The part program has been generated using tool path data generated from geometric data of the feature. The appropriate G&M codes have been selected for FANUC controller. The process sequence and cutting parameters make up for the remaining data required to generate the part program.

On the Click() event of the Part program button the part program gets generated in a Rich Textbox arranged block by block. The initial blocks are standard for any program highlighting the billet size, tools used, units followed etc. Also the tool change procedure has to be executed for every new process. At the end of machining, the spindle has to be stopped and the tool should to be taken back to home position followed by program end. These functions have been taken care by the coded logic.

The functionality for machining is written for canned cycles or for basic linear and circular interpolations. In this case, the canned cycles G73 (drilling) and G85 (Reaming) have been used and the parameter values for speed, feed, DOC, total depth etc., have been taken from the Time-cost form.

The part program can be saved in a .txt file or a print can be taken.

The Part program (G & M codes) for the object is shown in Fig 4.17. It has been generated based on FANUC controller, since a majority of manufacturers use the FANUC controlled CNC machines.
4.4 **SCHEDULING MODULE**

The processes in the Time-cost calculations form can be scheduled using this module. The data required for scheduling include:

- part name and number
- operation type
- time taken
- machine name and number
- tool number

This data is made available in the Scheduling module by click() event of the Scheduling button on Time-cost calculation form. The function `InsertToSchedulingDrilling()` and `InsertToSchedulingReaming()` get executed and the data gets added to
the scheduling database, which can be accessed by the Scheduling module.

The capacity of the shop floor in terms of the number of machines and their capabilities has been coded in the logic of the Scheduling module.

The user can select the operation & schedule it at a particular time on a particular machine. On the click() event of the OK button 2 functions get executed.

- CheckScheduledTask() - which checks whether or not, any other operation has been scheduled at the same time on the same machine.
- InsertScheduledTask() - schedules the operation to the machine at the selected time and this data gets updated in the Status Report.

The system allows the operation to be scheduled provided the selected slot is "free", else a "machine busy" message gets displayed. A success message is displayed for the operations which get scheduled. The Scheduling process has been shown in Figure 4.18.
Another feature called the "Machine Status" has been provided, with the help of which the status of the machine at any point of time can be checked. The function GetScheduledDetails() gets executed on the click() event of the View Graph button. The machine status (engaged/ free) can be observed in the status chart as shown in Fig 4.19. The ticks in the check boxes indicate the *allocated slots* whereas the blank check boxes indicate the *vacant slots*. This can be taken as a basis for optimal capacity utilization on the shop floor.
The modules in the system have been designed & developed meticulously so as to capture, store & exchange all the geometric and technological data required to smoothly carry out the machining, sales, cost, time, process planning and scheduling functions of product.