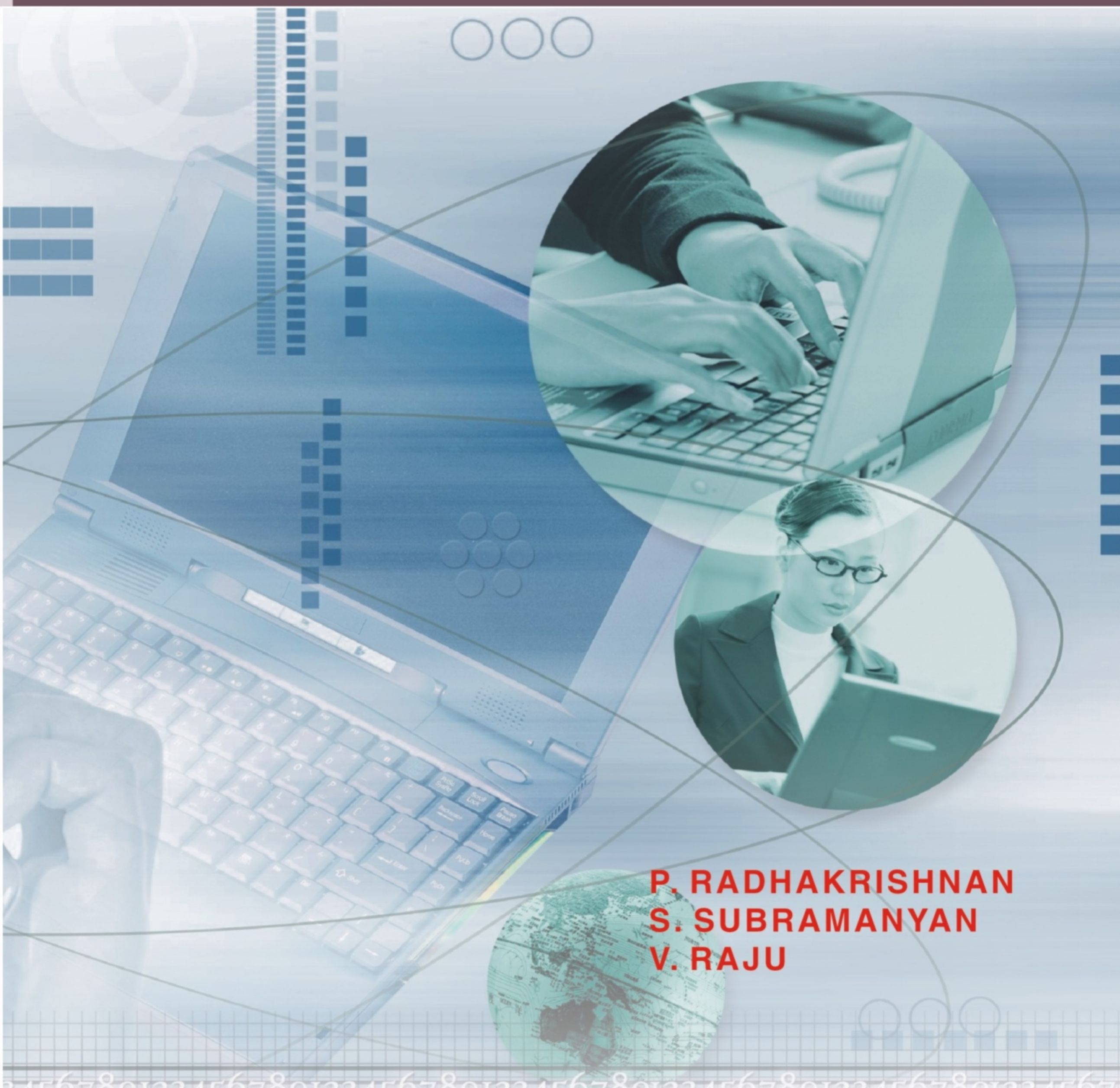
NEW AGE

THIRD EDITION

CAD/CAM/CIM



P. RADHAKRISHNAN
S. SUBRAMANYAN
V. RAJU

NEW AGE INTERNATIONAL PUBLISHERS

CAD/CAM/CIM

(THIRD EDITION)

This page intentionally left blank

CAD/CAM/CIM

(THIRD EDITION)

P. RADHAKRISHNAN
Vice Chancellor
VIT University, Vellore
Tamil Nadu

S. SUBRAMANYAN
Former Principal
PSG College of Technology, Coimbatore
&
Former Vice Chancellor
Bharathiar University, Coimbatore
Tamil Nadu

V. RAJU
Dean
School of Engineering Technologies
Farmingdale State University
New York

PUBLISHING FOR ONE WORLD

NEW AGE INTERNATIONAL (P) LIMITED, PUBLISHERS
New Delhi • Bangalore • Chennai •Cochin • Guwahati • Hyderabad
Jalandhar • Kolkata • Lucknow • Mumbai • Ranchi
Visit us at **www.newagepublishers.com**

Published by New Age International (P) Ltd., Publishers

ISBN (13) : 978-81-224-2711-0

PUBLISHING FOR ONE WORLD

NEW AGE INTERNATIONAL (P) LIMITED, PUBLISHERS
4835/24, Ansari Road, Daryaganj, New Delhi - 110002
Visit us at **www.newagepublishers.com**

PREFACE

Manufacturing managers and engineers are ever concerned with improvement in quality, reduction in both manufacturing cost and delivery time. The globalization of economy requires introduction of new products with enhanced features at competitive costs. Another challenge is the reduction in product life span. This necessitates considerable time compression in product development cycle. Yet another significant trend is mass customization which calls for extreme flexibility in manufacturing. The massive outsourcing in manufacturing is another important development in recent years.

The new edition of CAD/CAM/CIM has been bought out to focus on the response of CIM technology to address to these challenges. Manufacturing in the new millennium is moving towards more and more sophistication in exploiting the capabilities of computer hardware and software. Robust design methodologies and integration of shape design and functional design are included in the present edition. Optimized manufacturing is a possibility now with the extensive use of FEA. Apart from design optimization, FEA is used to model and simulate complex manufacturing processes to evolve several iterations. This enables engineers to make right parts first time every time. An additional chapter on simulation softwares has been added in the present edition to introduce this powerful tool to the students.

The authors would like to acknowledge the contribution of our erstwhile colleagues in the PSG CAD/CAM Centre as well as Krishnaveni and Sasikala in word processing the earlier editions and Govindaswamy for helping with some chapters in the present edition. Acknowledgements are due to K.J. Reddy for providing some models for reproduction in this edition and to Pradeep for critical suggestions. The excellent support and encouragement extended by Padmini, Anitha and Hari during the revision of this edition is gratefully acknowledged.

P. Radhakrishnan
S. Subramanian
V. Raju

This page intentionally left blank

Contents

How to Use This Book

This book is an examination study text on Computer Integrated Manufacturing (CIM). It is organised around three syllabus modules and follows the terminology, classifications and sequence of the supplied course material.

The first module (numbered Chapter 3) deals with CAM programming: manual part programming, computer assisted part programming, automated programming of CNC machine tools, machining of free form surfaces, automatic feature recognition, tolerance based machining and knowledge based machining. The second module (numbered Chapter 5) deals with computerized manufacturing planning and control: computer aided process planning, retrieval and generative systems, computer integrated production management, shop floor control and computer aided quality control. The third block (numbered Chapter 6) deals with the integration of CAD, CAPP, CAM and CNC through STEP and STEP-NC, and with intelligent manufacturing: artificial intelligence and machine learning in CNC machining, the intelligent autonomous CNC machine tool, real-time machine monitoring, real-time CAM simulation, digital twins, virtual manufacturing, knowledge based engineering and Industry 4.0.

Each major section ends with a box of *Key Examination Points*. Definitions, examples, comparison tables and procedures are numbered automatically. Where the source slides gave a diagram, the diagram has been redrawn or converted into a table so that its information is retained. A second pass has enriched the chapters with material on cutter contact and cutter location data, post processors and machine kinematics, virtual machining, production monitoring and the ERP, MES, SCADA and PLC hierarchy, computer aided inspection planning, machine learning paradigms, fault diagnosis and remaining useful life, cyber-physical systems and product lifecycle management.

Chapter 3

CAM Programming

3.1 Introduction to CAM Programming

Computer Aided Manufacturing (CAM) programming is the activity of preparing the coded instructions that drive a numerically controlled machine tool so that it produces a part from a raw workpiece. The instructions are collected in a *part program*. The part program is a set of statements that a machine control system can interpret and convert into signals which move the spindles and drive the machine tool. Today the part program can be generated directly from the CAD database by NC software, and the result becomes the input for an NC machine tool.

Definition 3.1 (Numerical control)

Numerical control (NC) refers to the use of coded numerical information in the automatic control of equipment positioning. It has to do with the motion of cutting tools, or of the part against a rotating tool, with positioning, inserting and similar actions. The production steps are stored in a part program. NC is a form of programmable automation in which the process is controlled by numbers, letters and symbols. The numbers form a program of instructions designed for a particular workpart or job; when the job changes, the program of instructions changes. This capability to change the program for each new job gives NC its flexibility.

Definition 3.2 (NC according to the Electronic Industries Association)

Numerical Control is a system in which actions are controlled by direct insertion of numerical data at some point. The system must automatically interpret at least some portion of this data.

A further statement in the course material describes a numerically controlled machine as a machine controlled by a set of instructions called the program. In numerical control the numbers form the basic program instructions for different types of jobs, hence the name. When the type of job changes, the program instructions of the job also change. It is easier to write new instructions for each job, hence NC provides a great deal of flexibility. NC technology can be applied to a wide variety of operations such as drafting, assembly, inspection and sheet metal working, but it is used most prominently for metal machining processes such as turning, drilling, milling and shaping. Because all the machining operations can be performed at a fast rate, bulk manufacturing becomes considerably cheaper.

3.1.1 Historical Development

The development of numerical control owes much to the United States Air Force, which recognised the need for more efficient manufacturing methods for modern aircraft. Following the Second World War the components used to fabricate jet aircraft became more complex and required more machining. Most of the machining involved milling operations, so the Air Force sponsored a research project at the Massachusetts Institute of Technology (MIT) to develop a prototype NC milling machine. The prototype was produced by retrofitting a conventional tracer mill with numerical control servomechanisms for the three axes of the machine. In March 1952 the MIT laboratory held the first demonstration of the NC machine, after which machine tool builders gradually began developing their own projects to introduce commercial NC units.

John T. Parsons (1913 to 2007) is widely considered the father of numerical control, which he conceived and implemented with the help of aircraft engineer Frank L. Stulen. Parsons began working in his father's factory as an assembler at the age of 14, holds the first NC patent and was inducted into the National Inventors Hall of Fame. The Society of Manufacturing Engineers described numerical control as marking the beginning of the second industrial revolution, the age in which control of machines and industrial processes passed from imprecise draft to exact science.

Table 3.1: Chronology of numerical control development

Year	**Event**
1952 (May)	Parsons files a patent for "Motor Controlled Apparatus for Positioning Machine Tool"; it is granted in 1958.
1952 (August)	MIT files a patent for "Numerical Control Servo-System".
1952 (March)	First demonstration of the NC machine at the MIT laboratory. The original mill was a retrofitted three axis milling machine using a seven track punched tape system that was complex and expensive (250 vacuum tubes, 175 relays, five refrigerator sized cabinets).
1955	Parsons and the US Air Force define the need for a machine tool able to machine complex, close tolerance aircraft parts with the same quality time after time (repeatability); MIT is the subcontractor. Giddings and Lewis develop the first commercial machine tool of this type around 1955, a five axis machine using a magnetic tape drive (advertised as the Numericord).

Year	**Event**
1959	MIT announces the Automatic Programmed Tools (APT) programming language. The MIT team also holds a press conference to show the new machine; a milled aluminium ashtray was handed out in the press kit.
1960	Direct Numerical Control (DNC): paper tape punch programs are eliminated and programmers can send files directly to machine tools.
1968	Machine tool builders market the first machining center.
1970s	CNC machine tools and Distributed Numerical Control.
1980s	Graphics based CAM systems introduced; Unix and PC based systems become available.
1990s	Price drop in CNC technology.
1997	PC and Windows NT based Open Modular Architecture Control (OMAC) systems introduced to replace firmware controllers.

Key Examination Points

- NC definition (EIA): direct insertion of numerical data, automatically interpreted at least in part.
- Flexibility of NC arises because only the program changes when the job changes.
- Key dates: 1952 first demonstration, 1959 APT, 1960 DNC, 1968 first machining center, 1970s CNC, 1997 OMAC.

3.2 Fundamentals of NC and CNC Systems

3.2.1 Basic Components of an NC System

An operational numerical control system consists of three basic components: (1) the program of instructions, (2) the controller unit, also called the machine control unit (MCU) or machine tool unit, and (3) the machine tool or other controlled process. The program of instructions serves as input to the controller unit, which in turn commands the machine tool or other process to be controlled.

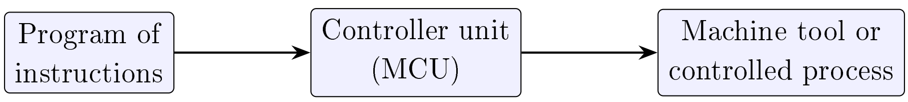


Figure 3.1: Three basic components of an NC system

Program of instructions. This is the detailed step by step set of directions for machine operations. It is coded in numerical or symbolic form on some type of input medium that the controller unit can interpret. The common input media are one inch wide punched tape, punched cards, magnetic tape and 35 mm motion picture film. The common input methods are manual entry of instruction data to the controller unit and a direct link with a computer. Program loading on modern controllers is through the keyboard, punched tape reader, diskette drive, RS 232 serial port or a network interface card.

Controller unit. The NC controller is the brain of the NC system and controls all functions of the machine. Motion control deals with tool position, orientation and speed. Auxiliary control deals with spindle rpm, tool change, fixture clamping and coolant. Its functions are to read and interpret the program of instructions and to convert the instructions into the mechanical actions of the machine tool. The controller consists of electronics and hardware whose typical elements are:

- the tape reader (an electro-mechanical device for winding and reading the punched tape),
- a data buffer,
- signal output channels to the servomotors or other controls of the machine tool,
- feedback channels from the machine tool,
- sequence controls to coordinate the overall operation of the foregoing elements, and
- the control panel or console.

Machine tool. The machine tool carries the spindle and cutting tool and performs the machining according to the signals sent by the MCU.

3.2.2 Elements of a CNC System

A CNC system is made up of six elements: the input device, the central processing unit or machine control unit, the machine tool, the driving system, the feedback devices and the display unit.

Table 3.2: Elements of a CNC system

Element	**Function**
Input device	Enters the part program (a coding system of symbols and numbers). The product is machined according to the program entered. Data may be entered from a computer or from punched tape.
Machine control unit (MCU)	The heart of the CNC system. It checks and decodes the program entered through the input device, runs the machine tool, collects feedback signals to rectify errors and indicates problems that occur in machining.
Machine tool	Carries the spindle which holds and rotates the cutting tool; it works according to the signals sent by the MCU and gives control over speed, depth and angle.

Element	Function
Driving system	Contains the motors that drive the spindle, table movement, saddle movement, angles and coolant supply. When the control unit sends a signal the driving unit starts and supplies the required movement.
Feedback system	Collects information about the driving unit and the cutting tool action, so that errors can be found and displayed on the monitor.
Display unit	Shows information about the complete machining process, displays error information and helps the operator change the program.

The CPU or MCU is the heart of a CNC system. It accepts the information stored in memory as the part program, decodes it and transforms it into specific position control and velocity signals. It also oversees the movement of the control axis or spindle and takes corrective action whenever the movement does not match the programmed values. The MCU:

- reads the coding instructions and sends the proper instruction to every part of the machine,
- recognises interpolations (circular, straight and helical) to form axis transfer commands,
- supports axis movement control and feeds the amplifier circuits that manage the axis drives,
- holds the reviewed signals of location and velocity for each drive axis, and
- performs auxiliary control functions such as coolant and spindle on and off and tool changing.

How a CNC machine works

1. The part program is inserted into the MCU of the CNC.
2. The MCU processes all the data, prepares all motion commands and sends them to the driving system.
3. The drive system acts on the motion commands and controls the motion and velocity of the machine tool.
4. The feedback system records the position and velocity of the machine tool and sends a feedback signal to the MCU.
5. The MCU compares the feedback signals with the reference signals; if there are errors it corrects them and sends a new signal to the machine tool.
6. The display unit is used to see commands, programs and other important data.

3.2.3 Main Functions of a CNC Controller

Axis control function. The number of axes the CNC can control, and the number it can control simultaneously. Axes may be moving axes or rotary axes, basic axes or additional axes. Lathes usually need two simultaneous axes; milling machines, boring machines and machining centers need three or more. Machine tools that process space surfaces need more than three simultaneous axes. The more axes, particularly simultaneous axes, the more complicated the controller and the harder the programming.

Preparation function. Also called the "G" function. It commands the movement mode of the machine: basic movement, plane selection, coordinate setting, tool interpolation, fixed cycles and metric conversion. It is written as G followed by two digits.

Interpolation function. The CNC realises the tool trajectory through software interpolation. Because contour control is strongly real time, interpolation is divided into *coarse interpolation* (software interpolates a small line segment each time) and *fine interpolation* (the servo interface divides the small segment into single pulse outputs). Contours are made of straight lines, arcs and more complicated curves, so the controller provides straight line, arc, parabolic, sine, cylindrical and spline interpolation. Methods to implement interpolation are the point by point comparison method, the numerical integration method and the direct function algorithm.

Feed function. Commanded with the F word; it directly sets the feed rate of each axis.

3.2.4 Types of NC Systems: Open Loop and Closed Loop

Definition 3.3 (Open loop control)

In an open loop control system stepping (step) motors are used. They are driven by electric pulses and every pulse rotates the motor shaft through a fixed amount. By counting the pulses the amount of motion is controlled. There is no feedback signal to check how close the actual machine movement comes to the programmed movement, so there is no error correction and positioning accuracy is lower.

Definition 3.4 (Closed loop control)

In a closed loop control system DC, AC or hydraulic servomotors are used. Their speed is variable and is controlled by the amount of current or fluid. The motors are connected to the spindle and the table. A position transducer continuously monitors the movement and sends a signal back to a comparator, which makes adjustments. Better accuracy is achieved, the system is more expensive and is suitable for large machine tools.

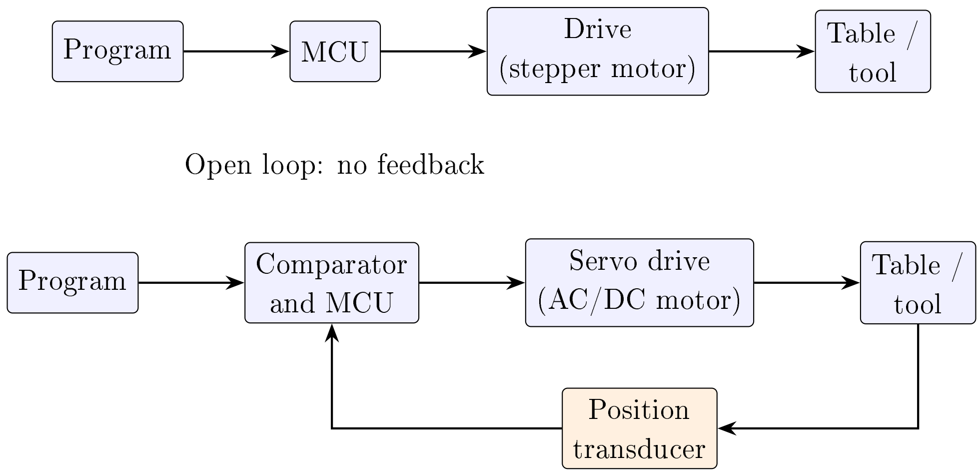


Figure 3.2: Open loop (above) and closed loop (below) control of tool movement

Table 3.3: Comparison of open loop and closed loop control

Feature	**Open loop**	**Closed loop**
Motor	Stepping motor driven by pulses	DC, AC or hydraulic servo-motor
Feedback signal	None	Position (and speed) transducers give feedback
Error correction	None	Comparator corrects the error
Positioning accuracy	Lower	Better
Cost	Lower	More expensive
Typical size of machine	Small, less demanding shop machines	Large machine tools

3.2.5 Drive System

A drive system consists of amplifier circuits, stepping motors or servomotors and ball lead screws. The MCU feeds control signals (position and speed) for each axis to the amplifier circuits. The signals are amplified to actuate the stepping motors or servomotors, which rotate the ball lead screws to position the machine table.

Stepping motors. A stepping motor provides open loop digital control of the position of a workpiece. The drive unit receives a direction input (clockwise or counter clockwise) and pulse inputs. For each pulse it manipulates the motor voltage and current so that the shaft rotates by a fixed angle (one step). The lead screw converts the rotary motion into linear motion of the workpiece. In machine tool applications there are often 200 steps in a full rotation (100 steps for 180 degrees). Microsteps, in which each step is divided into smaller steps, give higher resolution. Stepper motors operate open loop, so position depends only on the pulses received.

Servo motors. Servo motors use closed loop circuitry. A regular DC or AC motor is connected to an encoder with a sensor, so accuracy and resolution are high. The servo amplifier powers the motor and counts the steps. The high torque to inertia ratio permits rapid load acceleration and efficiency may reach 90 percent with light loads. Servo motors use fewer electromagnetic poles (usually fewer than a dozen), receive an exact position signal when powered on, and need no re-establishment of zero. Rotary or angle encoders are the primary feedback devices.

Table 3.4: Comparison of stepper and servo motors

Feature	**Stepper motor**	**Servo motor**
Control	Open loop; position dictated by the pulses received	Closed loop with encoder feedback
Cost and operation	Simple to operate, inexpensive	More costly, more complicated to operate
Torque and speed	Torque falls with rapid acceleration and at high speed; low torque to inertia ratio	Keeps nearly all torque at maximum speed; high torque to inertia ratio
Efficiency	Typically lower; resonance prone, smooth movement often needs microstepping; noisy and overheats at high performance	Up to 90 percent with lighter loads; brushes need servicing after about 2000 hours; more susceptible to overheating and overloading
Typical use	Less demanding workhorse machines and hobbyists	Advanced industrial machines needing high precision, speed and torque

Recirculating ball screws. They transform the rotational motion of the motor into translational motion of the nut attached to the machine table. Accuracy of CNC machines depends on rigid construction, care in manufacture and the use of ball screws, which almost eliminate slop in the screws used to move portions of the machine.

Positioning resolution. The positioning resolution of a ball screw drive is directly proportional to the smallest angle the motor can turn, which is controlled by the motor step size. Microsteps decrease the step size. CNC machines typically have resolutions of 0.0025 mm or better.

3.2.6 Feedback System

The feedback system, also called the measuring system, uses position and speed transducers to monitor continuously the position of the cutting tool. The MCU uses the difference between reference signals and feedback signals to generate control signals that correct position and speed errors.

Table 3.5: Feedback devices of CNC machines

Device	**Description and use**
Potentiometer	A three terminal variable resistor that converts linear or rotary motion into a proportional voltage. A resistive track and a movable wiper: as the shaft or slider moves the wiper changes position, changing the output voltage in proportion to displacement. Used for axis position feedback in older CNC systems (modern CNCs generally use encoders or linear scales), and as a rotary knob for feed rate override (for example 0 to 150 percent) and spindle speed override. Types: rotary (most common), linear (older position sensing), single turn (typically 270 to 300 degrees, control knobs), multi turn (fine adjustment, calibration).
Encoder	An electromechanical sensor that converts rotary or linear position into an electrical output signal. It provides position, speed and direction feedback to the CNC controller for precise control of machine axes and spindle.
Resolver	A rotary transformer that produces an output signal that is a function of the rotor position. It is used with servomotors, and positioning feedback on machining centers is provided by resolvers.
Tachometer	Its electrical output is proportional to the rate of angular rotation (velocity feedback).

For velocity feedback, encoders, resolvers and potentiometers give a number of pulses per unit time that is proportional to the rate of change of position.

3.2.7 NC, CNC and DNC

Definition 3.5 (Computer numerical control)

CNC is an NC system that utilises a dedicated, stored program computer to perform some or all of the basic numerical control functions. Numerical control (also computer numerical control) is the automated control of machining tools such as drills, lathes, mills and 3D printers by means of a computer.

Definition 3.6 (Direct numerical control)

DNC is a manufacturing system in which a number of machines are controlled by a computer through direct connection and in real time. The tape reader is omitted, relieving the system of its least reliable component; the part program is transmitted to the machine tool directly from computer memory. In principle one computer can control more than 100 separate machines.

DNC arose because conventional NC suffers from part programming mistakes, punched tape, the tape reader, the controller and lack of management information. A DNC system has four elements: a central computer, bulk memory holding NC part programs, telecommunication lines and machine tools. The computer calls part program instructions from bulk storage and sends data back from the machines. This two way information flow occurs in real time, so a machine's request for instructions must be satisfied almost instantaneously. Depending on the number of machines and the computational load, satellite computers may be used. In distributed numerical control the central computer downloads complete programs to the CNC machines (workstations or PCs); speed increases, large files can be handled and the number of machine tools is expanded.

Table 3.6: Main differences between DNC and CNC

Aspect	**DNC**	**CNC**
Number of machines	Distributes instruction data to, and collects data from, a large number of machines	Controls only one machine or a small number of machines
Location	Typically remote from the machines it controls	Located very near its machine tool
Software purpose	Controls individual production technology and also serves as part of the management information system of the manufacturing sector of the firm	Developed to augment the capabilities of a particular machine tool

Additional CNC advantages noted in the material

Increased program storage at the machine tool, program editing at the machine tool, control system upgrades, an option for a resident CAM system at the machine tool, and tool path verification.

3.2.8 CNC and PLC

A programmable logic controller (PLC) is a general purpose logic controller designed to let users create their own applications by selecting appropriate inputs and outputs and programming process and sequence control functions. A CNC controller is a particular application, usually to control a multi axis machine tool. The forerunner of CNC was NC, which used solid state logic, relays and timers. In a CNC machine the PLC (often called the programmable machine controller, PMC) controls the auxiliary functions: spindle forward and reverse and on and off, tool changer, clamping and unclamping, cooling, chip removal and safety interlocks. The NC realises the geometric motion of the tool relative to the workpiece; the PLC accomplishes machine auxiliary equipment control.

A CNC has three parts: the man machine interface (MMI), the numerical control kernel (NCK), which parses G code and runs the interpolation algorithm, and the PLC.

PLC applications include safety interlocks, emergency stop, axis limit signals and feed drive alarms. PLC program design proceeds process by process, analysing the starting conditions and constraints of each process, which makes the program modular and easy to debug independently.

3.2.9 Extensions of NC and Adaptive Control

The success of NC led to four extensions: direct numerical control, computer numerical control, adaptive control and industrial robots. NC reduces the nonproductive time in manufacturing. In machining, *adaptive control* means a control system that measures certain output process variables and uses them to control speed and feed. The variables tried include spindle deflection or force, torque, cutting temperature variation, amplitude of vibration and horsepower; in principle every metal cutting variable has been tried in experimental adaptive control systems.

3.2.10 Suitability, Applications, Advantages and Disadvantages of NC

NC is suited to parts that are processed frequently and in small lot sizes, have complex geometry, need many operations, require much metal removal, are subject to engineering design changes, must hold close tolerances, are expensive so that processing mistakes are costly, or require 100 percent inspection. Potential applications beyond machining include pressworking machine tools, welding, inspection machines, automatic drafting, assembly machines, tube bending, flame cutting, plasma cutting, laser beam processes, cloth cutting, automatic riveting, wire wrap machines and automated knitting machines.

Table 3.7: Advantages and disadvantages of NC

Advantages	**Disadvantages**
Reduced nonproductive time; reduced fixtures; reduced manufacturing lead time; greater manufacturing flexibility; improved quality control; reduced inventory; reduced floor space requirement	Higher investment cost; higher maintenance cost; finding and training NC personnel

Key Examination Points

- Three components of NC: program of instructions, controller unit, machine tool.
- Open loop uses stepping motors and no feedback; closed loop uses servomotors with transducers and a comparator.
- CNC uses a dedicated computer at the machine; DNC uses a central computer serving many machines in real time.

- Feedback devices: potentiometer, encoder, resolver, tachometer.
- CNC controller functions: axis control, preparation (G) function, interpolation (coarse and fine), feed.

3.3 Machine Tools, Axes and Coordinate Systems

3.3.1 Machining Centers, Turning Centers and Multitasking Machines

A *machining center* is a machine tool capable of multiple operations and processes in a single set up utilising multiple axes, and it typically has an automatic mechanism to change tools. Machine motion is programmable, servo motors drive the feed mechanisms for the tool axes and positioning feedback is provided to the control system. Machining centers with automatic tool changers can change 90 or more tools and can perform milling, drilling, tapping and boring on many faces. *Turning centers* execute many different types of lathe cutting operations, possibly simultaneously, on a rotating part. An example is a turning center able to perform OD turning, external threading, cross hole drilling, engraving and milling in one set up; such a machine may have multiple spindles.

Multitasking machines combine several cutting processes (turning, milling, drilling, tapping, deep hole boring) on one machine instead of handling them on multiple machines. In their most common form they combine one or two lathes with a machining center; more recently grinding, gear cutting and laser hybrid capabilities have been integrated. A mill/turn machine combines milling (tool rotating) and turning (workpiece rotating) in one machine, completing complex operations faster and with greater accuracy. A typical configuration has a turning center with two spindles and two turrets and eight controlled axes. Typical parts are drive shafts, pump covers and aircraft housings.

3.3.2 Standard Axes

Machine axes follow the industry standard EIA RS 267A. The axes form a right handed coordinate system.

- The **Z axis** is aligned with the spindle. +Z moves the tool away from the workpiece; a move toward the work is in the $-Z$ direction. The Z axis is parallel to the spindle for a rotating workpiece (lathe) and parallel to the machine tool axis for a rotating tool (milling, drilling, boring machine). Spindle motion is assigned the Z axis.
- The **X axis**: on a lathe it is perpendicular to the spindle and in the direction of tool movement; on a horizontal machine it is parallel to the table; on a vertical machine +X points to the right when facing the machine (the lecture material also notes that X moves from right to left as one faces the machine and that Y moves toward and away from the operator).
- The **Y axis** completes the right handed system.

- Additional linear slide axes are labelled U, V, W; rotary motions around axes parallel to X, Y and Z are labelled A, B, C.

3.3.3 Classification by Number of Axes

Table 3.8: Machine tool axis configurations

Type	**Description**
2 axis	The tool can be controlled simultaneously along two axes and guides the cutting tool along a 2D contour; independent movement is specified along the third axis, whose control plane is parallel to the XY plane.
2.5 axis	The tool is controlled to follow an inclined Z axis control plane.
3 axis	The tool is controlled along X, Y and Z simultaneously, but the tool orientation does not change with tool motion.
Multi axis (4, 5, 6)	The tool axis orientation varies with tool motion in three dimensional space (for example five axis milling machines).

3.4 Positioning Methods and Motion Control

3.4.1 Absolute and Incremental Programming

Definition 3.7 (Absolute positioning)

In the absolute positioning system all coordinates are measured from a fixed point of origin (the zero point). Selected with G90.

Definition 3.8 (Incremental positioning)

In the incremental positioning system each set of coordinates has its point of origin at the last point established, so each coordinate is measured from the previous position. Selected with G91.

3.4.2 Point to Point and Continuous Path Motion

The lecture material identifies three categories of motion system: point to point (no contouring capability), straight cut control (one axis motion at a time is controlled for machining) and contouring (multiple axes controlled simultaneously).

Table 3.9: Point to point versus continuous path control

Feature	Point to point	Continuous path (contouring)
Action	Tool moves to a point on the part and performs an operation at that point only	Tool maintains continuous contact with the part as it cuts a contour
Cutting during motion	Tool is not in continuous contact while moving	Operation is performed during movement
Typical operations	Drilling, reaming, punching, boring, tapping (also robotics)	Milling along lines at any angle, milling arcs, lathe turning (profile milling)

Key Examination Points

- Z axis lies along the spindle; +Z is away from the workpiece.
- G90 absolute, G91 incremental.
- Point to point: drilling, boring, tapping; continuous path: milling contours, turning.
- Machine classes: 2, 2.5, 3 and multi axis (4, 5, 6).

3.5 Part Programming: Manual and Computer Assisted

Definition 3.9 (Part programming)

Part programming is the planning and documenting of the sequence of processing steps to be performed on the NC machine.

Table 3.10: Methods of part programming

Method	Description
Manual part programming	The programmer writes the machining instructions on a special form called the part programmer manuscript. It is a tedious task subject to error.
Computer assisted part programming	A high speed digital computer assists in the part programming process, saving time and giving a more accurate and efficient part program.

3.5.1 The NC Procedure

1. Process planning.

2. Preparation of a route sheet (a listing of the sequence of operations to be performed on the workparts).
3. Part programming and verification: manual part programming or computer assisted part programming.
4. Tape preparation.
5. Tape verification.
6. Input of the part program to the NC controller.
7. Production.

3.5.2 Part Program Content

A part program specifies which tool should be loaded on the machine spindle, the cutting conditions (speed, feed, coolant on or off), the start point and end point of each motion segment, and how to move the tool with respect to the machine. Part programming contains geometric information about the part and motion information to move the cutting tool with respect to the workpiece. The first thing to define is the coordinate system; then the syntax of part programming can follow.

The basic elements of an NC program are:

(a) Preparatory functions: which unit, which interpolator, absolute or incremental programming, which circular interpolation plane, cutter compensation and so on.

(b) Coordinates: three translational and three rotational axes.

(c) Machining parameters: feed and speed.

(d) Tool control: tool diameter, next tool number, tool change.

(e) Cycle functions: drill cycle, ream cycle, bore cycle, mill cycle, clearance plane.

(f) Coolant control: coolant on or off, flood, mist.

(g) Miscellaneous control: spindle on or off, tape rewind, spindle rotation direction, pallet change, clamp control and so on.

(h) Interpolators: linear and circular interpolation.

3.6 Part Program Structure, Blocks and Word Address Format

The machine receives instructions as a sequence of blocks containing commands to set machine parameters such as speed, feed and other relevant information. A *block* is equivalent to a line of code in a part program. Each command has an identifying letter followed by an associated number. Formats and syntax are well defined but vary with differences between machines.

Identifying letters used in commands: sequence number (N code), preparatory command (G code), dimension words (X, Y, Z, A and B words), feed command (F code), speed command (S code), tool selection (T code) and miscellaneous (M code).

Formats for arranging commands into a block are: fixed sequential format, block address format, tab sequential format and word address format. In the word address format each word starts with a letter. For example:

```
N040 G00 X0 Y0 Z300 T01 M06
```

Here N is the block (identifier) number, G the preparatory command, X, Y and Z the coordinates along the axes, T the tool number and M the miscellaneous command.

A second example from the material is `N135 G01 X1.0 Y1.0 Z0.125 T01 F5.0`, which contains a block number (N135), a G code (G01), coordinates (X, Y, Z), a tool number (T01) and a feed (F5.0).

3.6.1 Programming Procedure

1. Analyse the drawing and identify the datum.
2. Select the coordinate system and the work coordinate system.
3. Select the tools and the machining sequence.
4. Determine the machining parameters (spindle speed, feed rate, depth of cut).
5. Write the program in blocks on the part programmer manuscript.
6. Load the program (keyboard, tape reader, diskette drive, RS 232 serial port or network card).
7. Verify the tool path (simulation).
8. Prove out on the machine and edit if necessary, then produce the part.

3.7 Preparatory Functions (G Codes)

The tables below list the G codes given in the course material. Codes are grouped as in the source.

Table 3.11: G codes (preparatory functions)

Code	**Function**	**Code**	**Function**
G00	Positioning (rapid traverse)	G41	Cutter compensation left
G01	Linear interpolation (feed)	G42	Cutter compensation right
G02	Circular interpolation CW	G43	Tool length compensation, + direction
G03	Circular interpolation CCW	G44	Tool length compensation, − direction
G04	Dwell	G45	Tool offset increase

Code	Function	Code	Function
G07	Imaginary axis designation	G46	Tool offset decrease
G09	Exact stop check	G47	Tool offset double increase
G10	Offset value setting	G48	Tool offset double decrease
G17	XY plane selection	G49	Tool length compensation cancel
G18	ZX plane selection	G50	Scaling off
G19	YZ plane selection	G51	Scaling on
G20	Input in inch	G52	Local coordinate system setting
G21	Input in mm	G54	Work coordinate system 1 select
G22	Stored stroke limit on	G55	Work coordinate system 2 select
G23	Stored stroke limit off	G56	Work coordinate system 3 select
G27	Reference point return check	G57	Work coordinate system 4 select
G28	Return to reference point	G58	Work coordinate system 5 select
G29	Return from reference point	G59	Work coordinate system 6 select
G30	Return to 2nd, 3rd and 4th reference point	G60	Single direction positioning
G31	Skip cutting	G61	Exact stop check mode
G33	Thread cutting	G64	Cutting mode
G40	Cutter compensation cancel	G65	Custom macro simple call
G66	Custom macro modal call	G84	Tapping cycle
G67	Custom macro modal call cancel	G86	Boring cycle
G68	Coordinate system rotation on	G87	Back boring cycle
G69	Coordinate system rotation off	G89	Boring cycle
G73	Peck drilling cycle	G90	Absolute programming
G74	Counter tapping cycle	G91	Incremental programming
G76	Fine boring	G92	Programming of absolute zero point
G80	Canned cycle cancel	G94	Per minute feed
G81	Drilling cycle, spot boring	G95	Per revolution feed
G82	Drilling cycle, counter boring	G96	Constant surface speed control
G83	Peck drilling cycle	G97	Constant surface speed control cancel

Code	Function	Code	Function
		G98	Return to initial point in canned cycle
		G99	Return to R point in canned cycle

3.8 Miscellaneous Functions (M Codes)

Table 3.12: M codes (miscellaneous functions)

Code	Function	Code	Function
M00	Program stop	M25	Tool clamp
M01	Optional stop	M26	Tool unclamp
M02	End of program (no rewind)	M27	Clutch neutral on
M03	Spindle CW	M28	Clutch neutral off
M04	Spindle CCW	M30	End of program (rewind stop)
M05	Spindle stop	M98	Call subprogram
M06	Tool change	M99	End subprogram
M08	Flood coolant on	M21 to M24	Tool related functions
M09	Flood coolant off		
M19	Spindle orientation on		

3.9 Spindle, Feed, Coolant and Tool Commands

F code (feed). Feed speed in inch per minute (ipm) or inch per revolution (ipr). An F value must be given before G01, G02 or G03 can be used. Example: `N0100 G02 X7.000 Y2.000 I5.000 J2.000 F6.00`.

S code (spindle speed). Cutting speed programmed in rpm. The S code does not turn the spindle on; the spindle is turned on by an M code (M03 clockwise, M04 counter clockwise, M05 stop). Example: `N0010 S1000`.

T code (tool). Tool selection, with M06 for tool change (for example `M06 T1`).

Coolant. M08 flood coolant on, M09 flood coolant off (APT: `COOLNT/MIST`, `COOLNT/FLOOD`, `COOLNT/OFF`).

G94, G95. Per minute feed and per revolution feed. G96 and G97 turn constant surface speed control on and cancel it.

3.10 Tool Movements and Interpolation

3.10.1 Rapid Positioning and Linear Interpolation

G00 is rapid traverse positioning. G01 is a preparatory function that moves the tool to a specified location along a straight line. It is called *linear interpolation* and is typically used for straight features such as turning a cylindrical surface or cutting a slot in milling.

```
Format:  N_ G01 X_ Y_ Z_ F_
```

Linear interpolation results in a straight line feed move. Unless tool compensation is used, the coordinates refer to the centerline of the tool. If the same maximum speed applies to the X and Y axes, the initial motion in the XY plane is at 45° to the axes until motion in one axis is complete, and the balance of the motion occurs in the other axis; this is the point to point type of motion.

Example from the material (lathe, XZ plane):

```
N10 G00 X1 Z1
N15 Z0.1
N20 G01 Z-0.125 F5
N25 X2 Z2 F10
```

Block N10 rapids to X1 Z1; N15 rapids to Z0.1 (modal G00 remains active); N20 feeds to Z$-$0.125 at F5; N25 feeds to X2 Z2 at F10.

3.10.2 Circular Interpolation

G02 moves the tool along a circular path in a clockwise direction and G03 in a counter clockwise direction. To describe the path to the MCU, the end point of the arc and the location of the arc center must be specified. Within the block the arc center is given by its location *relative to the start of the arc*, using I, J and K (parallel to X, Y and Z). The G02 command may instead use the end point and a radius R.

```
N_ G02 X2 Y1 I0 J-1 F10
or
N_ G02 X2 Y1 R1
```

3.10.3 Interpolation Methods

Table 3.13: Interpolation methods

Method	Description
Linear	Straight line between two points in space.
Circular	Circular arc defined by starting point, end point, center or radius, and direction.

Method	Description
Helical	Circular plus linear motion (three dimensional circular interpolation producing a helical shape).
Parabolic and cubic	Free form curves using higher order equations.

Advanced controller features listed in the material include execution of part of a program in a rotated or mirrored position (G68, G69), the ability to scale a program (G50, G51), helical interpolation, and parabolic and cubic interpolation.

Circular interpolation in NC is an approximation of a curved path by a series of straight line segments, where the tolerance is defined only on the outside of the nominal curve.

3.11 Tool Compensation

Tool compensation lets the programmer program the part geometry rather than the tool center path. G41 and G42 apply cutter (radius) compensation to the left and right of the programmed path and G40 cancels it. G43 and G44 apply tool length compensation in the positive and negative directions using an offset number (for example `G43 H01`) and G49 cancels it. G45 to G48 increase or decrease tool offset (single or double). G10 sets offset values. Work coordinate systems G54 to G59 select fixture offsets, G52 sets a local coordinate system and G92 programs the absolute zero point.

3.12 Canned Cycles

The sequence of some machining operations is the same for any part and any machine. For example, drilling a hole involves: (1) position the tool above the point where the hole will be drilled; (2) set the correct spindle speed; (3) feed the tool into the workpiece at a controlled feed rate to a predetermined depth; (4) retract the tool at a rapid rate to just above the point where the hole started. A canned cycle packages such a sequence in a single block.

Table 3.14: Commonly used canned cycles

Code	Function	Down feed	At bottom	Retraction
G81	Drilling	Continuous feed	No action	Rapid
G82	Spot face, counter bore	Continuous feed	Dwell	Rapid
G83	Deep hole drilling	Peck	No action	Rapid
G84	Tapping	Continuous feed	Reverse spindle	Feed rate
G85	Through boring (in and out)	Continuous feed	No action	Feed rate

Code	Function	Down feed	At bottom	Retraction
G86	Through boring (in only)	Continuous feed	Stop spindle	Rapid

G80 cancels a canned cycle. G98 returns to the initial point and G99 returns to the R point after the cycle. Other cycles in the G list: G73 peck drilling, G74 counter tapping, G76 fine boring, G87 back boring, G89 boring.

3.13 Subprograms and Repetitive Programming

Repeated machining sequences can be written once as a subprogram. M98 calls a subprogram and M99 ends it and returns to the main program. Custom macros are called with G65 (simple call) or G66 (modal call), and G67 cancels the modal call. In the APT language the equivalent is the macro facility, which uses variable arguments like a FORTRAN subroutine (see the APT section later in this chapter).

Key Examination Points

- Word address format: N (sequence), G (preparatory), X Y Z (dimensions), F (feed), S (speed), T (tool), M (miscellaneous).
- G02/G03 use I, J, K relative to the arc start point, or R radius.
- S code does not start the spindle; M03/M04 do.
- Canned cycles G81 to G89 package positioning, feed, bottom action and retraction.
- M98/M99 for subprograms; G65/G66/G67 for macros.

3.14 Worked Manual Programming Examples

Example 3.1 (Drilling one hole on a vertical milling machine)

The steps needed to drill a hole are: (1) rapid X and Y to the hole position; (2) rapid Z just above the hole (0.100 in); (3) turn on coolant; (4) turn on spindle; (5) feed Z to drill the hole; (6) rapid Z out of the hole; (7) turn off spindle; (8) turn off coolant; (9) rapid X and Y home. The program is:

```
O0001
N005 G54 G90 S600 M03
N010 G00 X1.0 Y1.0
N015 G43 H01 Z.1 M08
N020 G01 Z-.75 F3.5
N025 G00 Z.1 M09
N030 G91 G28 X0 Y0 Z0
```

```
N035 M30
```

Block	Explanation
O0001	Number assigned to this program.
N005 G54 G90 S600 M03	N005 sequence number; G54 fixture offset; G90 absolute programming mode; S600 spindle speed 600 rpm; M03 spindle on clockwise.
N010 G00 X1.0 Y1.0	G00 rapid motion; X1.0 and Y1.0 are coordinates 1.0 in from zero.
N015 G43 H01 Z.1 M08	G43 tool length compensation; H01 specifies the tool length offset; Z.1 is 0.1 in from zero; M08 flood coolant on.
N020 G01 Z-.75 F3.5	G01 straight line cutting motion; Z−.75 is 0.75 in below zero; feed rate 3.5 in/min.
N025 G00 Z.1 M09	Rapid motion back to Z.1; M09 coolant off.
N030 G91 G28 X0 Y0 Z0	G91 incremental mode; G28 zero (reference point) return; X0 Y0 Z0 at zero.
N035 M30	End of program.

Example 3.2 (Profile and pocket milling of a plate)

Tool 1 is a half inch diameter end mill. The program is described in four passes.

First pass: conventional mill to a depth of 0.125 around the edge profile.

```
% :1002
N5 G90 G20
N10 M06 T1
N15 M03 S1200
N20 G00 X0.125 Y0.125
N30 Z0.125
N35 G01 Z-0.125 F5
N40 X3.875
N45 Y4.125
N50 X0.125
N55 Y0.125
```

Second pass: conventional mill to a depth of 0.25 around the edge profile.

```
N35 Z-0.250
N40 X3.875
N45 Y4.125
N50 X0.125
N55 Y0.125
N60 Z0.125
```

Third pass: conventional mill to a depth of 0.125 around the pocket profile.

```
N65 G00 X1.25 Y1.0
N70 G01 Z-0.125 F5
N75 X1.75
N80 Y2.5
N85 X1.25
N90 Y1.0
N95 Z0.125
```

Fourth pass: climb mill to a depth of 0.125 across the remaining material.

```
N100 Y2.125
N105 X2.625
N110 Z0.125
N115 G00 X-5 Y-5 Z5
N120 M05
N125 M30
```

Explanation: N5 selects absolute programming and inch units; N10 calls tool 1; N15 starts the spindle at 1200 rpm; N20 rapids to the start corner; N30 lowers Z to 0.125; N35 plunges to -0.125 at F5 and N40 to N55 cut the four sides of the rectangle. In the second pass the plunge in N35 goes to -0.250 and the same rectangle is cut again, then N60 retracts to Z0.125. The third pass moves rapidly to the pocket corner, plunges and cuts around the pocket rectangle. In the fourth pass blocks N100 to N105 clear the remaining material in a climb milling pass, N110 retracts, N115 rapids clear of the part, N120 stops the spindle (M05) and N125 ends the program (M30).

Example 3.3 (Manuscript program using EOB block ends)

The material gives the following incremental program (G91, G71 metric) with end of block (EOB) characters:

```
N001 G91 EOB
N002 G71 EOB
N003 G00 X0.0 Y0.0 Z40.0 T0.1 M06 EOB
N004 G01 X65.0 Y0.0 Z-40.0 F950 S717 M03 EOB
N005 G01 X10.0 F350 M08 EOB
N006 G01 X110.0 EOB
N007 G01 Y70.0 EOB
N008 G01 X-40.86 EOB
N009 G02 X-28.28 Y0.0 I14.14 J5.0 EOB
N010 G01 X-40.86 EOB
N011 G01 Y-70.0 EOB
N012 G01 X-75.0 Y0.0 Z40.0 F950 M30
```

Explanation: N001 selects incremental mode (G91) and N002 metric units (G71). N003 lifts Z by 40 and calls tool change with M06. N004 moves to the start of the cut at F950 with spindle at 717 rpm clockwise. N005 sets F350 and turns coolant on (M08). N006 to N008 and N010 to N011 are straight moves; N009 is a clockwise arc (G02) with center offsets I14.14 and J5.0 relative to the arc start. N012 returns and ends the program with M30.

Key Examination Points

- Program header: program number (O0001), then mode and offset (G54 G90), speed (S) and spindle start (M03).
- G43 Hnn applies tool length offset; G28 returns to reference point; M30 ends the program.
- Conventional versus climb milling passes appear in the milling example (passes one to three conventional, pass four climb).

3.15 Computer Assisted Part Programming

In computer assisted part programming software programs automatically generate the CNC data. Very complex part shapes lead to very large NC programs; the NC controller memory may not be able to hold such a huge part program. In that case the computer (DNC) feeds a few blocks of the NC program to the controller, and when almost all blocks have been executed the controller requests more blocks.

3.15.1 Flow of Computer Aided CNC Processing

1. Develop the 3D geometric model of the part using CAD.
2. Decide which machining operations and cutter path directions are required (computer assisted).
3. Choose the tooling required (computer assisted).
4. Run CAM software to generate the CNC part program (make 3D model, define tool, generate CNC data, simulate cutting).
5. Verify and edit the program.
6. Download the part program to the appropriate machine.
7. Verify the program on the actual machine and edit if necessary.
8. Run the program and produce the part.

Table 3.16: Division of work in APT type computer assisted programming

Agent	**Tasks**
Part programmer	Defines the workpart geometry; specifies the operation sequence and tool path; writes the English like statements of the APT part program.
Computer	Input translation; arithmetic calculation; cutter offset computation; post processor.

3.15.2 Comparison of Manual and Computer Assisted Programming

Table 3.17: Manual versus computer assisted part programming

Aspect	**Manual**	**Computer assisted**
Medium	Part programmer manuscript written by hand	Computer generates CL data and NC program
Errors	Tedious and error prone	More accurate and more efficient
Time	Longer	Saves time
Geometry work	Programmer computes coordinates	Computer performs arithmetic, offsets and translation
Suitable for	Simple point to point work and short programs	Complex parts, three dimensional and multi axis work

3.16 APT and Other Part Programming Languages

Definition 3.10 (Automatically Programmed Tools (APT))

APT is a text based part programming system in which the programmer defines a series of lines, arcs and points that describe the overall part geometry. These features are used to generate a cutter location (CL) file. APT was developed as a joint effort between the aerospace industry, MIT and the US Air Force. It is still used today and accounts for about 5 to 10 percent of all programming in the defense and aerospace industries.

3.16.1 APT Statement Types

APT statements belong to one of five types: identification statements, geometry statements, motion statements, post processor statements and auxiliary statements.

3.16.2 Geometry Statements

The general form is `Symbol = geometry_word/descriptive data`.

Table 3.18: APT geometry definitions given in the course material

Entity	**Definitions**
Points	`P1 = POINT/X, Y, Z`; `P2 = POINT/L1, L2`; `P3 = POINT/CENTER, C1`; `P4 = POINT/YLARGE, INTOF, L1, C1`; `P5 = POINT/XLARGE, INTOF, L1, C1`; `P6 = POINT/XLARGE, INTOF, C1, C1`; `P7 = POINT/YLARGE, INTOF, C1, C1`
Lines	`L1 = LINE/X1, Y1, Z1, X2, Y2, Z2`; `L2 = LINE/P1, P2`; `L3 = LINE/P1, PARLEL, P2`; `L4 = LINE/P1, PERPTO, L0`; `L5 = LINE/P1, LEFT, TANTO, C1`; `L6 = LINE/P1, RIGHT, TANTO, C1`; `L7 = LINE/LEFT, TANTO, C1, LEFT, TANTO, C2`; `L8 = LINE/LEFT, TANTO, C1, RIGHT, TANTO, C2`; `L9 = LINE/RIGHT, TANTO, C1, LEFT, TANTO, C2`; `L10 = LINE/RIGHT, TANTO, C1, RIGHT, TANTO, C2`; `L11 = LINE/P1, ATANGL, L0`
Circles	`C1 = CIRCLE/X, Y, Z, R`; `C2 = CIRCLE/CENTER, P1, RADIOUS, R`; `C3 = CIRCLE/CENTER, P1, TANTO, L0`; `C4 = CIRCLE/P1, P2, P3`; `C5 = CIRCLE/XSMALL, L1, XSMALL, L2, RADIOUS, R` (and the same with XLARGE, YLARGE or YSMALL)
Planes	`PL1 = PLANE/P1, P2, P3`; `PL2 = PLANE/PARLEL, PL0, XLARGE, D` (and the same with XLARGE, YLARGE, YSMALL, ZLARGE or ZSMALL)

3.16.3 Motion Statements

For point to point operation there are three motion statements for positioning the tool at a desired point: `FROM/point_location`, `GOTO/point_location` and `GODLTA/dx, dy, dz`.

Table 3.19: APT motion statement syntax

Statement		**Meaning**
`FROM/X,Y,Z` `FROM/X,Y,Z,I,J,K`	or	Starting point; X, Y, Z are the coordinates of the tool position and I, J, K the components of the tool axis.
`GOTO/X,Y,Z` `GOTO/X,Y,Z,I,J,K`	or	Linear tool motion to the given position.
`GODLTA/X,Y,Z` `GODLTA/X,Y,Z,I,J,K`	or	Incremental motion by the given amounts.

Table 3.19: APT motion statement syntax

Statement	Meaning
`GO/{TO}, Drive surface, {TO} Part surface, {TO}, Check surface`	Contouring motion; the same applies with PAST or ON instead of TO; TANTO may replace the final TO. Example: `GO/TO, L1, TO, PS, TANTO, C1` and `GO/PAST, L1, TO, PS, TANTO, C1`.
`GOLFT/, GORGT/, GOUP/, GODOWN/, GOFWD/, GOBACK/`	Directional continuous path motion commands (left, right, up, down, forward, back).

3.16.4 Additional Statements

`MACHIN/DRILL, 2` selects the machine and post processor, `COOLNT/` (MIST, FLOOD, OFF), `FEDRAT/`, `SPINDL/` (for example `SPINDL/ON`, `SPINDL/1250, CCLW`), `TOOLNO/`, `TURRET/` and `END`.

3.16.5 APT Examples

Example 3.4 (Point to point drilling)

Points P0, P1, P2 are defined and two holes are drilled:

```
P0 = POINT/0.0, 3.0, 0.1
P1 = POINT/1.0, 1.0, 0.1
P2 = POINT/2.0, 1.0, 0.1
FROM/P0
GOTO/P1
GODLTA/0, 0, -0.7
GODLTA/0, 0, 0.7
GOTO/P2
GODLTA/0, 0, -0.7
GODLTA/0, 0, 0.7
GOTO/P0
```

The tool starts at P0, moves to P1, drills down 0.7 with GODLTA and retracts by 0.7, then repeats at P2 and returns to P0.

Example 3.5 (Contour motion with GO, GORGT and GOLFT)

Two contour programs from the material follow. In the first, the tool starts at SP and traverses a closed contour of lines L1 to L4 and circle C1:

```
FROM/SP
GO/TO, L1, TO, PS, ON, L4
GORGT/L1, PAST, L2
```

```
GOLFT/L2, PAST, L3
GOLFT/L3, PAST, C1
GOLFT/C1, PAST, L3
GOLFT/L3, PAST, L4
GOLFT/L4, PAST, L1
GOTO/SP
```

In the second program a tangential transition onto the circle is used:

```
FROM/SP
GOTO, L1, TO, PS, ON, L6
GORGT/L1, PAST, L2
GORGT/L2, TANTO, C1
GOFWD/C1, TANTO, L3
GOFWD/L3, PAST, L4
GOLFT/L4, PAST, L5
GOLFT/L5, PAST, L6
GOLFT/L6, PAST, L1
GOTO/SP
```

Example 3.6 (APT macro facility)

The macro facility uses variable arguments as in a FORTRAN subroutine. The macro DRILL is called twice:

```
P0 = POINT/0.0, 0.3, 0.1
FROM/P0
CALL/DRILL, X=1.0, Y=1.0, Z=0.1, DEPTH=0.7
CALL/DRILL, X=2.0, Y=1.0, Z=0.1, DEPTH=0.7
GOTO/P0
```

with the definition

```
DRILL = MACRO/X, Y, Z, DEPTH
GOTO/X,Y,Z
GODLTA/0,0, -DEPTH
GODLTA/0,0, DEPTH
TERMAC
```

Example 3.7 (Complete APT milling program)

The complete program from the course material defines a profile, a milling macro and two tool calls:

```
PARTNO PART11
MACHIN/MILL, 3          ;machine selection
CLPRINT                 ;prints out CL data file
```

```
OUTTOL/0.002
SP =POINT/5,0,1
P1 =POINT/2,2,0.5
P2 =POINT/4,2,0.5
P3 =POINT/6,4,0.5
P4 =POINT/8,5,0.5
P5 =POINT/9,7,0.5
P6 =POINT/2,7,0.5
PL1 = PLANE/P1, P2, P3
PS = PLANE/PARALEL, PL1, ZSMALL, 0.5   ;define part surface to be z = 0
C1 = CIRCLE/CENTER, P4, RADIOUS, 1.0
L1 = LINE/P2, P3
L2 = LINE/P3, RIGHT, TANTO, C1
L3 = LINE/P5, LEFT, TANTO, C1
L4 = LINE/P5, P6
L5 = LINE/P6, P1
L6 = LINE/P1, P2
MILL = MACRO/CUT, SPIN, FEED, CLNT
CUTTER/CUT
FEDRAT/FEED
SPINDL/SPIN
COOLNT/CLNT
FROM/SP
GO/TO, L1, TO, PS, ON, L6
GORGT/L1, TO, L2
GORGT/L2, TANTO, C1
GOFWD/C1, TANTO, L3
GOFWD/L3, PAST, L4
GOLFT/L4, PAST, L5
GOLFT/L5, PAST, L6
GOLFT/L6, PAST, L1
GOTO/SP
TERMMAC
TURRET/4
CALL/MILL, CUT=0.52, SPIN=600, FEED=3.0, CLNT=ON
TURRET/6
CALL/MILL, CUT=0.5, SPIN=900, FEED=2.0, CLNT=ON
SPINDL/0
COOLNT/OFF
END
FINI
```

Explanation: PARTNO names the part; `MACHIN/MILL, 3` selects a three axis mill; `CLPRINT` prints the CL data file; `OUTTOL/0.002` sets the outside tolerance for approximating curves (0.002). Points P1 to P6 define the corners and SP the start point. The plane PL1 passes through P1, P2, P3 and PS is a plane parallel to PL1 offset in the ZSMALL sense by 0.5 to define the part surface (z = 0). C1 is a circle centered at P4 of radius 1.0, and L1 to L6 define the contour lines, including tangent lines to C1. The macro MILL takes the cutter, spindle speed, feed and coolant, then drives the contour with GO, GORGT, GOFWD and GOLFT statements and returns to SP. Two turret calls execute the macro with cutter 0.52, 600 rpm, feed 3.0 and then cutter 0.5, 900 rpm, feed 2.0 (roughing then finishing), after which the spindle and coolant are switched off.

3.16.6 Other Part Programming Languages

Table 3.20: Languages related to APT

Language	Origin and purpose
ADAPT	ADaptation of APT: first attempt to adapt the APT programming system for smaller computers.
AUTOSPOT	AUTOmatic System for POsitioning Tools, developed by IBM and first introduced in 1962.
EXAPT	EXtended subset of APT, developed jointly in Germany in about 1964 by several universities to adapt APT for European use; compatible with APT and can use the same processor.
COMPACT	Developed by Manufacturing Data Systems, Inc. (MDSI).
SPLIT	Sundstrand Processing Language Internally Translated, developed by Sundstrand Corporation for its own machine tools.
MAPT	Micro-APT, a subset of APT to be run on microcomputers.

Key Examination Points

- APT statement types: identification, geometry, motion, post processor, auxiliary.
- Motion statements: FROM, GOTO, GODLTA (point to point) and GO/-GOLFT/GORGT/GOFWD/GOBACK/GOUP/GODOWN (contour).
- Programmer defines geometry, sequence and tool path; the computer performs translation, arithmetic, cutter offset computation and post processing.
- Related languages: ADAPT, AUTOSPOT, EXAPT, COMPACT, SPLIT, MAPT.

3.17 Automated Programming of CNC Machine Tools

Automated programming means that the CAM system, not the programmer, generates most of the program. The textbook describes the technology of CAM as a sequence starting from the CAD file: a program is created from solid or surface models, or from data in SAT (ACIS solids), IGES, VDA, DXF, STL and ASCII files using suitable translators.

3.17.1 Procedure of CAM

1. Create a manufacturing model from the design model and the workpiece.
2. Set up the tool database; tools must be defined before an operation is performed (tool libraries can be created and retrieved, or tools created when the operation is defined).
3. Select the set up for the machining operation (a component may need more than one).
4. Define fixtures for each set up, displayed with the workpiece so that tool paths do not interfere. Fixtures consist of locating and clamping elements; for turning, appropriate chucking devices are used. Standard fixture elements can be held in a library.
5. Create a machinability database from which spindle speed and feed rate are selected.
6. Create the manufacturing operations to generate the CL data. As each operation takes place material is removed from the workpiece to simulate the actual operation, and the tool path is displayed for each operation.
7. If needed, modify the CL data by modifying the operation parameters or by editing the CL data file.
8. Create a manufacturing route sheet at the end of the session.
9. Post process the CL file to create the NC program.

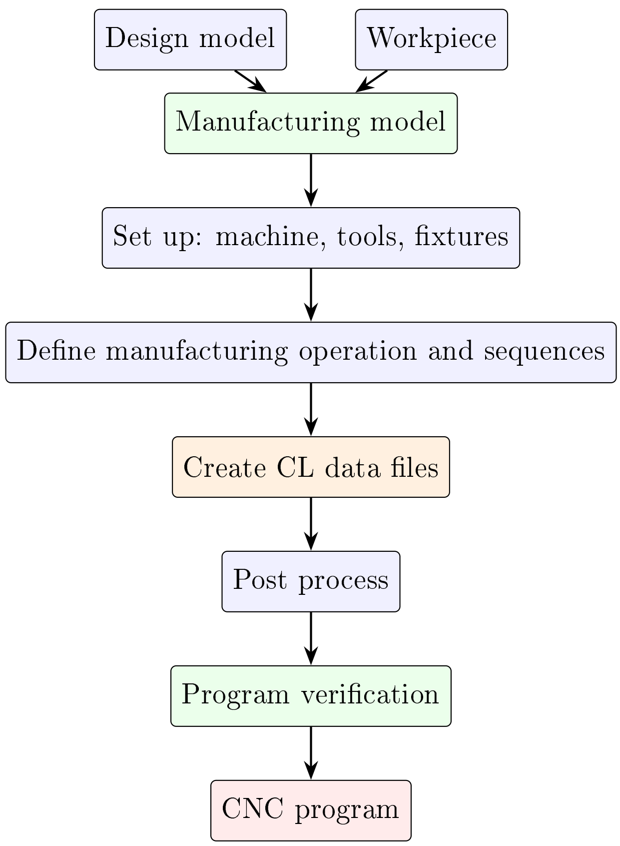


Figure 3.3: Steps involved in NC program creation using a CAM package

3.17.2 Design Model, Workpiece and Manufacturing Model

The CAD model represents the finished product and is the basis for the manufacturing operations. Features, surfaces and edges are selected on the design model as references for each manufacturing operation; referencing sets up a parametric relationship, so when the design model changes all associated manufacturing operations update. The workpiece is the raw material (bar stock, casting, billet, forging). It may be created by copying the design model and modifying dimensions or suppressing features. In the textbook example a cast aluminum part needs a datum surface (the bottom surface); dimensions of milled surfaces are increased by the machining allowance, bored holes below about 30 mm diameter are normally not cored and are machined from solid, and drilled holes are absent in the casting. The design model and workpiece assembled together form the manufacturing model. Small features such as threads, grooves, fillets and chamfers may be omitted from the workpiece because they are created during machining; for aerospace parts machining may start from a billet with allowances added to the envelope.

3.17.3 Coordinate System and Fixtures in CAM

Coordinate systems define the orientation of the workpiece on the machine and are the origin of CL data generation; they may belong to the workpiece or the design model. The positive Z axis should point away from the holding fixture. In milling the main cut feed direction is parallel to the X axis. In NC turning the Z axis is collinear with the turning axis, so the turning cut is sketched in the XZ plane.

3.17.4 Manufacturing Operations

Volume milling. A closed volume is removed (rough face milling or pocket milling); the tool always stays within the milling volume, defined by sketching the volume and intersecting it with the workpiece model.

Surface milling. A surface is selected for milling: three, four or five axis milling of surfaces, profiling, pocket finishing. Types: conventional surface milling (a tool path that completely machines the specified surface) and trajectory milling (slots, face milling, many end milling operations specified by a trajectory).

NC turning. The approach differs from milling; the cut is sketched in the XZ plane.

Popular CAM packages listed in the textbook: Pro/Manufacturing, CVCNC, I-DEAS Generative Manufacturing, UG Manufacturing, SurfCAM, Virtual Gibbs, EdgeCAM, Prospector, MasterCAM and Cimatron.

3.18 CAM Tool Path Generation Strategies

Tool path planning is a critical task in the machining of freeform surfaces. Specific constraints apply at different machining stages to achieve optimal time and quality. The final shape of a freeform part is obtained in three operations: roughing, semi-finishing and finishing.

3.18.1 Tool Path Topology and Parameters

Tool path planning has two aspects. *Path topology* is the pattern in which the cutter moves to produce the surface. *Path parameters* are the tool side step (step over) between successive paths and the tool forward step in each path. With a defined cutting tool the tool path generation problem is converted into three sub problems: (1) specify the path pattern and linking strategy (path direction), (2) specify the points on the surface the tool should track, and (3) check local and global tool interferences.

In finish machining, machining time should be minimised while the scallop height is kept below the specified level. An ideal tool path produces uniformly distributed scallops over the whole surface. Smaller scallops do not necessarily mean a better tool path, since this is achieved at the cost of increased machining time; the minimum machining time is achieved when the scallop height is set to the maximum allowable value.

3.18.2 Commonly Used Tool Path Distribution Strategies

1. Zig-zag or raster curves.
2. Contour curves.
3. Spiral curves.
4. Space filling curves.
5. Sequentially generated curves.
6. Radial curves.

Table 3.21: Standard tool paths

Tool path	**Description**
One way	Uses climb milling to obtain good surface quality.
Zig-zag	Both climb and conventional milling are used.
Helix curves	The tool trajectory goes from outside to inside with corner radii over the trajectories.
Radial	Similar to helix curves from the trajectory point of view, but with sharp corners.
Zig-zag at 45 degrees	The trajectory is inclined 45° to the X direction; linear movement is obtained by interpolating the X and Y axes. Used by some mold makers because the surface quality is good.

3.18.3 Roughing Strategies

Roughing removes bulk material quickly. Pocket milling removes material inside an arbitrary closed boundary on a flat surface to a fixed depth using flat end mills. The types of pocket milling are: spiral cutting from outside to inside, spiral cutting from inside to outside, and parallel cutting. Volume milling (closed volume removal) is the roughing form used for face and pocket milling in the CAM procedure. High speed roughing uses corner radii, trochoidal (small loop) motion and constant engagement so that the tool does not hit corners.

Strategies for successful tool paths:

- The tool path should be free from gouges.
- Sharp turns in cutter motion cause tool marks on the surface.
- Climb milling is favoured over conventional milling because of better quality, less noise, less vibration and higher tool life.
- The best way to approach the workpiece is by linear ramping or helical ramping (circular interpolation with helix).

3.18.4 Finishing Strategies

Finishing generates the final surface. Constraints are minimum machining time with scallop height kept within the specified level, uniform scallop distribution, avoidance of gouging, and, for free form surfaces, the use of zig-zag, contour, spiral or 45 degree zig-zag patterns. In the source, CATIA finish paths add corner radius, corner limit angle and transition radius, angle and length between passes.

3.18.5 Cutter Selection, Accessibility and Tool Holder Collision

Cutter selection is influenced by production cost, geometric constraints, machining quality, cutting tool life, machining accuracy and machine tool performance. For finishing, the material left after roughing is assumed to be small enough that the machine can support the given feed rates for all available tools, so the optimisation problem becomes: achieve the required machining quality at minimum production cost while the geometric constraints are satisfied. Surface quality directly affects machining time and is usually expressed through the scallop height.

Accessibility. A tool must reach every region of the surface. The minimum curvature radius ρ_{min} over all discretised surface points is found, and a tool whose radius r is smaller than ρ_{min} is selected. Small diameter tools are used for finishing but are also shorter, so even when the tool radius is acceptable the tool may be unable to enter a region because the *tool holder* would collide with the surface. Tool holder collision is therefore a constraint on finish machining, and collision detection must be carried out for both the tool body and the tool holder. The result is a *gouge free* cutter location path.

3.18.6 Side-Step, Scallop Height and Discretisation Tolerance

Definition 3.11 (Side-step, scallop and cusp)

The distance between two adjacent tool paths is the *side-step* (step over, g). The unmachined region left between two adjacent paths is the *scallop* (cusp), and the scallop height h is the maximum height of that region. The scallop height depends on the tool geometry, the surface geometry and the step over.

The side-step varies along the machined surface. It is normally computed *from* a prescribed scallop height: it is a function of the scallop height, the tool radius and the local radius of curvature of the surface. As an explanatory result (general knowledge, not taken from the slides), for a ball end mill of radius R on a flat surface the side-step is $g = 2\sqrt{2Rh - h^2} \approx 2\sqrt{2Rh}$; on convex or concave surfaces the local curvature modifies this value.

In milling, the cutter touches the surface at a cutter contact point and moves linearly to the next one, so a curved path is approximated by straight segments. The accuracy of this linear approximation is controlled by the allowed deviation, the *tolerance*. Consequences noted in the source:

- removing scallops afterwards by grinding or polishing is expensive and time consuming, so a good finish tool path reduces secondary operations;
- large scallops increase the time needed to smooth the machined surface;
- more line segments mean more machining time, so a tool path with fewer cutter contact points for the given tolerance and scallop height is desirable.

Table 3.22: Constraints on tool path design in roughing and finishing

Aspect	**Roughing**	**Finishing**
Main objective	Minimise machining time; remove as much material as possible	Keep scallop height within the allowable deviation given in the technical documentation
Surface quality	Not very important	Critical
Machining time	Dominant criterion	Not a dominant criterion
Pitfall		Scallop smaller than prescribed is not automatically better: more passes increase time and cost

3.18.7 Stages of Surface Machining and Cutting Force

Surface machining is classified into roughing, semi-finishing and finishing. In roughing most of the material is removed to generate an approximate shape. Shoulders left by large roughing tools are removed in semi-finishing to give a continuous offset surface for finishing. In finishing the surface is brought to its exact shape. In roughing, one applied method is *feed rate scheduling* that keeps the cutting force at a constant value, which reduces total machining time considerably compared with keeping the depth of cut and feed constant. Such optimisation is justified only in roughing because in finishing the depth of cut, and so the cutting force, is very small.

Cutting forces are the main factors governing machining accuracy, surface quality, machine tool vibration, power requirement and tool life. Predicting them is useful for designing machine tool structures and cutting tools and for controlling and optimising the process for accuracy and productivity. Tool deflection and force measurement setups are used experimentally to validate such predictions.

3.18.8 Tool Engagement Angle

The region of contact between tool and workpiece material is the engagement angle. In straight milling the step over s equals the radial immersion a_e, so the step over directly determines the engagement; at arcs and corners the two differ. When radial immersion changes, the chip thickness changes and so does the cutting force. With a constant step over tool path the engagement angle decreases at convex curvatures and increases at concave curvatures; for concave corners the cutting force can increase manifold. In difficult to machine materials even a small increase in engagement can be dangerous, so controlling the cutter engagement is essential. One solution is to control the feed rate. For a spiral pocket path the tool engages at 180°, 0° and 90° at different points when cutting outside to inside, and 180°, 180° and 90° when cutting inside to outside, so engagement is not constant.

For a line, the engagement angle in the material is

$$\theta_{\text{line}} = \cos^{-1}\left(\frac{r-s}{r}\right),$$

and for an arc

$$\theta_{\text{arc}} = \pi - \cos^{-1}\left(\frac{2r^2 + 2R(s-r) - s^2}{2r(R-r)}\right),$$

where s is the step over, r is the radius of the tool and R is the radius of the arc.

3.18.9 Comparison of Tool Path Approaches in CAM Software

Table 3.23: High speed tool path approaches in different CAM systems

System	**Approach**
Cimatron	Pocket, SRFPKT (surface pocket) and WCUT (water line cut). In pocket and WCUT, spiral cutting outside to inside with tool motions connected by loops at corners. SRFPKT creates tool motion on surface within a contour region using bidirectional parallel cutting; connection between passes can be inward loop, outward loop or golf club. Ramping uses circular interpolation with linear transition between passes and helical ramping to the next layer.

Table 3.23: High speed tool path approaches in different CAM systems

System	Approach
Unigraphics	Avoids sharp tool motion by filleting corners and stepping between two cuts; trochoidal cutting, milling in small loops resembling a stretched spring, with a path width and step over.
Mastercam	Trochoidal cutting with specified loop spacing and loop radius; the path looks like a spirograph and the cutter is surrounded by material on all sides, avoiding straight path movement.
CATIA	Corner radius in the tool path trajectory; limit angle for rounding corners; transition radius, angle and length from one pass to the next; the tool plunges directly to the next layer.

3.18.10 High Speed Machining Tool Path Issues

Traditional tool paths use linear interpolation (G01) with discontinuous transitions. The tool slows down at corners and stops briefly before the motor changes direction, which at high speed and feed affects machining time, creates friction and heat, gives overshoot on the tool path and gouges in the surface. High speed tool paths for pockets use arc based or spline based paths instead of cornered paths.

In an experimental comparison (spindle 20000 rpm, feed 1500 mm/min, radial depth of cut 2.5 mm, axial depth 1 mm) the source reports for chatter amplitudes measured at corners: traditional path about 3 microns, arc based path about 1 micron and spline based path about 0.3 micron. The spline path keeps the radial depth of cut almost constant, so feed rate variation is not needed and no chatter marks appear.

Table 3.24: Arc based versus spline based tool path (source study)

Factor	Arc based approach	Spline based approach
Radial depth of cut	Less in linear and more in arc interpolation; uneven tool movement causes gouging (extra metal removal)	Almost constant in linear and spline interpolation; no gouging
Average cutting force	78 N linear, 85 N circular	78 N linear, 81 N spline
Maximum stress induced	276.22 N/mm^2 linear, 630.976 N/mm^2 circular; work hardening of machined areas	276.22 N/mm^2 linear, 426.53 N/mm^2 spline (peak 387.501 N/mm^2 in transient analysis)
Feed rate	Must be reduced when going from linear to circular because radial depth changes	No variation needed

Table 3.24: Arc based versus spline based tool path (source study)

Factor	Arc based approach	Spline based approach
Average dynamic displacement of cutter	About 15 micrometers linear, 20 micrometers circular	Variation smaller; chatter less felt

The source concludes that increased feed and axial depth of cut increase machine tool chatter, and that feed contributes most to cutting forces.

Key Examination Points

- Tool path = topology (pattern) + parameters (step over, forward step).
- Six distribution strategies: zig-zag, contour, spiral, space filling, sequential, radial.
- Pocket milling: spiral out to in, spiral in to out, parallel.
- Climb milling and helical or linear ramping are preferred; avoid sharp corners.
- Engagement angle changes at corners; spline paths keep engagement nearly constant.

3.19 Tool Path Verification, Simulation and Program Debugging

Before real machining it is essential to simulate the machining virtually, to verify the geometry of the finished part and to predict physical factors needed to optimise cutting parameters. The development of simulation techniques for multi axis machining is key to the productivity and quality of parts with complex shapes (aerodynamic shapes, molds). Tool paths for such parts are generated taking into account cutting conditions, tool shapes and surface models. Simulation of the tool path (NC program) may use CNC editors, CNC simulators, CAD/CAM systems and CAM systems. All except CNC editors can verify the NC program by simulating the processing with a complete virtual model of the machine tool. Commercial software for verification and optimisation includes VERICUT, NC Simul, NC Brain and CAMWorks; Predator Virtual CNC and DELMIA Virtual CNC simulate and verify machine operation on a PC.

3.19.1 G-Code Verification: What Is Really Being Checked

NC program errors are a common cause of machine crashes. The two main sources of error are programmer mistakes and errors introduced by the CAM post processor. Manual proving out is slow, holds up the machine and risks missing an error, particularly on complex multi axis systems. An alternative is CAM simulation and verification software.

Table 3.25: CAM verification versus independent G code verification

Aspect	**CAM (CL data) verification**	**Independent G code verification (for example VERICUT)**
Data simulated	CL data that will later be post processed, so it simulates something different from what the machine will run	The post processed G code, the same code that runs in the NC system
Trust	Cannot be fully trusted; it projects what the machine will do; post processor errors are not seen; machine parameters (rotary behaviour, home and tool change positions) are hard to obtain	Verifies the exact code, so post processor bugs are caught
Machine model	Often a generic third party plug in, mainly for material removal rather than true machine motion	Machine features, components, fixtures and stock identical to the physical process; multiple set ups with cut stock transferred
Measurement	Usually cannot check dimensional accuracy of cut models	In process measurement (X-Caliper), AUTO-DIFF comparison of CAD design to simulation to find differences, gouges and excess material
Optimisation	Limited	Adjusts feed rates for each cut to shorten cycle time, reduce tool breakage and lengthen tool life

VERICUT supports look ahead or 3D cutter compensation, tool tip programming and tool length compensation, gage length reference point programming, canned cycles and fixture offsets, rotary axis pivot points, variables, subprograms, macros, and subroutines with looping and branching logic. Verification detects: incorrect or misread blueprints, inaccurate programming, incorrect tool path motions, rapid motion contact, collisions with fixtures and clamps, tool shank and holder collisions, and CAD/CAM and post processor bugs. Checking is done during production planning before any real machining starts.

3.19.2 Program Debugging

Debugging combines line by line review of the code, simulation on a virtual machine, independent verification of the post processed code and, finally, prove out on the actual machine with editing. The material stresses that debugging must include the post processor, since the post processor itself can introduce errors that could damage the part or the machine.

3.19.3 Virtual Machining

Virtual machining (VM) simulates the machining process on a computer before the real cut. It has three major components: computer aided process planning (CAPP), geometric modelling and process modelling.

1. A CAD model is the input to CAPP, where the machining operations, tool paths and process parameters (spindle speed, feed rate) are generated.
2. In *geometric modelling* the NC tool paths are verified to detect uncut material and gouges on the final surface and to prevent cutter and workpiece collisions. The cutter workpiece engagements (CWE), the removed volume and the chip geometry are extracted here.
3. In *process modelling* cutting force, power and torque are predicted using the laws of metal cutting and the CWEs from geometric modelling. Instantaneous cutting force depends on feed rate, spindle speed and CWE, and is a key input to simulating machine tool vibration (chatter) before real machining. The simulation can be used to optimise instantaneous process parameters to avoid chatter and improve quality.
4. The optimised process parameters and tool paths are sent to the CNC machine to produce the final workpiece.

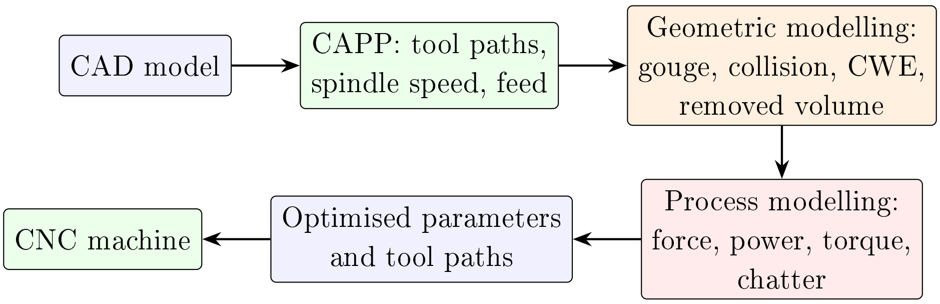


Figure 3.4: Components and data flow of virtual machining

3.20 CL Data, Post Processing, Program Transformation and Code Optimisation

Definition 3.12 (CL data)

Cutter location (CL) data is the machine independent description of the tool path generated by the CAM system (or by APT). It is fed to a post processor that produces the machine specific NC format.

The current CAx data model interface has two stages. Interface 1 is between CAD and CAM, with interface file formats such as IGES, SET, VDA and STEP. Interface 2 is between CAM and CNC: the CAM output (interface file formats APT and CL) goes to a *post processor*, which generates the specific NC format (for example N10 G0 G90 G40 G21, N20 G54 X45 Y25 S500 M3, N30 G43 Z100 H1) in the ISO 6983 or RS274D format. That format carries low level information, has limited scope and has vendor specific extensions.

Figure 3.5: Conventional CAD/CAM/CNC data flow and interface file formats

3.20.1 Cutter Contact Data and Cutter Location Data

A part to be machined is designed in CAD/CAM software as surfaces, and the tool trajectories are generated by discretising these surfaces within an allowable tolerance.

Definition 3.13 (Cutter contact (CC) point and cutter location (CL) point)

A *cutter contact point* is a point on the tool path at which there is direct contact between the tool and the part being manufactured. A *cutter location point* is a fixed reference point on the tool (the tool reference point) that is carried along the tool path; the CL data give its position and the orientation of the tool with respect to the workpiece coordinate system.

Generation of CL points from a surface (as described in the source):

1. **Discretise** the surface into cutter contact points according to the tolerance settings. The required geometric information (the CC point and the surface normal) is taken from the CAD model.
2. **Compute the tool axis** by applying a rotational transformation to the surface normal vector through the *lead* and *tilt* angles.
3. **Obtain the CL point** from the CC point by vector offsetting along the surface normal and tool axis by the tool radius. With a ball end cutter on a 3 axis machine the CL point lies along the normal to the surface at the CC point, at a distance equal to the ball radius, that is $\mathbf{CL} = \mathbf{CC} + r\,\mathbf{n}$ (general form; $\mathbf{n}$ is the unit surface normal).

To minimise manufacturing errors the CC point must lie on the designed portion of the surface, and CC points must be transformable to CL points so that machining error is reduced.

3.20.2 Tool Axis Orientation: Lead and Tilt Angles

For multi axis machining the tool axis is not fixed. It is obtained by rotating the local surface normal by a *lead angle* and a *tilt angle*. As an explanatory note (general knowledge), the lead angle is the inclination of the tool axis in the plane containing the feed direction and the surface normal, and the tilt angle is the inclination in the plane perpendicular to the feed direction; they avoid cutting with the tool centre of a ball end mill, and help avoid gouging and tool holder collision. In APT the orientation appears as the I, J, K components of the tool axis (Section 3.16).

3.20.3 CL Data Files and Their Role

CL data are generated without considering the structure of the multi axis machine tool: the part is assumed fixed and all motions are performed by the cutter. Different machine structures therefore share the same CL data (position and orientation of the cutter relative to the workpiece coordinate system). CL data are stored in a machine independent file, the Cutter Location Source file (CLS or CLSF), an ASCII file containing the coordinates of the geometry and auxiliary codes; commercial CAM systems usually output it in ISO format (for example the CLS file converted from the operations in UG CAM). APT is not only a language but also the program that processes APT statements to compute cutter positions and machine tool control commands; the programmer's viewpoint is that the workpiece is stationary and the tool moves relative to it, with speeds, feeds, tool calls and circular interpolation tolerances specified to complete the program.

Table 3.26: CC data, CL data and NC code compared

Aspect	**CC / CL data**	**NC code (G code)**
Dependence on machine	CL data are machine independent (part fixed, tool moves)	Machine and controller dependent
Content	Tool position and tool axis orientation (x, y, z, i, j, k) in the workpiece coordinate system; CC points on the surface	Motion type (G01, G02) plus the next location; axis values such as X, Y, Z, A, B or C
Produced by	CAM system (discretisation within tolerance)	Post processor from CL data
Format	CLS or CLSF (ISO format); APT CL data	ISO 6983 (RS 274D) with vendor extensions
Role	Describes the tool path	Instructs the machine to hold the cutter at each location

A note on terminology: the machine tool slides describe a *cutter location* loosely as the position the milling machine is instructed to hold the cutter at by the program (for example `G01 X20 Y1.3 Z4.409`). In CAM usage the term CL data means the machine independent tool path record that precedes post processing. Both descriptions are consistent: the G code line is the machine dependent expression of a CL point.

3.20.4 Post Processing

NC post processing converts the CL data file into the machine specific NC program. The post processor formats words, applies the axis and control conventions of the specific machine and controller, inserts G and M codes and outputs the file in the format the controller accepts. It is a possible source of errors, which is why independent verification of the post processed code is recommended.

3.20.5 Post Processor Development and Machine Kinematics

Definition 3.14 (CNC post processor)

A CNC post processor is software that converts the tool paths created in a CAM system into an NC program that the machine controller can read, so that the cutting tool moves along the programmed paths in a safe, consistent and predictable manner. It is the interface linking the CAM system with the NC machine: it reads and interprets the manufacturing instructions from the CAM system and converts them into NC code appropriate to the combination of machine and controller.

Tool path data are held in machine independent files that contain all the information required to move the cutting tools, but converting them into a controller specific format is often not enough. The post processor must also consider the *kinematics* of the machine, that is, how its axes move, especially the rotary axes of a five axis machine. Rotary axes may be attached to the table, to the spindle or to both; this configuration strongly affects how CAM data must be post processed, including length compensation, feed rate calculation and transitions between tool paths. A good post processor therefore generates edit free NC programs specific to the machine controller and kinematics and uses the latest controller functions.

The three key elements needed to develop a post processor are the *kinematic model* of the machining centre, the *format of the CL data* and the *post processing strategy*.

3.20.6 Inverse Kinematic Transformation and Multi-Axis Post Processing

The tool trajectory results from the axis displacements of the machine tool. For every CL point computed by the CAM software, the tool position and the orientation of its axis relative to the surface are obtained by combining the joint (axis) positions. From the programmed tool path the *inverse kinematic transformation* (IKT) calculates the corresponding axis configuration. Two calculation modes are possible: the IKT may be computed in real time by the NC unit during machining, or off line by a dedicated post processor.

The main functions of a post processor are therefore (i) to understand and interpret the CL data generated by the CAM software and (ii) to transform the machine independent CL data (x, y, z, i, j, k) into machine dependent NC commands such as (X, Y, Z, A, B), (X, Y, Z, A, C) or (X, Y, Z, B, C) according to the machine structure. (As a general remark, one tool axis vector can usually be reached by two different rotary axis solutions, so the post processor must also choose between them and avoid discontinuities.)

Two post processing methods are described for the Post Builder type tool. In the first, the tool tip position and tool axis direction from the CAM tool path are used to produce the CLSF, then a specific machine post processor produces the NC code for the machine axes of the given structure and controller. In the second, the post processor reads the cutting path data directly and translates it into NC code.

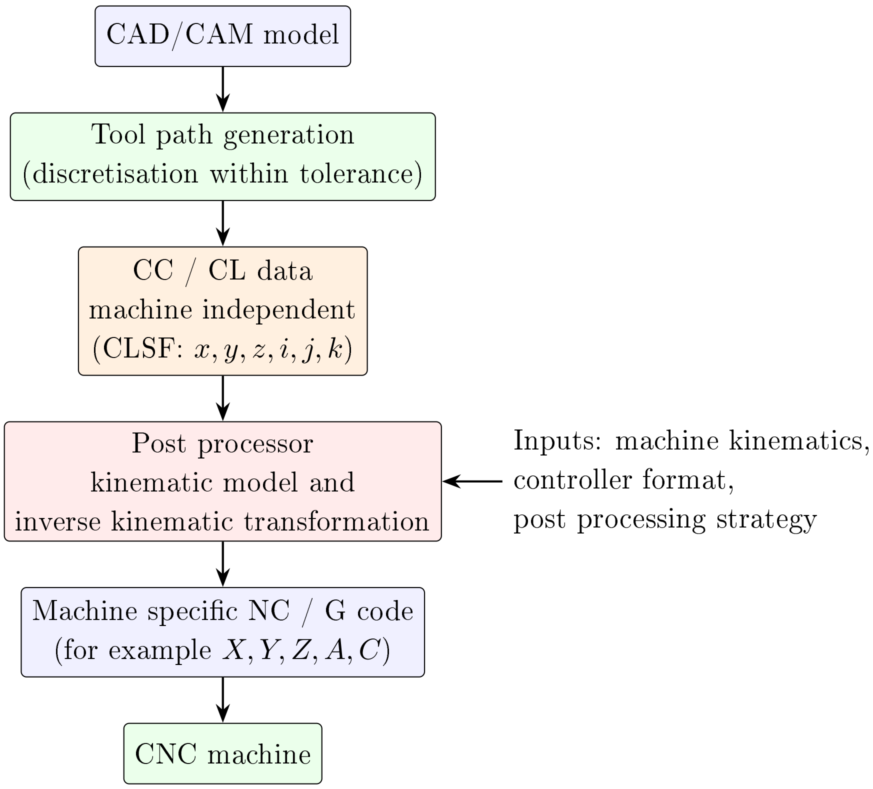


Figure 3.6: From CAD/CAM model to machine specific NC code

3.20.7 Limitations of Current Post Processors

The source lists expectations for future post processors: current post processors do not take tool collision constraints into account; verification of tool collision is therefore needed; real time compensation is desirable; and the lack of controller capability to execute real time compensation limits what can be done. This connects directly to independent G code verification (Section 3.19).

3.20.8 Program Transformation and Realisation

Program transformation and realisation covers the conversion of the tool path from its CAD/CAM representation into an executable program. The material identifies the steps: CL data is transformed into NC blocks by the post processor, the blocks are transferred to the CNC (DNC transfer or Ethernet) and loaded into the controller buffer for execution, where blocks are interpolated and split into movements of the different machine axes.

Table 3.27: Data transfer and interpolation issues in high speed machining

Concept	Description
Chordal deviation	Deviation of line segments from the arc. Point departures are the distances between successive point to point moves. High speed machining of an arc gives smaller chordal deviation but larger point departures, that is, a mass of data.
Data throughput	DNC transfers blocks at 960 characters per second (about 48 blocks per second), which requires faster input to move the tool quickly. Ethernet offers transfer of 1 or 10 million characters per second.
Interpolation time	The time required to create interpolation segments; it limits how small the increments can be at any feed rate. Interpolation accuracy meeting the required tolerance improves as increments shrink: with a chord of length 1 and a departure of 0.004, $AB = \sqrt{1 - 0.004^2} = 0.999992$ and $BD = 8 \times 10^{-6}$; with a departure of 0.002, $AB = 0.999998$ and $BD = 2 \times 10^{-6}$.
Feedrate and acceleration fluctuation	At joints of linear and circular segments axes see a step velocity change, which needs infinite acceleration and saturates motor drives. Interpolators therefore use S-curve velocity profiles, decreasing feedrate gradually near a corner and accelerating again after it. This lowers average feedrate and causes vibration, which worsens for the high density short segments used on free form surfaces.

3.20.9 Code Optimisation

Code optimisation reduces data volume and cycle time while keeping tolerance. Measures mentioned in the source are: replacing many short linear segments by NURBS blocks (a typical NURBS block can replace five to ten blocks of straight line moves), tool path smoothing, look ahead feedrate scheduling, optimisation of feed rates for each cut by verification software, ordering of operations to reduce tool changes, rapid distance and number of tools used, and use of the machining knowledge base.

Table 3.28: Toolpath smoothing techniques

Classification	Meaning
Local smoothing	Works with two adjacent segments.
Global smoothing	Acts on a group of segments.

Table 3.28: Toolpath smoothing techniques

Classification	Meaning
Curve based smoothing	A two step solution: a parametric curve connects two segments at the corner or approximates a group of segments, then a specially designed parametric curve interpolator interpolates it.
Velocity blending	A one step solution that schedules the velocity profile of each axis directly at the corner and creates a smooth transition.

The degree of continuity is used to evaluate smoothing algorithms. Linear and circular segments remain widely used on the shop floor because of straightforward coding and good CNC support, but they are becoming the bottleneck of high speed, high precision machining of free form surfaces.

3.20.10 Look Ahead and High Speed CNC Controls

With *look ahead*, the CNC controller keeps the feed rate as close to the maximum programmed rate as possible without violating geometric integrity, planning the feedrate by looking ahead in the path. Controller functions in multi axis machine tools are NC code interpretation, look ahead feedrate scheduling, tool path interpolation and servo motion control. In sharp corners and curvature areas the feedrate must slow down to reduce chord error. The performance of a high speed machine is limited by its slowest link, often the CNC control; a key measure is block processing speed (time per block, or blocks per second). Required high speed CNC features listed are: block look ahead of 150 blocks, high speed servo update, automatic jerk control, high speed PLC, spline interpolation algorithms, post interpolation filters to remove resonance, gains and adjustments controllable through parameters or the part program, N-blocks horizon to avoid tool collision, Ethernet and RS 232 communication, large on board hard drive, open control for OEM customisation, tool life management and high speed digital fibre optic communication between drive and motor.

3.21 Machining of Free Form Surfaces

Modern CAD/CAM systems allow designers to use free form or contoured shapes. Conventional CNC machines provide only linear (G01) and circular (G02, G03) interpolation, so CAM systems create many linear and circular segments to approximate the contour within a tolerance before the NC code is sent. As accuracy requirements tighten:

- tighter tolerances give shorter data segments and a larger volume of data;
- data transmission cannot keep up in high speed machining;
- feed rate fluctuation and velocity discontinuity occur at the junction of two segments;

- acceleration discontinuity and excessive jerk cause system vibration and reduce machining quality.

To overcome these drawbacks, parametric curves and surfaces, from early Bezier to B-splines and NURBS, have been adopted by CAD/CAM and CNC systems. Since NURBS can represent both free form and analytical geometry, it has emerged as the standard geometry representation tool. Only a few top grade CNC systems such as FANUC and SIEMENS support NURBS spline interpolation.

3.21.1 Surfaces, Interpolation and Approximation

A surface is a two dimensional set of continuous points with no thickness. While interpolation produces a curve or surface that contains the given data points, it may oscillate through every point. Approximation overcomes this: the curve or surface captures the shape of the data without containing all the points.

3.21.2 Bezier Curves

Definition 3.15 (Bezier curve)

A Bezier curve is a parametric curve defined by a set of control points, related to Bernstein polynomials and named after the French engineer Pierre Bezier who used it in the 1960s to design car bodywork. Two points are the ends of the curve and the other points determine its shape.

The curve always lies inside the convex hull of the control polygon and never oscillates wildly away from the control points. With end points A and D and control points B and C the curve is cubic; with one control point it is quadratic (useful with limited processing power). It is constructed by repeated linear interpolation with the same parameter t: interpolate between A and B, B and C, C and D to get Q, R, S; then between Q and R, and R and S to get P and T; finally between P and T to get the point U on the curve. In Bernstein form a curve of degree n with control points $\mathbf{P}_i$ is

$$\mathbf{C}(t) = \sum_{i=0}^{n} \binom{n}{i} (1-t)^{n-i} t^i \, \mathbf{P}_i, \qquad 0 \leq t \leq 1.$$

Its disadvantage is global control: moving any control point changes the shape of the whole curve.

3.21.3 B-Spline Curves

The B-spline concept resolves the disadvantage of the Bezier curve: control points give *local* control, so changing a control point affects only a specific segment of the curve. Spline interpolation uses low degree polynomials in each interval and chooses the pieces so that they fit smoothly together; its advantage is higher accuracy with less computational effort. A composite curve may comprise, for example, a line, an arc, a Bezier curve, another line and an arc.

3.21.4 NURBS

Definition 3.16 (NURBS)

Non-Uniform Rational Basis Spline (NURBS) is an accurate mathematical representation of curves and surfaces. The shape is defined by control points and the degree of the surface in each of the two directions (u and v).

Its key benefit is that it is not an approximation of a smooth shape: the mathematics gives an exact definition, smooth however closely examined. A NURBS surface is defined by a network of control points that pull the surface patch into shape like a flexible sheet. NURBS represents anything from a 2D line, circle, arc or curve to a complex 3D free form surface or solid. A NURBS curve is

$$\mathbf{C}(u) = \frac{\sum_{i=0}^{n} N_{i,p}(u)\, w_i\, \mathbf{P}_i}{\sum_{i=0}^{n} N_{i,p}(u)\, w_i},$$

where $N_{i,p}$ are B-spline basis functions of degree p, $\mathbf{P}_i$ are control points and w_i are weights; a NURBS surface uses the tensor product in u and v.

Table 3.29: Bezier, B-spline and NURBS compared

Feature	**Bezier**	**B-spline**	**NURBS**
Control	Global: moving one point changes the whole curve	Local: only a segment changes	Local, with weights
Defined by	Control points and Bernstein polynomials	Control points and piecewise polynomial pieces	Control points, degree in u and v, weights and knots
Represents analytic shapes exactly	No	No	Yes (line, circle, arc, free form)
Support in controllers	Interpolation available on some controls	Spline interpolation	G06.2 on FANUC type controls

3.21.5 NURBS Interpolation in the CNC

NURBS interpolation consolidates information that would otherwise be expressed through several chords into one block. It is selected with G06.2 in high precision contour control mode. G06.2 is a modal G code of group 01; the mode ends when another group 01 G code (G00, G01, G02, G03 and so on) is specified, and it must end before the command that ends high precision contour control mode. CAD systems that design metal dies for automobiles and aircraft express sculptured surfaces with NURBS, and this function lets the NURBS curve be specified directly to the CNC. The CAM system creates the NURBS curve from the NURBS expression output by the CAD system after compensating for the tool holder length, tool diameter and other tool elements. Advantages:

- no error due to approximating a NURBS curve by small line segments,
- a short part program,
- no break between blocks when small blocks are executed at high speed,
- no need for high speed transfer from the host computer to the CNC.

Where NURBS is used, the curve may be divided into segments according to curvature with a different feedrate limit for each segment; the look ahead module then computes the final feedrate of a segment considering its length and the following segments.

3.21.6 Surface Representation Relevant to Machining

Machining of surfaces relies on surface representations built from Bezier, B-spline or NURBS patches, whose control net defines the surface. The CAM system generates tool paths (scallop controlled zig-zag, contour, spiral and other patterns) on these surfaces using a tool with defined shape, and checks local and global interference. Machining is done in roughing, semi-finishing and finishing operations on multi axis machines (three, four or five axes), with the tool axis orientation controlled in the five axis case (I, J, K components of the tool axis in APT).

Key Examination Points

- Conventional CNC supports only G01 and G02/G03; free form surfaces need many short segments, which causes large data volumes and feed fluctuation.
- Bezier: global control, convex hull property; B-spline: local control; NURBS: exact, standard in CAD/CAM, G06.2 in FANUC type CNCs.
- NURBS block replaces five to ten linear blocks; no approximation error and shorter program.
- Look ahead, S-curve velocity profiles and toolpath smoothing (local or global; curve based or velocity blending) address feed fluctuation.

3.22 Automatic Feature Recognition

Definition 3.17 (Feature recognition)

Feature recognition means identifying the design geometry that represents holes, slots, pockets, bosses, fillets, chamfers and other design elements that can be machined.

Solid models have all the information needed to describe a part's shape, so users need not re-create part data for the CAM system. Most prismatic parts comprise standard features such as holes, pockets and bosses. Where other CAM systems require the user to define geometry, create boundaries and then specify cutting operations, automatic feature recognition (AFR) eliminates those steps; typical 2D parts can often be programmed in a few minutes rather than an hour or more.

3.22.1 Machining Features

Machining features include holes (drilling, counterbore), slots, pockets (roughing and finishing), bosses, steps, profiles, fillets and chamfers. In the STEP-NC example of the material a part has Hole 1 (drilling, counterbore), Hole 2, Hole 3, Boss 1 and Pocket 1 (roughing, finishing).

3.22.2 Feature Extraction and Recognition Procedure

Users typically run a wizard that automatically finds and sorts the different features. The input stock material size gives an initial tool orientation, and the software analyses the solid model and creates machinable features. These are listed and automatically sorted for logical machining operations. Alternatively users who want fine control can select features interactively and sort them manually. The best approach for an individual company is usually a combination of AFR with some interactive selection or sorting.

Table 3.30: Automatic versus interactive feature recognition

Aspect	Automatic	Interactive
Operation	Wizard finds and sorts features from the solid model	User selects features and sorts them manually
Speed	Fast; 2D parts programmed in minutes	Slower, but gives fine control
Best practice	Combine both: AFR plus interactive selection or sorting of features for company preference	

3.22.3 Feature Recognition as the Missing Link to Automated CAM

Part features in design geometry must be identified correctly and linked with the appropriate machining strategy; otherwise the result is what the source calls “white elephants on the shop floor”. Feature recognition complements knowledge based machining: identified machinable features can be linked to corresponding machining routines stored in a database, tool path generation (already highly automated in most CAM packages) follows, and the whole gives “push button CAM”. Feature recognition is essential to automating other manufacturing applications: cost estimates can be prepared with a few clicks and less guesswork, requests for quotes are fulfilled faster, and inspection routines for coordinate measuring machines can be generated quickly. It also plays a key role in creating a digital product model that can be used directly as part program input, bypassing CAM programming altogether (STEP-NC).

3.23 Knowledge Based Machining and Automated CAM

Definition 3.18 (Knowledge based machining (KBM))

Knowledge based machining is machining intelligence built into the CAM software. It is used at various levels and degrees in different CAM systems and provides push button automation while retaining control over the machining process.

Machining intelligence in the program reduces programming time and produces more consistent NC code. Combined with feature based machining it provides a high level of automation that is flexible to design changes and to a wide range of parts. When a part model is defined by a set of features, the CAM system analyses individual features and their relationship to one another. Automation occurs not only within individual features but between all features of a part, resulting in tool path optimisation. As features are created the CAM system constructs a process plan: operations are ordered on the basis of a system analysis to reduce tool changes, rapid distance and the number of tools used.

Table 3.31: Knowledge based machining features and truly intelligent CAM

Knowledge based machining provides	**Truly intelligent CAM systems additionally have**
Predefined machining rules adaptable to the user's preferences; a database of standard tools; a database of preprogrammed speeds and feeds; a database of standard materials	No requirement for users to create databases of standard machining data; no library of static machining processes because operations are dynamically created; no hand coding before the NC program is sent to the machine

3.23.1 Integration of Feature Recognition, Knowledge and Machining Strategy

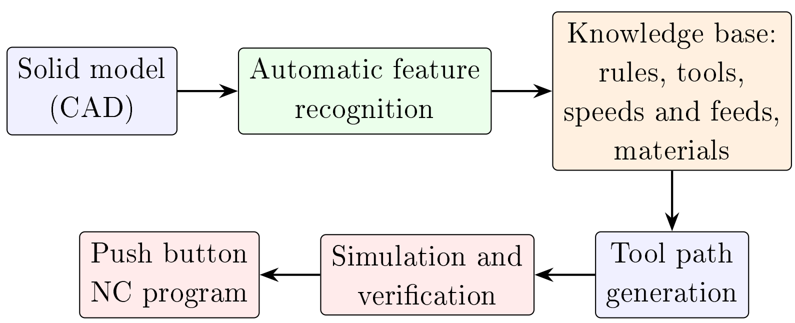


Figure 3.7: Feature recognition, knowledge based machining and tool path generation combine into push button CAM

3.23.2 Tolerance Based Machining

Tolerance enters CAM in several ways in the course material. First, the CAM system approximates curved geometry by line segments within a tolerance defined on the outside of the nominal curve; in APT the outside tolerance is set with statements such as `OUTTOL/0.002`, and a smaller tolerance produces shorter segments and more data. Second, in finish machining the scallop height must be kept below the specified level while machining time is minimised, so the allowable scallop tolerance controls step over. Third, the integrated CAD/CAPP model treats tolerance and surface roughness as non-geometric information attached to features: dimension and tolerance specifications are extracted or supplemented in the CAD model so that process plans can select processes able to hold them (see Chapter 5). Fourth, independent verification compares the cut model with the CAD design (AUTO-DIFF) to detect gouges and excess material so that the final component reflects design intent. Fifth, STEP-NC carries tolerance data as part of the product model (see Chapter 6). NURBS interpolation removes the tolerance loss caused by chord approximation.

3.23.3 Advantages and Limitations of Automated CAM

Table 3.32: Advantages and limitations of automated CAM

Advantages	**Limitations (as noted in the source material)**
Less programming time; more consistent NC code; flexible to design changes; automation between features gives optimised tool paths; cost estimates, quotations and CMM inspection routines generated quickly; parts cut as fast as possible with efficient tool use; programming becomes as automated as the latest CNC machines	Feature recognition must be correct and correctly linked to strategy or wrong parts result; CL data based verification can differ from what the machine runs; post processor errors may remain; the best results come from combining AFR with interactive selection; NURBS interpolation is supported only by a few top grade CNC controls; data exchange still meets incompatible formats and post processing issues

3.23.4 Case Study and Productivity Perspective

The source notes that what can be gained from higher spindle speeds, faster cutting tools, pallet shuttles and set up efficiency has plateaued. For many shops CNC programming is the "elephant in the doorway" holding up workflow and tying up resources. Programming tools such as AFR and other steps toward fully automated CAM signal a shift in the route to increased productivity. An NC simulation case study with NC Simul software illustrates verification of the machining before use on the machine, and closed loop manufacturing processes feed inspection results back to programming.

3.24 Chapter Summary and Examination Points

Key Examination Points

1. NC components: program, controller, machine tool. Open loop (stepper, no feedback) versus closed loop (servo, transducer feedback).
2. CNC versus DNC: dedicated computer at the machine versus central computer serving many machines.
3. Coordinate axes: Z along the spindle; right handed X, Y, Z; rotary A, B, C; G90 absolute, G91 incremental.
4. Word address format: N, G, X Y Z, F, S, T, M. Know the G and M code tables, canned cycles G81 to G86, circular interpolation with I J K or R.
5. Manual programming procedure and worked examples (drilling program O0001; four pass milling program; manuscript program).
6. APT: five statement types; geometry definitions for points, lines, circles, planes; motion statements; macro facility; related languages ADAPT, AUTOSPOT, EXAPT, COMPACT, SPLIT, MAPT.
7. CAM steps: manufacturing model, tools, set ups, fixtures, machinability data, operations, CL data, post processing, verification.
8. Tool path: topology plus parameters; six distribution strategies; roughing (pocket, volume, trochoidal), finishing (scallop control); tool engagement angle.
9. Verification: CAM verification on CL data versus independent G code verification on post processed code.
10. CC and CL data: CL = machine independent (part fixed, tool moves); CC on surface; tool axis by lead and tilt angles; post processor uses kinematic model, CL format and strategy; inverse kinematics maps (x, y, z, i, j, k) to axis values; virtual machining = CAPP + geometric modelling + process modelling.
11. Free form surfaces: Bezier (global), B-spline (local), NURBS (exact, G06.2); look ahead; smoothing.
12. Automation chain: automatic feature recognition, knowledge based machining, tool path generation and verification give push button CAM; STEP-NC is the end point of this chain.

Chapter 5

Computerized Manufacturing Planning and Control

5.1 Introduction to Process Planning

Process planning deals with the sequence of the different operations required to manufacture a given work part. It works as a bridge between the design and manufacturing functions. Planning begins with engineering drawings, specifications, a parts or material list and a forecast of demand, and it translates design information into the process steps and instructions needed to manufacture the product efficiently and effectively. Process planning is an important stage of product development because production tooling such as jigs, fixtures and special tools can be designed only after the process is finalised. Process plans have a direct bearing on the cost of the part; because new processes and machines keep being introduced, process planning is a dynamic activity that must be updated as cost reduction is pursued. Cost information can be included in a CAPP system using an automatic time standards (ATS) system.

5.2 Definition and Objectives of Process Planning

Definition 5.1 (Process planning)

Process planning can be defined as the systematic determination of the detailed methods by which workpieces or parts can be manufactured economically and competitively from the initial stage (raw material form) to the finished stage (desired form). The activity of developing such a plan is called process planning. Geometrical features, dimensional sizes, tolerances, materials and surface finishes are analysed and evaluated to determine an appropriate sequence of processing operations.

Definition 5.2 (Process planning (textbook form))

Process planning is the function within a manufacturing facility that establishes which machining process and process parameters are to be used to convert a work material (blank) from its initial form (raw material) to a final form defined by an engineering drawing. It is also a set of instructions used to make components so that design specifications are met. Manufacturing planning, process planning, material processing, process engineering and machine routing are titles given to this topic.

Process planning is a common task in small batch, discrete part metal working industries. Its objective is to select and sequence processes, machines, tools and parameters so that the part meets the design specification at minimum cost and lead time.

5.3 Process Planning Activities

The activities required to generate a process plan are:

(a) Study the overall shape of the part; use this information to classify the part and determine the type of workstation needed.

(b) Thoroughly analyse the part requirements: part features, dimensions and tolerance specifications.

(c) Select the raw material and its best shape.

(d) Identify the datum surfaces to identify set ups.

(e) Select the machine tools and manufacturing equipment for each set up.

(f) Select the manufacturing operations and their sequence.

(g) Select jigs, fixtures and other work holding devices.

(h) Select the inspection instruments.

(i) Determine speed, feed and depth of cut.

(j) Prepare the final process plan document.

The textbook divides the activity into these steps: selection of processes and tools; selection of machine tools and equipment; sequencing the operations; grouping of operations; selection of workpiece holding devices and datum surfaces (set ups); selection of inspection instruments; determination of production tolerances; determination of the proper cutting conditions; determination of cutting times and non-machining times (setting time, inspection time) for each operation; and editing the process sheets.

5.4 Responsibilities of a Process Planner

- Part print analysis and symbols.
- Gathering the fundamental details of the product design.

- Selecting the machining processes.
- Selecting the proper machining tool.
- Sequencing the operations.
- Deciding on the inspection equipment.
- Determining appropriate production tolerances.
- Selecting proper cutting tools and cutting conditions.
- Calculating the overall times using work measurement techniques.

5.5 Process Plan, Route Sheet and Operation Sheet

All information determined by process planning is recorded on a sheet called the *process plan*, also called the operation sheet, route sheet or operation planning sheet. It gives the instructions for production of the part and contains the operation sequence, processes, process parameters and machine tools used. The operation sheet lists the production operations and associated machine tools for a work part or assembly. Some plans also give inspection tool details, jigs and fixtures, and a remarks column with instructions to the operator.

Table 5.1: Format of a typical process planning sheet (header: part number, part name, material; example part: a pin in cast iron FG 200)

No.	**Process details**	**Machine**	**Tool holder / tip**	**Cutting speed**	**Feed**	**Set up / process time**
01	Hold on chuck (HOC), centre drill (CC)	CNC lathe		m/min	mm/rev	min
02	Drill	CNC lathe				
03	Face and turn	CNC lathe				
04	Grooving	CNC lathe				
05	Hold on chuck	CNC lathe				
06	Hold on chuck	CNC grinder				

The header of a plan carries: part number, drawing number, revision number, classification code, IGES or STEP file number, standard plan identity, plan revision number, material, quantity, machining time and unit cost. A typical manufacturing sequence table has these columns: serial number, machine, operation description (assembly operations), set up time, production time, tool, cutting speed, feed rate, rpm and number of cuts or passes. Example rows: serial 04, CNC turning lathe, tool P40, cutting speed 240 m/min, feed 0.5 mm/rev, 1400 rpm, 4 passes; serial 05, CNC lathe drilling with a BW drill, 200 m/min, 0.2 mm/rev, 2200 rpm, 1 pass. The software provides standard strings for operation description, for example: hold between centres; rough turn xxxx dia. stock to xxxx dia. for entire length of part; rough turn xxxx dia. to xxxx dia.; finish turn xxxx dia. to xxxx dia.; finish face xxxx dia. and chamfer.

5.5.1 Outputs of Process Planning: Routings, Process Plans and Drawings

Process planning begins with engineering drawings, specifications, parts or material lists and a forecast of demand. Its results are of three kinds:

Table 5.2: Results of process planning

Output	**Content and use**
Routings	Specify operations, operation sequences, work centres, standards, tooling and fixtures. The routing is a major input to the manufacturing resource planning system: it defines operations for production activity control and the required resources for capacity requirements planning.
Process plans	More detailed, step by step work instructions: dimensions related to individual operations, machining parameters, set up instructions and quality assurance checkpoints.
Fabrication and assembly drawings	Drawings that support manufacture, as opposed to the engineering drawings that define the part.

Manual process planning depends on the manufacturing engineer's experience and knowledge of production facilities, equipment, capabilities, processes and tooling; it is very time consuming and the result varies with the person doing the planning.

5.6 Factors Affecting Process Plan Selection

The information required for process planning includes the geometric model of the part (engineering drawing or CAD model), the parts list, annual demand or batch size, accuracy and surface finish requirements (CAD database), equipment details (work centre database), data on cutting fluids, tools, jigs and fixtures and gauges, standard stock sizes, and machining data and data on handling and set up. Process selection matches requirements with process capability. *Process capability* depends on: the shape a process can generate; size limitations (boundaries of tooling, machine tool and fixtures); tolerances (dimensional and geometric); surface finish attainable; cutting force; and power consumption. The database of knowledge for each process includes the shape and size of part a process can produce, dimensions and geometric tolerances obtainable, surface finish attainable, material removal rate, relative cost and technological parameters. A *process boundary* is the limiting size, tolerance and surface finish for a process, obtained by careful control of cutting conditions (feed, speed, depth of cut). Cost and time models are single pass (depth of cut fixed) or multi pass (depth of cut also a control variable).

The dimensions and tolerance based selection rule quoted in the textbook for holes is: for a diameter up to 8 mm with true position greater than 0.01, tolerance coarser than H10 means drill the hole, and tolerance H8 or finer means drill and ream.

5.7 Manual Process Planning

In conventional (manual, experience based) process planning a process plan is created by a process planner and requires a significant amount of time and expertise to determine an optimal routing for each new part. Individual planners have their own opinions of the best routing, so operation sequences differ among planners. Problems of experience based planning: experience is gathered over time; experience may not represent correct and current knowledge; and no experience is available for new processes. Knowledge may also be stored in handbooks, data books and manuals (tables, figures and guidelines), and decision tables and decision trees describe the actions associated with combinations of input conditions.

5.8 Computer Aided Process Planning

Definition 5.3 (Computer aided process planning (CAPP))

Computer aided process planning is a link or bridge between product design and manufacturing design. The CAPP system is used to optimise variables such as cost, lead times, equipment availability, production volumes, potential material substitution, routings and testing requirements. CAPP is a key tool for engineers in the modern manufacturing industry.

Attempts to capture the logic, judgment and experience of the planner in computer programs led to CAPP: based on the features of a part, the program automatically generates the sequence of manufacturing operations and provides production routings that are rational, consistent and perhaps even optimal.

5.8.1 Comparison of Manual (Conventional) and Computer Aided Process Planning

Table 5.3: Conventional versus computer aided process planning

Aspect	Conventional (manual)	Computer aided (CAPP)
Accuracy and consistency	Less accurate, may not be consistent	Produces accurate and consistent results
Time	More time consuming	Less time consuming

Table 5.3: Conventional versus computer aided process planning

Aspect	**Conventional (manual)**	**Computer aided (CAPP)**
Skill	Requires expert and highly skilled persons	Does not require highly skilled persons
Dependence on experts	Totally dependent on expert knowledge	Not only dependent on expert knowledge
Clerical work	High routine clerical work	Reduces routine clerical work of the industrial engineer
Cost and lead time	More costly, high lead time	Highly productive
Productivity	Lower	Higher
Requirements		Requires a computer and supporting devices

5.8.2 Benefits of CAPP

CAPP *increases* productivity, consistency of plans, accuracy of plans, capacity utilisation and the ability to introduce new technology. It *decreases* the lead time of process plans, routine clerical work, manufacturing and process planning cost, work in process and the skill requirements of the planner. The textbook lists the advantages as: reduces the skill required of a planner; reduces process planning time; reduces process planning and manufacturing cost; creates more consistent plans; produces more accurate plans; increases productivity.

5.8.3 Structure and Operation of a CAPP System

The structure of a CAPP system consists of input (based on the solid model), processing and output (a process plan). The modules of the structure are: part classification; feature recognition; material selection; process selection; machine selection; fixture selection; end effector selection; process sequencing; cutter path generation; and intermediate surface generation. These draw on software databases: process capability, machinability data, machine tool data, standard material, standard time, standard fixture, standard cost, process plans and end effector data. Modules are not necessarily arranged in the proper sequence, and each may be executed several times to obtain the optimum plan. The generated and validated process plan can be routed directly to production planning and production control systems.

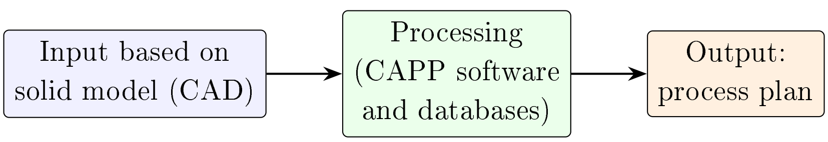


Figure 5.1: Structure of a CAPP system

Typical basic functions of CAPP software are process planning (create a new plan, retrieve a plan and edit, delete a plan, display a plan, list summary information), an initialisation function and report generation. A plan can be retrieved using any field of its header; SQL capabilities allow reports on parts with identical classification, parts using a specific machine type, parts machined using specific tool materials, and the bill of materials. An automatic time standards module takes inputs such as material (for example C45), hardness (for example 220 BHN) and work centre; it displays available tool materials (coated carbide, ceramic, cubic boron nitride, diamond coated tools), retrieves starting and finish diameter, cut length and depth of cut, and calculates number of passes, recommended depths, spindle speeds, power required and machining time; loading and unloading time modules also exist.

5.8.4 Limitations of CAD Based Process Planning

Geometric models (2D, surface, solid) do not readily identify surfaces requiring machining and cannot represent special manufacturing techniques such as coatings and surface treatment; a human computer interactive system is often used to identify the surface. A CAD front end needs: location and interactive identification of lines and surfaces; easy dimension retrieval; storage and retrieval of tolerance information; display of multiple objects; built-in special tags (such as drill angle for a hole); and combination of special process information (heat treatment, surface treatment, finishing) with the geometric model. Automatic feature recognition (AFR) highlights machining features one by one so the planner can define processes or check that no feature is missed. For precision components the process design is extremely complex; the inability to model the dynamics of the machine tool, material, workpiece and fixture system, the metallurgical behaviour of the work material, the non-deterministic nature of tool wear and tolerance accumulation are major bottlenecks.

Key Examination Points

- Definition: systematic determination of methods to convert raw material to the finished part economically and competitively.
- Ten planning activities (part shape study through final plan document).
- Nine planner responsibilities.
- CAPP is the link between CAD and CAM; two main approaches, variant and generative (plus semi-generative).
- Benefits: increases productivity, consistency, accuracy, capacity utilisation, ability to introduce new technology; decreases lead time, clerical work, cost, WIP, planner skill requirement.

5.9 Role of CAPP in CAD/CAM and CIM Integration

The role of CAPP is the planning strategy for manufacturing the part. It is the activity that translates part design specifications from an engineering drawing into the manufacturing operation instructions required to convert a part from a rough to a finished state.

It represents the link between engineering design and shop floor manufacturing and is a major determinant of manufacturing cost and profitability. The gap between CAD and CAM can be shortened considerably by developing better systems for process planning.

5.9.1 CIM Subsystems

Table 5.4: CIM subsystems and their technologies

Subsystem	**Function group**	**Technologies shown**
Product design	CAD-CAE	Computer aided design and engineering
Production planning	MRP, JIT, OPT	Material requirements planning, just in time, optimised production technology
Computer aided manufacturing	FMS, GT, CAPP, AGVS, AS/RS	Flexible manufacturing systems, group technology, process planning, automated guided vehicles, automated storage and retrieval

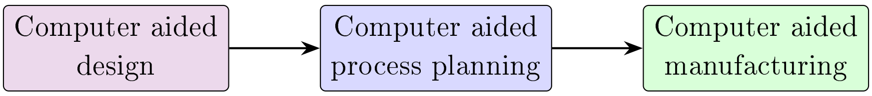


Figure 5.2: Interaction between CAD, CAPP and CAM

5.9.2 Flow of Information from CAD through CAPP to CAM

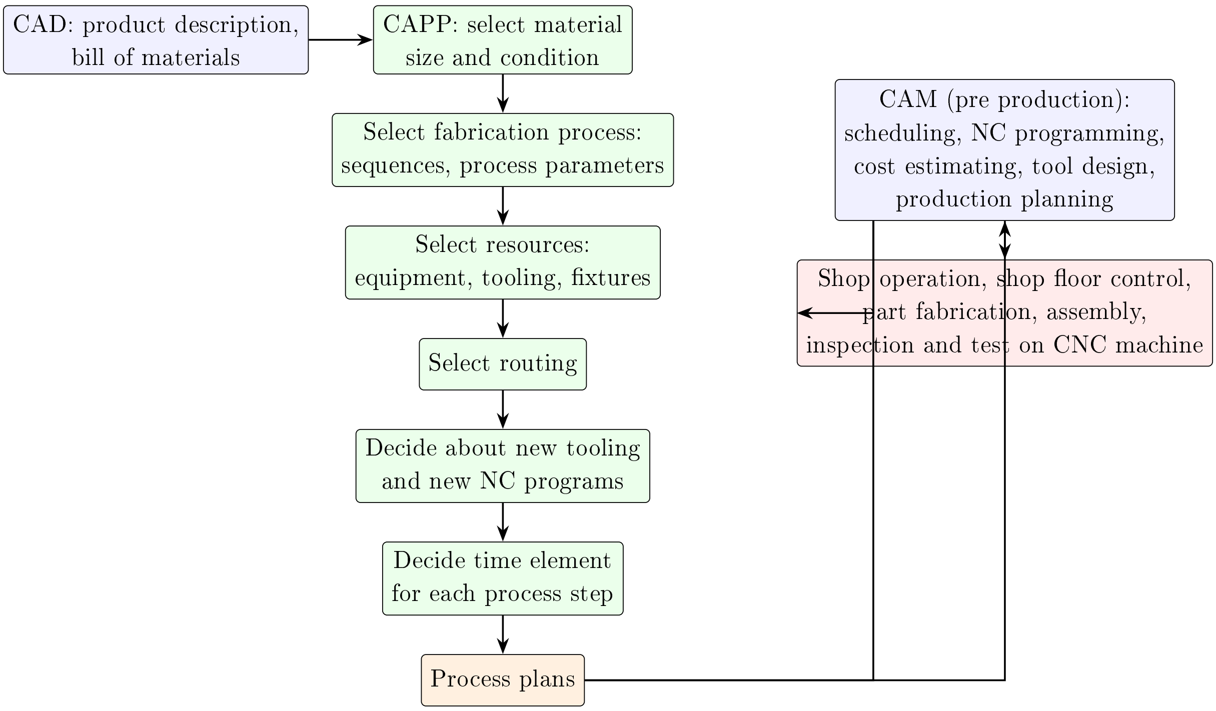


Figure 5.3: Role of CAPP between CAD and CAM

5.9.3 Framework for Computer Aided Process Planning

In the framework shown in the source the organisational planning system feeds the MRP (material resource planning and capacity planning) module with product design and development requests. CAD supplies the part list and geometry data; CAPP receives geometry data and parts master files from CAD, exchanges process plans with MRP, and draws on machine, tool and fixture data banks. MRP sends production orders to CAM and production control; production control returns actual data to MRP and corrected data to CAPP, and CAPP delivers the NC program to production control.

Table 5.5: Information flows in the framework of computer aided process planning

From, to	Data
Organisational planning system to MRP	Product design and development request
CAD to MRP	Part list
CAD to CAPP	Geometry data (connected data)
CAPP and MRP	Parts master files; process plans
CAPP and data banks	Machine, tool and fixture data banks (two way)
MRP to CAM/production control	Production order
Production control to MRP	Actual data
Production control to CAPP	Corrected data
CAPP to production control	NC program

5.9.4 Integrated CAD/CAPP Model

The activity model of the integrated CAD/CAPP system has five functional modules, from customer requirements to process plans.

Table 5.6: Modules of the integrated CAD/CAPP system

Module	Name	Function
A1	Product design and CAD modelling	The user finishes product design and CAD modelling according to customer requirements; the output is a geometry oriented CAD model.

Table 5.6: Modules of the integrated CAD/CAPP system

Module	**Name**	**Function**
A2	Feature recognition	Translates the geometry oriented CAD model into a feature based CAD model through feature recognition and/or feature conversion, feature parameter and constraint extraction, and feature precedence tree reconstruction (using a standard feature template).
A3	Technical information processing	Supplements or extracts the necessary technical specifications, such as location dimension and tolerance, surface roughness and hardness, which may not be recorded explicitly in the CAD model, to create a feature based product model with geometric and non-geometric feature information from which process plans are created.
A4	Process planning	Creates the process plans using process rules and constraints, the process database, machinability data, standard process models and machining resource descriptions; also produces comments on design and the integrated product information model (IPIM).
A5	Drawing and material stock creation	Automatically marks technical specifications and process plans on 2D engineering drawings and creates the corresponding 3D material stock geometry model, inter process stock model and removed allowance model.

Feature recognition and conversion (A2) and technical information processing (A3) are preconditions for process decision making. They transfer the geometry oriented CAD model into the IPIM, preserving the integrity and consistency of the feature model and product information so that downstream applications can share them. Modules run on computer and software platforms (MC).

5.9.5 Role of Computer Aided Inspection Planning (CAIP)

Besides CAPP there is computer aided inspection planning (CAIP). In the CAD/CAI/-CAM triangle CAPP links CAD to CAM and CAIP links CAD to CAI (computer aided inspection). Data flows: geometric data from CAD to CAM; inspection data from CAD to CAI, with error feedback from CAI to CAD; in process inspection results from CAI to CAM and process information from CAM to CAI.

Table 5.7: Data flows between CAD, CAM and CAI

From	**To**	**Data**
CAD	CAM	Geometric data (planning by CAPP)
CAD	CAI	Inspection data (planning by CAIP)
CAI	CAD	Error feedback
CAI	CAM	In process inspection data
CAM	CAI	Process information

Global and Local Inspection Planning

Definition 5.4 (Computer aided inspection planning (CAIP))

CAIP is the integration bridge between CAD/CAM and computer aided inspection (CAI). It plans how, in what order and with which measurements a part is inspected, using the same feature and geometry information that CAPP uses to plan manufacture.

A CAIP system has been proposed for *on-machine measurement* (OMM), that is, inspection of complicated mechanical parts efficiently during machining or after machining while the part is still on the machine. The inspection planning has two levels:

- **Global inspection planning (GIP)**: the system creates the optimal inspection sequence of the features of a given part by analysing the feature information. Feature groups are formed for effective planning and special feature groups are determined for sequencing. An *integrated process and inspection plan* is generated using a series of heuristic rules, so that the optimum manufacturing sequence for both inspection and machining operations is determined.
- **Local inspection planning (LIP)**: planning of the measurement details within an individual feature (the source names LIP together with GIP but gives detail mainly for GIP; the detailed content of LIP is not described in the supplied material).

The results are simulated and analysed to verify the effectiveness of the plan.

Conceptual Chain from CAD to Process Correction

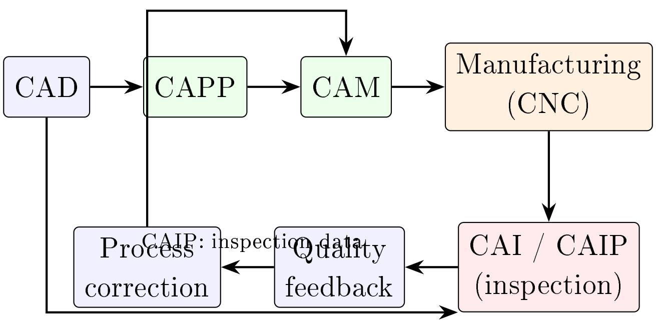


Figure 5.4: CAD, CAPP, CAM, manufacturing, CAI/CAIP, quality feedback and process correction

CAPP links CAD to CAM by transferring geometric data; CAIP links CAD to CAI by transferring inspection data. CAI returns an error check to CAD, and exchanges in process inspection results and process information with CAM.

5.10 Methodical Approaches to CAPP: Variant, Generative and Semi-Generative

CAPP methodologies are of three kinds. *Variant methods* are based on group technology (GT) and on the assumption that similar parts have similar technological processes; they work with a specific structure pattern, and their advantage is the possibility of obtaining a process typical for a given part. *Generative methods* are based on mathematical rules: on the basis of design knowledge written in the form of frames and rules, each time the technological process of a given technological element (part) is generated; their advantage is the possibility of generating a detailed machining process for a given production system. *Semi-generative methods* lie between the two, combining a retrieved standard structure with generated details.

Table 5.8: Comparison of manual, variant and generative process planning

Aspect	**Manual**	**Variant (retrieval)**	**Generative**
Basis	Experience of the planner	Group technology and coding; similarity of parts	Decision logic, mathematical rules, knowledge base
Plan origin	Written by the planner	Standard plan retrieved and edited	Synthesised automatically for a new part
Human role	Total	Editing of the retrieved plan	Minimal
Suits	Any, but inconsistent	Small components in part families, machine shops with variety of small parts	New components, interface to automated facilities
Consistency	Low	Good within a family	Good, rapid
Main input	Drawing	Part code	Geometric model with features

Table 5.8: Comparison of manual, variant and generative process planning

Aspect	Manual	Variant (retrieval)	Generative
Main limitation	Slow, planner dependent, high clerical work	Needs coding and families; poor for dissimilar parts	Needs complete captured manufacturing knowledge; hard to develop

5.11 Group Technology and CAPP

Definition 5.5 (Group technology)

Group technology is an operations management philosophy based on the recognition that similarities occur in the design and manufacture of discrete parts; similar parts are arranged into part families. To implement it some form of part classification and coding is required.

Group technology principles let code numbers retrieve existing and preferred manufacturing information. Similarities are of two types: design attributes (geometric shape and size) and manufacturing attributes (the sequence of processing steps). A major benefit is avoiding duplication of similar components with savings in design, processing and tooling cost.

5.11.1 Part Families

A part family is a collection of parts that are similar either because of geometry and size or because similar processing steps are required. The parts within a family differ but are close enough to merit identification as members. The three methods of grouping parts into families are: visual inspection, production flow analysis, and parts classification and coding systems. In a machine tool industry large part families might be grouped as heavy parts (beds, columns), shafts (large L/D ratios), spindles (long shafts, screw rods), non-rounds (small prismatic parts), and gears and disc type parts (small L/D ratios).

5.11.2 Part Classification and Coding

Systems are of three types: based on design attributes (design retrieval and standardisation), based on manufacturing attributes (CAPP, tool design and production functions) and based on both.

Table 5.9: Attributes included in classification schemes

Part design attributes	**Part manufacturing attributes**
Basic (external and internal) shape; axisymmetric, prismatic or sheet metal; length to diameter ratio; material; major dimensions; minor dimensions; tolerances; surface finish	Major process of manufacture; surface treatments and coatings; machine tool or processing equipment; cutting tools; operation sequence; production time; batch quantity; production rate; fixtures needed

Code structures. There are three basic structures: hierarchical (the interpretation of each succeeding symbol depends on the preceding symbols), chain type (the interpretation of each symbol is fixed and independent of preceding digits) and hybrid (a combination). Factors when selecting a coding system: component geometry (rotational, prismatic, deep drawn, sheet metal), code structure, digital representation (binary, octal, hexadecimal) and material (ferrous, non ferrous, plastics, composites). Coding systems used in process planning: OPITZ, CODE, KK-3, MICLASS, DCLASS and COFORM (coding for machining).

5.11.3 Opitz Classification System

The Opitz code has nine digits, extendable by four more: `12345 6789 ABCD`. Digits 1 to 5 are the *form code* (primary design attributes); digits 6 to 9 the *supplementary code* (work material, raw workpiece shape, accuracy and dimensions; useful to manufacturing); `ABCD` is the *secondary code* identifying the production operation type and sequence, designed by the firm.

Table 5.10: Meaning of the Opitz form code digits (rotational parts)

Digit	**Meaning**
1	Part class: 0 is $L/D \leq 0.5$; 1 is $0.5 < L/D < 3$; 2 is $L/D \geq 3$ (rotational parts); 3 to 5 are rotational parts with deviation or special (L/D and main shape); 6 to 9 are non-rotational parts (shape ratios such as A/B, A/C) and special
2	External shape and external shape elements (for example smooth, stepped on one end or both ends, threads, grooves, functional cone, functional operating speed)
3	Internal shape and internal shape elements (no hole or break through, smooth or stepped, threads, grooves, cone)
4	Plane surface machining (no surface machining, plane or curved surface, external plane surface, slots, splines, polygons, internal plane surface)
5	Auxiliary holes and gear teeth (no auxiliary holes, axial or radial holes on pitch circle diameter, spur, bevel or other gear teeth)

Example 5.1 (Opitz form code)

For the part with dimensions 20 dia, 25 dia and 5 dia with an M10 by 1.5 thread, $L/D = 1.6$, so digit 1 is 1. The part is stepped on both ends with a screw thread on one end, so digit 2 is 5. Digit 3 is 1 because of the through hole. Digits 4 and 5 are 0 since no surface machining is required and there are no auxiliary holes or gear teeth. The form code is **15100**. The supplementary code (digits 6 to 9) would code dimensions, material, starting workpiece shape and accuracy.

5.11.4 MICLASS and CODE Systems

Table 5.11: MICLASS and CODE classification systems

System	**Description**
MICLASS	Number of 12 to 30 digits; the first 12 are universal and up to 18 more are company specific (lot size, piece time, cost, operation sequence). First 12 digits: 1st main shape; 2nd and 3rd shape elements; 4th position of shape elements; 5th and 6th main dimensions; 7th dimension ratio; 8th auxiliary dimension; 9th and 10th tolerance codes; 11th and 12th material codes. Parts are coded interactively by answering computer questions (as few as seven for a simple part; 10 to 20 for an average part).
CODE	Developed by Manufacturing Data System, Inc. Eight digits, each with 16 possible values (0 to 9 and A to F). The initial digit is the major division (cylinder, flat, block, other); the remaining digits form a chain type structure, so the system is hybrid. Used in design retrieval, process planning, purchasing, tool design and inventory control.

5.11.5 Benefits of Group Technology

Table 5.12: Impact of group technology

Before	**After**
Discontinuous, random flow of parts through the shop	Structured flow of parts
Reinvent new parts	Retrieve parts already in production
Multitude of process plans for some parts	Consistent, single, best process plans
Continuous purchase of components due to lack of total visibility	Regulated purchase of components

Table 5.12: Impact of group technology

Before	**After**
Inflexible, rigid, unable to respond to a changing environment	Flexible

Benefits by function: *product engineering*: reduce part proliferation, help design standardisation, provide manufacturing feedback; *manufacturing engineering*: process selection, tool selection, machine purchases, material handling; *production engineering*: reduce lead time, delays and set up time, improve product quality; *production planning and control*: group scheduling, stock accountability, reduced expediting. Other benefits: increased productivity, improved accuracy in cost estimation, greater standardisation and variety reduction, reduced set up times, better delivery (helps just in time), reduced purchasing cost, improved plant efficiency, improved product design, reduced materials handling and better employee satisfaction.

5.12 Variant (Retrieval) CAPP

A variant process planning system uses the similarity among components to retrieve existing process plans. A plan usable by a family of components is a *standard plan*, stored permanently with the family number as key. A family is represented by a *family matrix* that includes all possible members. There are two operational stages: the preparatory stage and the production stage.

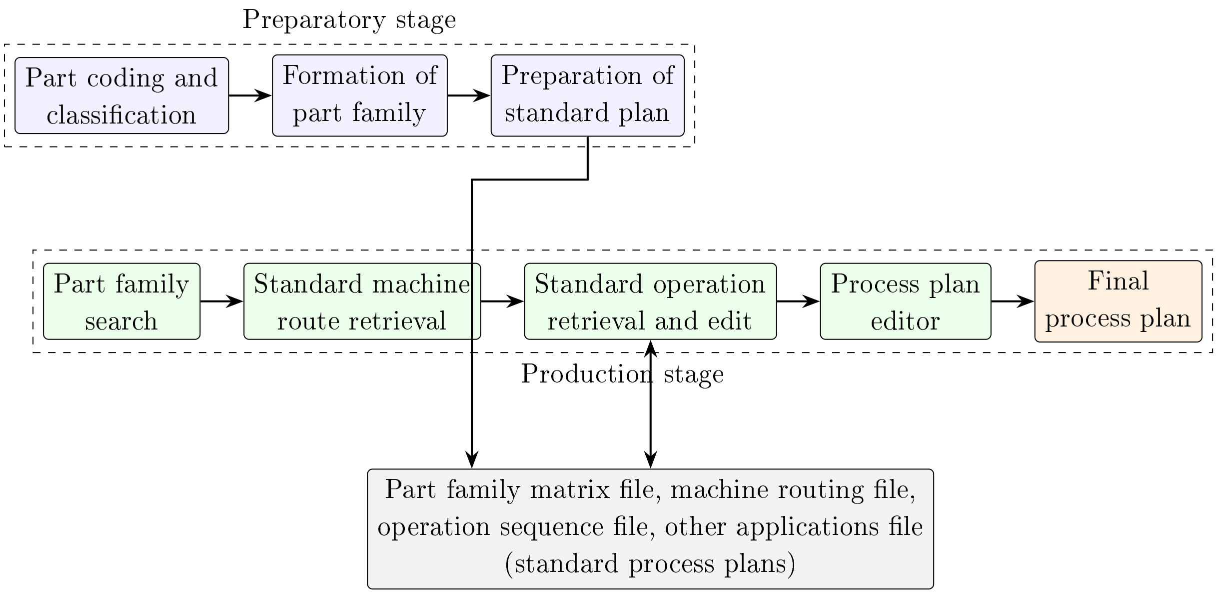


Figure 5.5: Variant or retrieval process planning

5.12.1 Procedure of Variant Process Planning

Preparatory stage (builds the database of standard process plans):

1. Classify the parts and label them with the appropriate code.
2. Prepare part families on the basis of design and manufacturing attributes.
3. Prepare the operation sheet or route card (standard plan) for each family.

Production stage:

4. Enter the new part with its part code.
5. The CAPP program searches the part family; if the file contains an identical code, the standard process plan is retrieved.
6. Edit the plan; add process parameters and standard times to issue the final process plan.

5.12.2 Design of a Variant System

The design sequence is: family formation, database structure design, search algorithm development and implementation, plan editing, and process parameter selection and updating. A general rule for family formation is that all parts must be related so that one standard plan can be shared with minimum modification. The database may be hierarchical, network or relational. Search is the matching of the family with a given code: family matrices act as *masks*, and whenever a code passes through a mask successfully the family is identified. Plan editing is of two types: editing the standard plan in the database, and editing the plan for the component. Process parameter files are linked so that the speed and feed of an operation can be found through the tree.

5.12.3 Framework of the Variant Process Planning Activity

The variant planning activity takes as inputs the finished part description, bill of materials, part number and process deviation, and uses the master variant file, process variant file and process selection logic. Outputs are the material requirement file, machine sequence file, machine tool file, tool fixture file, manufacturing times file and labour cost file. These feed an activity text generation step (with a text generator) which produces the process plan and an activity report.

5.12.4 Advantages and Limitations of Variant CAPP

Table 5.13: Variant CAPP: advantages and limitations

Advantages	**Limitations**
Proven plans reused; consistent, single best plan per family; simpler and quicker to implement than generative; suited to a machine shop producing a variety of small components; possibility of obtaining a process typical for a given part; retrieval by code is efficient	Requires classification and coding of all parts and formation of families; quality depends on the planner who prepared the standard plan; not suitable for new or dissimilar parts; plans still need editing and parameter selection; dependent on the family concept

5.13 Generative CAPP

Definition 5.6 (Generative process planning)

Generative process planning is a system that synthesises process information in order to create a process plan for a new component automatically. Process plans are created from information available in the manufacturing database without human intervention. Upon receiving the design model, the system generates the required operations and their sequence; the knowledge of manufacturing is captured and encoded into software, and decision logic imitates the process planner.

Advantages: (i) it generates consistent process plans rapidly; (ii) new plans can be created as easily as retrieving plans of existing components; (iii) it can be interfaced with an automated manufacturing facility to provide detailed and up to date control information.

5.13.1 Generative CAPP Architecture

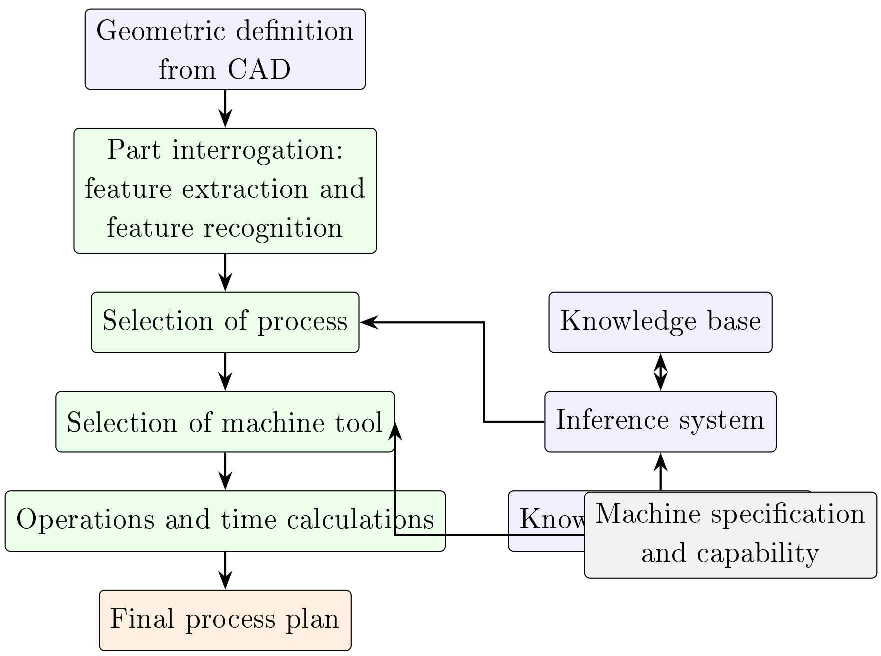


Figure 5.6: Generative process planning

The modular structure of a generative CAPP system (textbook) consists of: component representation module (new or old component; geometry file creation and code generation); feature extraction module; feature process correlation module; operation selection and sequencing module; machine tool selection module (with machine tool and material data and machinability data); standard time and cost computation module (with standard time and cost library); and report generation module (editing, printing the process sheet).

Feature Extraction and Feature Recognition

Part interrogation extracts the machinable features (holes, slots, pockets, steps and so on) from the CAD geometry, and recognition classifies them and correlates each with candidate processes. In the integrated CAD/CAPP model this is module A2 (feature recognition), followed by technical information processing (A3) that adds tolerance, roughness and hardness.

Machine, Tool and Operation Selection; Sequencing

Machine selection uses machine specification and capability (size, accuracy, power). Tool selection uses the tool database and standard tool libraries. Operations are selected by correlating features with processes (for example, hole with tolerance coarser than H10 is drilled, H8 or finer is drilled and reamed) and are sequenced using precedence relations between features and set ups.

Process Parameter Selection, Time and Cost

Parameters (speed, feed, depth of cut, number of passes) are obtained from machinability databases (for example commercial machinability data systems) and from the ATS module. Standard time and cost are computed from machining time plus loading, unloading, set up and inspection times, using the standard time and cost library.

Knowledge Base and Inference System

The *knowledge base* holds manufacturing knowledge in the form of rules, frames, decision tables or trees. The *inference system* applies the rules to the part description; knowledge acquisition populates the knowledge base. Approaches proven in generative planning: forward and backward planning; input format based planning; CAPP based on CAD models; CAPP based on decision logic (decision trees or tables); and CAPP based on artificial intelligence.

Forward and backward planning. In forward planning the initial state is the raw material and the goal is the final design; the planner modifies the raw workpiece until it takes on the final design qualities. For forward planning the successor surface must be known before a process is selected, since the postcondition of the first process becomes the precondition of the second. In backward planning the process starts with the finished component and works back to the unmachined workpiece, each machining process being treated as a filling process; it begins with the final surfaces and selects processes to satisfy the initial requirements, which removes the successor surface problem.

5.13.2 Advantages and Limitations of Generative CAPP

Table 5.14: Generative CAPP: advantages and limitations

Advantages	**Limitations**
Rapid, consistent plans; plans for new parts as easily as retrieving old ones; no dependence on similar parts; interface with automated manufacturing for detailed control information; possibility of a detailed machining process for a given production system	Needs complete, quantifiable manufacturing knowledge encoded in software; difficult to represent shape knowledge and to model machine, tool, workpiece, fixture dynamics; input from drawings is tedious to automate; process boundaries and cost models must be defined; complex to develop and maintain

5.13.3 Implementation Considerations

For successful CAPP: a good knowledge of process capability; availability of coded or feature based part input; integration with CAD, tool and machine databases; and interfaces to MRP and shop floor systems. In precision component process design, several process modifications are needed to realise the product.

Key Examination Points

- Variant: GT based, retrieval of a standard plan by family code; two stages (preparatory, production). Generative: rules and decision logic; part interrogation, feature recognition, process, machine and operation selection, time and cost; knowledge base plus inference system.
- Semi-generative lies between variant and generative.
- Opitz code: 12345 (form), 6789 (supplementary), ABCD (secondary); example part code 15100.
- Code structures: hierarchical, chain, hybrid.
- Forward planning: raw material to part; backward planning: finished part back to raw material.

5.14 Computer Integrated Manufacturing and Production Management

5.14.1 Evolution and Concepts of CIM

Computer Integrated Manufacturing is an ideal concept in which computer based manufacturing applications communicate information to coordinate design, planning and manufacturing processes. The components of CIM in the course material include: product design (CAD, CAE, IGES, PDES, DMIS, DFM), process planning (CAPP, GT, cellular manufacturing), manufacture (robotics, FMS, NC/CNC/DNC, cells and centers, automated inspection, AGV and ASRS), and systems management (JIT/Kanban, MRP, TQM, DSS/ES/AI, LAN, TOP, satellites, MAP and STEP, bar codes and EDI).

The evolution of management and manufacturing concepts adopted in the source chart is summarised below (each period adds items to the previous list).

Table 5.15: Cumulative manufacturing concepts adopted by year

Year	**Concepts (new concepts added in that period)**
1980	JIT/Kanban, TQM, TPM, cellular manufacturing, MRP II
1986	Adds CIM, empowerment, kaizen
1989	Adds technology management, system management, training, total material flow, design management, manufacturing strategy, performance measures
1991	Adds benchmarking, structure and culture, innovation strategy, corporate strategy, group technology, batch size reduction, non-financial measures
1996	Adds simultaneous engineering, business process re-engineering (BPR), vendor quality, supplier partnership

5.14.2 CIM Architecture and Models

Modern CIM architecture integrates business processes, people, computers and software through an infrastructure of enabling platforms. Traditional integration builds translations between two systems or a single data file acting as a database for all tools, but an integration architecture supporting each application's local data requirements is needed, modelled by a unified information model. Data in CIM can be grouped into life cycle related, domain related and level of abstraction related. The textbook describes these CIM models: ESPRIT CIM-OSA, NIST-AMRF hierarchical model, the Siemens model, the model of Digital Equipment Corporation and the IBM concept of CIM.

ESPRIT CIM-OSA. The European Strategic Program for Research and Development in Information Technology (ESPRIT) created an environment where multi vendor production systems can be implemented at reasonable cost; OSA means Open System Architecture. CIM-OSA was developed by AMICE, a consortium of about 30 major European vendors and users. It separates functions using the CIMOSA modelling framework (specific and generic functions clearly separated) and the CIMOSA integrating infrastructure (supporting execution of generic functions and linking specific functions, that is, the communication system connecting all functions). Activities, information and control are treated as three entities so one can change without greatly affecting another. Four modelling views: the function view (work flows), the information view (inputs and outputs of functions), the resource view (structure of resources such as humans and machines) and the organisation view. The integrating infrastructure comprises business process management, information management, front end services, system wide data exchange, common services and communication services over communication and data processing hardware, connecting humans, machines, applications and other non CIM-OSA systems.

5.14.3 Manufacturing Information Flow and Planning Levels

Information flows from customer requirements and product design (CAD) to process planning (CAPP), which produces process plans, tool and fixture requirements and NC programs; MRP II and capacity planning issue production orders; shop floor control releases, schedules and monitors orders; data collection returns actual data to MRP and process planning. MRP II plays the role of the central planning system in CIM, with major modules covering manufacturing, engineering, financial and marketing applications; ERP and supply chain management extend the scope to the enterprise and its partners.

5.14.4 Manufacturing Databases

Manufacturing databases needed in CIM include the engineering and manufacturing database (product structure, process plans), machine data bank, tool data bank and fixture data bank (used by CAPP), parts master files, part family matrix and standard plan files, machinability and standard time libraries, and shop status files. A database (hierarchical, network or relational) is managed by a database management system with a query language such as SQL, and product data management (PDM) controls product data across the organisation.

5.14.5 Flexible Manufacturing Systems

Definition 5.7 (Flexible manufacturing system)

A flexible manufacturing system (FMS) is a system that consists of numerous programmable machine tools connected by an automated material handling system and controlled by a centralised computer system. FMS is a reprogrammable manufacturing system capable of producing a variety of products automatically.

Characteristics. Programmable machine tools; control by a common computer network; combines flexibility with efficiency; reduces set up and queue times; part handling system consisting of automated guided vehicles (AGV).

Components. (1) Processing stations or workstations: machining centers, milling modules, turning modules, assembly workstations, inspection stations, sheet metal processing machines and forging stations. (2) Materials handling and storage: random independent movement of workparts between workstations, handling of a variety of workpart configurations, temporary storage, convenient access for loading and unloading and compatibility with computer control. (3) Computer control system: control of each workstation, distribution of control instructions to each workstation, production control, traffic control, shuttle control, work handling system monitoring, tool control, and system performance monitoring.

Layouts. In-line (basic in-line configuration; in-line with transfer), loop, ladder, open field and robot centered cell.

Table 5.16: Types of FMS layout configurations

Layout	Description
In-line	Machines arranged in a straight line along a material handling system; variants: basic in-line and in-line with transfer.
Loop	Workstations arranged in a loop served by a loop handling system.
Ladder	Workstations arranged on rungs between two handling rails.
Open field	Complex arrangement of workstations with several handling paths for a large part family.
Robot centered cell	Workstations arranged around a robot that handles the workparts.

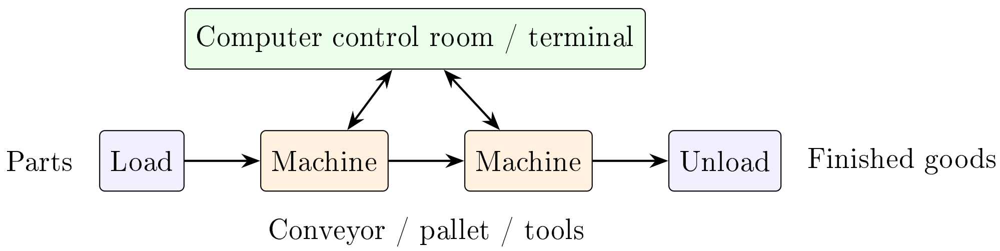


Figure 5.7: Schematic of a flexible manufacturing system

Planning, design and operation. FMS planning and design considers part family, processing requirements, physical characteristics of workparts, production volume, variations in process routings, work in process and storage capacity, pallet fixtures and tooling. Operational issues: scheduling and dispatching, machine loading, part routing, tool management, pallet and fixture allocation. Challenges: deciding whether FMS is the best production system economically and socially, expansion costs and day to day maintenance.

Benefits. Higher machine utilisation; reduced work in progress; lower manufacturing lead times; greater flexibility in production scheduling; higher labour productivity.

Manufacturing Flexibility

Different approaches to flexibility: the capability of producing different parts without major retooling; a measure of how fast the company converts its processes from making an old product line to a new one; the ability to change a production schedule, to modify a part or to handle multiple parts (operational); the ability to produce highly customised and unique products (customer); exploiting various dimensions of speed of delivery (strategic); offering a wide variety of products (capacity); the ability to rapidly increase or decrease production levels or shift capacity between products or services.

Table 5.17: Types of manufacturing flexibility

Group	**Type**	**Definition**
Basic	Machine	Ease with which a machine can process various operations
	Material handling	Ease with which different part types can be transported and positioned at the machine tools
	Operation	Ease with which alternative operation sequences can be used for a part type
System	Volume	Capability to operate profitably at different volumes of existing part types
	Expansion	Ability to build a system and expand it incrementally
	Routing	Alternative paths a part can follow through the system for a given process plan
	Process	Volume of the set of part types the system can produce without incurring any set up
	Product	Volume of the set of part types that can be manufactured with minor set up
Aggregate	Program	Ability to run for reasonably long periods without external intervention
	Production	Volume of the set of part types produced without major investment in capital equipment
	Market	Ability to adapt efficiently to changing market conditions

Examples in the source: different models designed to be assembled in the same sequence can be built in the same plant, and in a paint shop robotic applicators are programmed to cover various body styles with equal precision, minimising waste and environmental impact while maximising quality.

Key Examination Points

- FMS = programmable machines + automated handling + central computer. Components: workstations, handling and storage, computer control.
- Layouts: in-line, loop, ladder, open field, robot centered cell.
- Flexibility: 3 basic, 5 system, 3 aggregate types.
- CIM models: ESPRIT CIM-OSA (function, information, resource, organisation views), NIST-AMRF, Siemens, DEC, IBM.

5.15 Production Planning and Control

Production planning and control (PPC) determines what is to be produced, when, and with which resources. In the CAPP framework, the organisational planning system and MRP (material resource planning and capacity planning) convert requests into production orders; the process plans from CAPP, the parts master files and the machine, tool and fixture data supply routings and times. Production control receives production orders, issues NC programs and instructions to CAM and collects actual data, returning it to MRP. Order scheduling, machine loading and job sequencing are implemented within shop floor control (next sections).

5.16 Shop Floor Control

Shop floor control deals with managing work in process. It consists of the release of production orders to the factory, controlling the progress of the orders through the work stations and getting current information on the status of the orders. The input to the shop floor control system is the collection of production plans: master schedule, manufacturing capacity planning and ERP data. The factory production operations are the processes controlled. A typical shop floor control system consists of three phases, managed by software in CIM: *order release*, *order scheduling* and *order progress*.

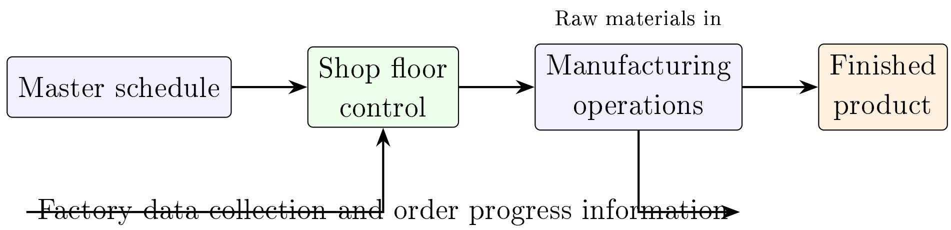


Figure 5.8: Factory information system with shop floor control

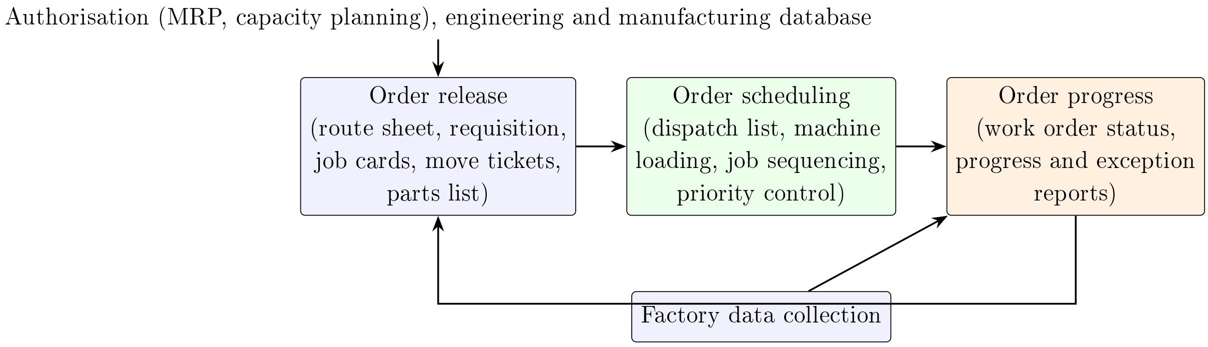


Figure 5.9: Phases in shop floor control

5.16.1 Order Release

Order release provides the documentation needed to process a production order. The shop floor order may consist of: (i) route sheet; (ii) material requisition to draw materials from stores; (iii) job cards or other means to report direct labour time given to the order; (iv) move tickets: instructions to material handling personnel to transport parts between work centres; and (v) parts list for assembly. In a manual factory these documents move with the order; in a CIM factory more automated methods track progress. Order release is connected with two inputs: authorisation through the planning functions (MRP, capacity planning) giving timing and scheduling, and the engineering and manufacturing database giving product structure and process planning information.

5.16.2 Order Scheduling

This module assigns production orders to work centres, machine tools, welding stations and moulding machines. It follows from order release and executes the dispatch function of production planning and control. It prepares a *dispatch list* indicating which order to accomplish at each work centre, with the relative priorities shown by due dates. It addresses two activities: *machine loading* (shop loading: allocating orders to work centres) and *job sequencing* (determining the order in which jobs are processed through a work centre, using priorities). Priority rules: earliest due date first; shortest processing time first; least slack time first. Fluctuations in demand, equipment breakdown, customer cancellation, and defective or late raw material affect priority; the priority control plan reviews relative priorities and adjusts the dispatch list.

5.16.3 Order Progress

The order progress module monitors the status of orders in the plant, work in process and characteristics indicating progress and performance. Reports: (i) work order status reports (current work centre, processing hours remaining, on time or behind schedule, priority level); (ii) progress reports (number of operations completed and not completed during the master schedule period); (iii) exception reports (deviations from the production schedule, for example overdue jobs). These support decisions on resource allocation, overtime authorisation and capacity issues.

5.16.4 Production Monitoring Systems

Definition 5.8 (Production monitoring system)

A production monitoring system records the performance of a production line in real time. It collects data from the various sections of the line, displays relevant data in real time dashboards, sends reports and alerts to employees and managers, and stores the data as analytics for future use.

Production monitoring data fall into two groups.

Table 5.18: Types of production monitoring

Group	Content
Status of jobs	Data of each completed operation, estimated production time, sequences; information about order flow to improve production sequences. Real time overview supports paperless reporting; planned versus actual production can be compared at any time, giving more realistic scheduling to meet delivery dates.
Status of resources	Closely connected to machinery, personnel and working environment. *Machine event monitoring* shows machine workload, downtime, availability and performance (if a machine is not working, operators know why and can rearrange planned operations). *Personnel monitoring* covers movement tracking and planned versus actual manpower; indoor positioning systems can locate people and equipment.

Benefits: easy installation and low up front cost with a modular system; compatibility with different production line computers and operating systems and integration with wireless networks; faster communication through e-mail alerts when a machine has a problem; data analysis that removes current causes of downtime and prevents future ones; easy upgrading as the line is upgraded.

5.16.5 Machine Monitoring Software Functions

The source illustrates a machine tool builder's data management software (Makino MP-max) as a central hub for production monitoring. Typical functions are:

- **Result viewer**: tracks spindle load, axis loads, feed and speed, active program, spindle tool, alarms and other operating factors.
- **Machine utilisation**: time in cycle versus out of cycle during a period, and *spindle cut time* (the time the spindle makes chips) to allow full efficiency analysis.
- **Tool data management**.

- **Work scheduling**: defines a part and the group of NC programs needed to make it; the remote monitoring system scans the programs, decides whether the machine has the tools required, and notifies the operator of missing tools.
- **Alarm analysis**: proactive maintenance is started for alarms whose frequency has increased over time.

The system must be installed and configured by a supplier with expertise in both data monitoring and machining processes.

5.16.6 Machine Utilisation and Overall Equipment Effectiveness

Machine event monitoring gives availability, performance and utilisation. As general explanatory material (the slides show OEE reports but not the formula), overall equipment effectiveness is

$$\text{OEE} = \text{Availability} \times \text{Performance} \times \text{Quality},$$

where availability is the fraction of scheduled time the machine runs, performance compares actual to ideal cycle time output, and quality is the fraction of good parts. The downtime pie chart of the source classifies losses (quality problem, control faults, scrap clearance, no operator, clean up, waiting, set up and tooling replacement), which is how the loss behind a low OEE is traced.

5.16.7 Industrial Process Control Fundamentals

In a simple control system a *process variable* (PV) is measured and compared with a *setpoint* (SP); the controller generates a *manipulated variable* (MV) output that goes to a final control element, which influences the process variable to give stable control. The controller algorithm is typically PID (proportional, integral, derivative). An industrial process control system has four basic elements: a measurement, a controller, an output signal that manipulates the process, and the process itself. The PV is the input signal to the controller, measured by field devices; the MV is the variable manipulated to control the PV (for example valve position as the MV to control flow rate as the PV).

Table 5.19: Basic control strategies

Strategy	**Description**
On-off control	The output is switched fully on or off around the setpoint.
Open loop	Control action is not a function of the process variable or load changes; not self correcting; the controller does not know the actuator output, process condition or disturbances, so it is limited to few applications.
Feed forward	A sensor detects disturbances or load changes and the controller calculates the actuator signal before the disturbance affects the process; more expensive and complicated and needs better understanding of the process.

Table 5.19: Basic control strategies

Strategy	**Description**
Closed loop	The output measurement is fed back and compared with the setpoint; the difference is the error and the controller drives the actuator until the error is minimised. Widely used, most often with PID controllers.

5.16.8 Industrial Control Systems: PLC, DCS and SCADA

Programmable logic controller (PLC). An industrial computer that can be programmed for various control applications, built from a CPU, power supply, digital and analog I/O modules and communication modules (modular or integrated types). Sensors connect to the input module and actuators to the output module; the CPU continuously reads inputs and, according to the program, produces outputs to operate the actuators.

Distributed control system (DCS). Used to control complex and geographically distributed applications; controllers are distributed across the plant, connected to field devices, and communicate continuously with operating PCs through a high speed network or bus. An engineering PC configures the controllers, and an HMI operating station monitors field parameters graphically, logs data and offers trend displays and face plates.

Supervisory control and data acquisition (SCADA). A system that collects data from instruments located in the plant or at remote sites, processes it at a central location for monitoring and control, and sends supervisory commands back to remote stations to control field devices.

Table 5.20: Comparison of PLC, DCS and SCADA

Aspect	**PLC**	**DCS**	**SCADA**
Nature	Programmable industrial controller (hardware)	Network of distributed controllers with operator stations	Supervisory data acquisition and control system (hardware and software)
Scope	A machine or cell	Complex plant, geographically distributed within a site	Wide area, local or remote sites
Main task	Reads inputs, executes program, drives outputs	Continuous control with HMI, trends, logging	Gathers data, displays it, sends supervisory commands, stores history

Table 5.20: Comparison of PLC, DCS and SCADA

Aspect	PLC	DCS	SCADA
Field connection	Sensors and actuators to I/O modules	Field devices to controllers	Through RTUs (or PLCs) to a master terminal unit

The relation to the CNC controller was given in Section 3.2.8: the CNC controller (NCK) executes G code interpolation for axis motion, whereas the PLC controls the auxiliary functions (spindle on and off, tool changer, clamping, coolant, interlocks).

Table 5.21: PLC versus CNC controller

Aspect	PLC	CNC controller
Nature	General purpose logic controller programmed by the user for any application	Purpose built controller for machine tool motion (multi axis mill, lathe)
Typical task	Sequence and process control, auxiliary and safety functions	Axis interpolation and servo motion control from G code
Programming	PLC languages (for example sequential function charts)	G code and part programs
In a machine tool	Programmable machine controller (PMC) for auxiliaries	NCK (numerical control kernel)

5.16.9 SCADA in Detail

SCADA is a computer based system for gathering and analysing real time data to monitor and control equipment that handles critical and time sensitive materials or events. First used in the 1960s, it is now an integral component of many industrial plants. Its hardware consists of the *remote terminal unit* (RTU), which is in effect a programmable logic controller collecting data from field instruments, and the *communication terminal unit* (CTU) or master terminal unit, which transfers the data to the central location where SCADA HMI software displays statuses and allows the user to send control signals to the plant.

SCADA works by connecting local sensors, devices and PLCs, which collect data points known as *tags*, to an on-site server or remote or virtual machine. Data can be saved in a *historical database* for later analysis. Users interact through operator workstations, HMIs or the SCADA server. Illustration from the source: in a bottling plant a supervisor is alerted when a filling station nozzle pressure has fallen by 20 percent (increasing cycle time by 5 seconds); a technician replaces the nozzle before the problem creates bottlenecks downstream.

Benefits: real time feedback on process performance; immediate detection and alert when output falls outside acceptable parameters; the ability to shut down or modify processes remotely; historical data to compare with real time data; fewer product errors and discards; improved productivity; maintenance benefits including accurate diagnostics and enablement of predictive maintenance; data driven insight into equipment performance. Note: the source remarks that SCADA is not as common in discrete manufacturing as in process industries such as oil and gas, utilities and wastewater, because discrete lines are not fully automated processes that a computer can control.

Wireless SCADA. Organisations replace sections of hard wired SCADA with wireless equipment to improve reliability and cost. Advantages: drastically lower installation cost and time for new sites, reduced permit costs, no trenching or conduit, less wire failure from degradation, and removal of long distance buried 4 to 20 mA analog cabling and of the analog to digital I/O converter modules used with PLCs or RTUs.

5.16.10 Automation Hierarchy: ERP, MES, SCADA and PLC

Solutions such as PLC, integrated production control systems and SCADA sit at an intermediary level between enterprise resource planning (ERP) and the computerised control system. The manufacturing execution system (MES) concept associates these solutions and includes all the production data needed to bind ERP with the control system ("e-manufacturing", "manufacturing intelligence", "transparent factory").

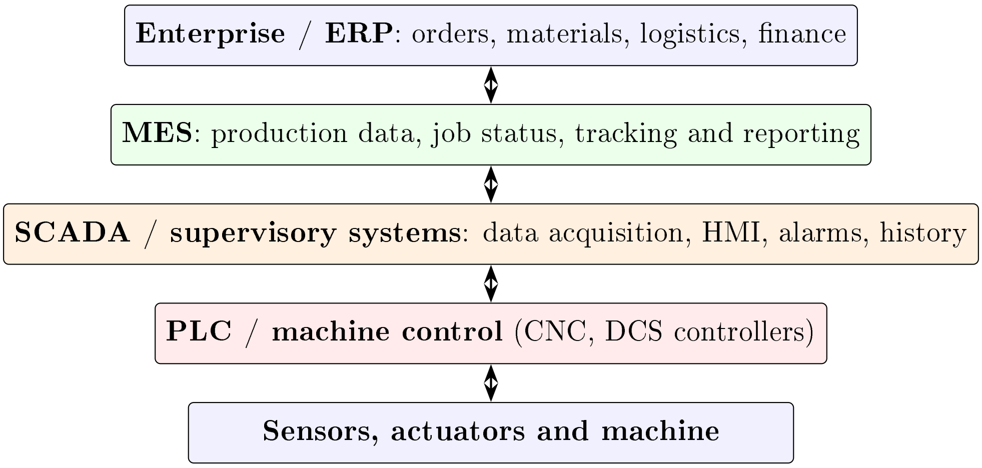


Figure 5.10: Automation hierarchy from the enterprise to the machine

Table 5.22: Roles of ERP, MES, SCADA and PLC (not interchangeable)

Level	Main role	Typical data	Time scale
ERP	Plans and manages enterprise resources: logistics, production, distribution	Orders, inventory, costs	Days to months (planning)

Table 5.22: Roles of ERP, MES, SCADA and PLC (not interchangeable)

Level	**Main role**	**Typical data**	**Time scale**
MES	Executes and tracks production on the shop floor between ERP and control; ties production data to ERP	Job status, resource status, genealogy, performance	Shifts, hours
SCADA	Supervises and acquires data from PLCs and instruments; HMI, alarms, history	Tags, alarms, trends	Seconds to minutes
PLC	Direct real time control of machine or cell	Sensor inputs, actuator outputs	Milliseconds

(The time scale column is general explanatory material; the slides give the ordering ERP, MES, control system but not the time scales.)

5.16.11 SCADA, Manufacturing Analytics and Industrial IoT

SCADA concentrates on control and on what is *not* supposed to happen; it falls short in explaining what is going right or wrong in a process. *Manufacturing analytics* focuses on productivity and inefficiency: it collects quality data as products are made and turns it into dashboards and scoreboards that show where bottlenecks occur, where downtime exceeds standards and where performance is down. The two work together: existing SCADA data are the data source for analytics, so implementation is easy.

Table 5.23: SCADA versus manufacturing analytics versus Industrial IoT

Aspect	**SCADA**	**Manufacturing analytics / IIoT**
Focus	Monitoring and control of the process	Productivity, inefficiency, granular machine data
Question answered	Is the process within limits?	Why is performance down, where is the bottleneck?
Data	Tags from PLCs, RTUs, historian	SCADA data plus other sensor data; scalability and analytics added
Role in Industry 4.0	Data source; limited as a fully connected ecosystem to the rest of the business	Implemented on top of SCADA; connects to the wider business

The IIoT is described as having been built on top of SCADA, adding scalability and data analytics, while SCADA data still act as a data source for it (see Section 6.13).

5.17 Shop Floor Data Collection

The techniques and technologies by which the status of production is collected are called shop floor data collection. Data to be collected: machine data, operator data, tooling data, data on jobs to be done, materials data, materials handling data, scheduling data, process planning data and inspection data (capture of accurate and timely process data). Variables affecting manufacturing productivity are people, machines, tooling, gauging and parts or raw materials. Shop data requirements: people (availability: at workstation, idle, absent), machines (status: idle, set up, production, delay, downtime), and parts or raw materials (location: raw stores, transit, on machine, waiting, finished stores, assembly, missing).

Table 5.24: On-line versus off-line (batch) data collection

Aspect	**On-line**	**Off-line (batch)**
Entry	Data entered directly into the computer	Collected in a storage device or stand alone computer and entered later in batch mode
Currency of status file	Always current; up to date information to production personnel	Delay in entry and processing; no real time status
Ease of implementation	Harder to install	Easier to install and implement

Manual data collection depends on cooperation and clerical accuracy of employees; errors include wrong dates, incorrect order numbers and incorrect operation numbers, and there is a time lapse between events and recording. Input techniques therefore include push button pads, keyboards with error checking routines, and automatic data collection: bar code technology, optical character recognition, magnetic ink character recognition, voice recognition, smart cards and data acquisition systems (DAS). Data collection serves to supply the order progress module, to provide up to date information to supervisors and production control, and to enable management to monitor implementation of the master schedule.

5.17.1 Smart Shop Floor and Industrial Information

Sensors send continuous parametric data from machines; machine to enterprise integration is essential for business intelligence, since one cannot improve what is not measured. Business intelligence aims at reducing machine downtime, enhancing productivity time, streamlining raw material procurement and order processing and eliminating waste, with the ultimate objective of streamlining cash flow. The shop floor dashboard shows machine ID, target, set pressure, current and pressure. Reports include downtime summary, downtime by reason, tabular uptime efficiency, production summary, OEE reports (by machine as form, graph and tabular, and by primary), operator summary, state summary, event log and track cycles reports. The down time summary chart classifies losses as quality problem, control faults, scrap clearance, no operator, clean up, waiting and set up (replace tooling). Business intelligence identifies areas of improvement, makes the most of limited investment capital, avoids inappropriate capital and material purchases, frees capacity to diversify and compete, decreases run time costs through waste elimination, shortens equipment return on investment period through increased utilisation, reduces investigation time for root cause analysis and ties production efficiency directly to fiscal reporting. (See Chapter 6 for Industry 4.0 context.)

Key Examination Points

- Three phases of shop floor control: order release, order scheduling, order progress.
- Order release documents: route sheet, material requisition, job cards, move tickets, parts list.
- Scheduling: machine loading and job sequencing; rules EDD, SPT, least slack.
- Progress reports: work order status, progress, exception.
- Data collection: on-line versus off-line; bar codes, OCR, MICR, voice, smart cards, DAS.

5.18 Computer Aided Quality Control

Quality cannot be inspected into a product; it must be built into it (TQM). Aspects: quality of design (right materials, appropriate raw material shapes, minimum number of parts, standardisation and variety reduction, reduction of material removed, economic use of materials, standard or bought out parts; analysis with FMECA and DFMA), selection of appropriate process and equipment, and training of personnel ("doing it right the first time, every time"). Traditional statistical quality control samples parts, accepts some defective parts and inspects after parts are made. In CIM, inspection is a prevention oriented process rather than appraisal oriented (accepted or rejected); computer aided inspection (CAI) is an integral part of CIM. Inspection situations: incoming materials, stage inspection, completion of processing and before shipping. Testing appraises the performance of the final product; categories are functional, fatigue or wear, overload, environmental and destructive tests.

Definition 5.9 (Computer aided quality control (CAQC))

CAQC is the automation of inspection and testing using computers and sensor technology. Its two major segments are computer aided inspection (CAI) and computer aided testing (CAT).

Objectives of CAQC. Improve product quality; increase productivity in the inspection process; increase productivity; reduce lead time; reduce wastage due to scrap and rework. Where technically possible and economically feasible, inspection is done on a 100 percent basis.

Benefits of CAQC. (i) 100 percent inspection rather than sampling; (ii) inspection integrated into manufacturing, reducing lead time; (iii) non contact sensors allow inspection while the part is in operation; (iv) on-line non contact sensors act as the feedback element of adaptive control systems, with trend analysis compensating gradual tool wear so parts are made close to nominal rather than merely within tolerance; (v) intelligent robots fitted with vision and sensors in automated test cells; (vi) the CAD/CAM database is used to develop the inspection plan. Statistical process control (SPC) uses X-bar, R and sigma charts to monitor variability, with Pareto charts and histograms; measuring instruments are linked to microprocessor based equipment.

5.18.1 Computer Aided Inspection and the Coordinate Measuring Machine

The coordinate measuring machine (CMM) is the most prominent contact inspection equipment; in CIM CMMs are CNC controlled. A typical 3D CMM has a table holding the part fixed and a movable head with a sensing probe moving along X, Y and Z. Typical accuracy is about ± 0.004 mm with resolution 0.001 mm; the accuracy of a typical CMM is quoted as $2.6 + L/300$ micrometers, where L is the measured length in mm. Major features: stationary granite measuring table; digital incremental length measuring system for each axis; air bearings; control unit (manual measurement and self teach programming, joystick); and software. CMM software features: measurement of diameter, centre distances, lengths, geometrical and form errors in prismatic components; on-line statistics for a batch; parameter programming; measurement of plane and spatial curves; data communications; digital input and output for process integration; programs for spur, helical, bevel and hypoid gears; interface to CAD software. Software can support reverse engineering: the component is digitised with a dense set of points using the CNC CMM and converted into a computer model. CMMs typically take 5 to 10 percent of the time of manual inspection. Advantages of CNC operation: shorter measuring times, higher throughput, better repeatability, economical even for small batches, simple operation and unmanned second and third shift inspection when parts are loaded automatically.

5.18.2 Non Contact Inspection

Non contact inspection (optical) covers inspection of part dimensions, surface defects and completed or semi completed parts. Advantages: no repositioning of the workpiece; faster than contact inspection; no mechanical wear of the probe; no damage to the surface from measuring pressure. Examples: laser interferometer measuring systems (calibration of CNC machines, accuracy of the order of 0.01 micrometer; also straightness, flatness, squareness, velocity, pitch and yaw), laser telemetric measuring systems (transmitter, receiver and processor electronics; the object casts a shadow in a scanning laser beam and the dimension is computed from the edges of the shadow), machine vision and optical gauging.

5.18.3 Hierarchical Quality Control

Quality assurance in modern manufacturing cannot be separated cleanly from processing: there is neither processing without quality assurance nor CAQC apart from the processing ground, even in an integrated unit such as a machining centre. The quality assurance system must be distributed to every work process, but product quality is judged on the whole finished product, so quality information must be relatively integrated. *Hierarchical quality control* places each machining process and work process under the control of a computer communication network or local area network. The lower level station collects quality data and controls the process directly; the upper level performs arbitration and diagnosis of the part's quality problems, calculates quality cost and forecasts the future quality of products. CIMS is suited to the two demands of automation (suitability for complex and variable processing, and higher reliability and stability), and CAQC sits inside it beside CAD, CAPP and CAM and linked to production planning and control (manufacturing resources planning, materials requirement planning, lot sizing and time phasing, order release, manufacturing control).

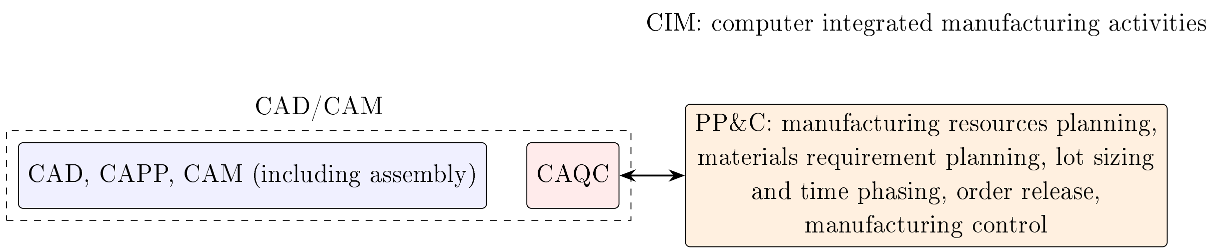


Figure 5.11: CAQC in the structure of computer integrated manufacturing activities

5.18.4 Computer Aided Inspection by Comparison with the CAD Model

Definition 5.10 (Computer aided inspection (CAI))

Computer aided inspection is a technology that compares a physical part with its 3D CAD model. It uses structured light and machine vision cameras, or a laser scanner or other 3D scanning device, to produce point cloud data of the surfaces of the object, which is analysed against the geometric model of the object.

Camera images are analysed by computer to obtain the locations of points on the object's surfaces in three dimensions. During a set up phase before inspection, the points are analysed with respect to the geometric model. Many points are eliminated to reduce data taking and analysis time and to prevent extraneous reflections from producing errors. When similar objects are inspected later, the points of each surface of interest are spatially averaged to give high accuracy dimensional measurement. The device uses several multiplexed sensors, each a camera plus a structured light source, so all sides of the object are measured in a single pass. The software gives a graphical comparison of the manufactured part with the CAD model. The source claims this is faster, more complete and more accurate than using a coordinate measuring machine or other traditional methods (a claim made in the supplied slides; Section 5.18.1 gives the CMM figures).

Computer aided comparison workflow. The CAD model and the scanned model are *aligned*, *compared* (a colour deviation map displays positive and negative deviation from the nominal surface, for example between −0.75 and +0.75 in the example shown), *analysed* (sections, feature dimensions) and *reported*. The process can be repeated automatically and can analyse results from millions of points of a point cloud; examples shown are a turbine blade and a mirror housing.

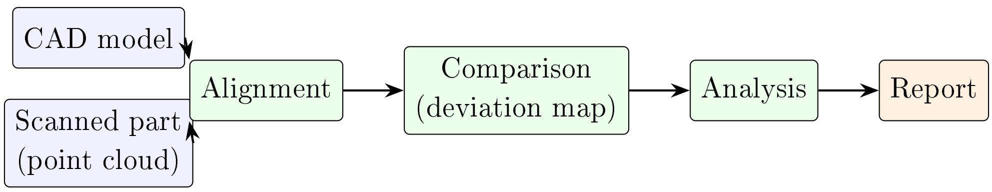


Figure 5.12: Computer aided comparison of a manufactured part with its CAD model

Table 5.25: Computer aided inspection (CAI) versus coordinate measuring machine (CMM)

Aspect	**CAI by scanning and comparison**	**CMM (contact)**
Data	Dense point cloud of the surface (structured light, laser scan)	Discrete probe touch points
Contact	Non contact	Contact probe on granite table machine
Result	Graphical comparison with the CAD model, colour deviation map	Coordinates, dimensions, form errors, statistics
Speed (source claim)	Faster, more complete; several sensors measure all sides in one pass	Typically 5 to 10 percent of manual inspection time
Typical accuracy quoted	Not given in the slides	About ± 0.004 mm, resolution 0.001 mm; $2.6 + L/300$ micrometers

5.18.5 Contact Inspection, Touch Probes and On-Machine Measurement

Inspection techniques are of two broad categories. In *contact inspection* there is physical contact between the object and the measuring or gauging instrument; in *non contact inspection* there is none. Touch probes help automate machine tools from set up to on-machine measurement after machining. Earlier probes measured basic quantities such as diameter or length but were not capable of detailed dimensional analysis; with added intelligence a shop can measure a part, compare it to a CAD model and send real time corrections to the machine CNC. This is the practical form of on-machine measurement (OMM) and of closed loop correction.

5.18.6 Optical and Non Optical Non Contact Methods

Non contact methods use a sensor located at a distance from the object. They are classified as *optical* (light is used for the measurement or gauging cycle) and *non optical* (other energy forms such as electrical fields, radiation and ultrasonics). Their advantages were listed earlier in this section (no repositioning, faster, no probe wear, no surface damage). A typical cell places a robot with a camera and a 3D scanner above a conveyor.

5.18.7 Machine Vision Inspection: Example of Optical Components

Surface defect inspection of LED lenses with uneven surfaces is difficult; human visual inspection is tedious, slow and depends on inspector experience, and judgments are erroneous because of subjectivity and eye fatigue. Machine vision is also difficult because the apparent area of a small defect can expand, shrink or disappear with uneven illumination and uneven surfaces. The source describes a wavelet packet transform based partial least squares method in three steps: (1) convert the spatial domain image to the wavelet packet transform domain and extract wavelet features of the sub band images; (2) apply partial least squares to multivariate transformation and data reduction, giving latent images that carry as much information as the original features; (3) fit the latent images with a regression model to obtain a predicted image and subtract it from the original image, leaving a residual image in which the visual defects are separated. Experiments showed a higher probability of correctly distinguishing defects from normal areas and a lower probability of falsely flagging normal areas. The principle illustrated is that computer vision improves quality and productivity of inspection.

5.18.8 CAD/CAI/CAM Integration, In Process Inspection and Closed Loop Quality Control

Integrated computer aided inspection systems tie CAD, CAM and CAI: CAD inspection data goes to CAI, inspection results return as error feedback, and in process inspection feeds CAM with process information so that process variables are corrected. Post process metrology and inspection by robots (flexible inspection systems) complete the picture.

Closed loop quality control means measurement data from in process inspection and CMM are fed back to process planning and NC programming to compensate tool wear and correct the process, ending the cycle of "manufacture then inspect". Shop floor control uses the same feedback (data collection, order progress) to correct schedules.

5.18.9 Integration of CAQC with the CAD/CAM Database

Integrating CAQC with CAD/CAM gives further benefits. Design defines the product; manufacturing uses and supplements the definition to make the plan; the quality control department should use the same CAD/CAM database, which holds the quality standards.

1. NC programs for tape controlled or computer controlled CMMs can be *generated automatically* from the database and downloaded to the CMM controller through a DNC link from the central computer; the same is possible for some non contact inspection methods.
2. Engineering changes are recorded in a common data file used by all departments including QC.
3. Computer production monitoring records help QC to trace the cause of poor quality in a particular production lot.

Key Examination Points

- CAQC = CAI + CAT; prevention oriented, 100 percent inspection.
- CMM: granite table, air bearings, probe; $2.6 + L/300$ micrometers; 5 to 10 percent of manual inspection time.
- Non contact methods: laser interferometer, laser telemetric, machine vision, optical gauging.
- CAD/CAI/CAM: CAPP plans manufacture, CAIP plans inspection; error feedback closes the loop.

5.19 Chapter Summary and Examination Points

Key Examination Points

1. Process planning: definition, ten activities, nine planner responsibilities, route sheet content.
2. CAPP versus manual planning table; benefits of CAPP (increase or decrease list).
3. Variant CAPP: GT, coding (Opitz 15100 example), families, standard plans, preparatory and production stages; generative CAPP: feature recognition, process, machine and operation selection, time and cost, knowledge base and inference system; semi-generative between the two.

4. Integrated CAD/CAPP modules A1 to A5; CAD/CAI/CAM interrelations.
5. CIM: subsystems, models (CIM-OSA), FMS components, layouts and flexibility types.
6. Shop floor control: order release, scheduling, progress; priority rules; data collection systems.
7. CAQC: objectives, CMM, non contact inspection, closed loop feedback.
8. Shop floor monitoring: status of jobs versus status of resources; PLC, DCS and SCADA compared; ERP, MES, SCADA and PLC form a hierarchy and are not interchangeable; SCADA supervises and controls, analytics explains productivity; OEE = availability $\times$ performance $\times$ quality.
9. CAI compares scan point clouds with the CAD model (align, compare, analyse, report); CAIP has global and local planning and supports on-machine measurement; hierarchical quality control uses lower level data collection and upper level diagnosis.

Chapter 6

CAD/CAPP/CAM/CNC Integration, STEP and STEP-NC, and Intelligent Manufacturing

6.1 Introduction: The Integration Problem

A major problem in computer aided systems is the transfer of product data between systems. The CAx systems include computer aided design (CAD), computer aided engineering (CAE) and computer aided manufacturing (CAM). Generally each vendor uses a proprietary data format that cannot be used directly by other systems, and these formats preclude direct data transfer between two independent systems. Design data must be converted from one format to another. A typical situation has a major company using CATIA, a major supplier using I-DEAS, small suppliers using AutoCAD and Solid Edge, and a partner using Unigraphics.

Three possible approaches exist to the data transfer problem:

1. **Eliminate** it, by using a single vendor for all product data development and analysis over the entire life cycle of the product.
2. **Translate** it, by purchasing seats for all needed systems and developing translation mechanisms for the various applications.
3. **Exchange** it, by using a common file format that each system can understand, allowing operability between systems.

6.2 Geometric Modelling Kernels and Data Translation

The *kernel* (geometric modelling kernel or solid modelling kernel) is a 3D solid modelling software component used in CAD packages. It is the code that determines how the image seen on screen is described mathematically, involving choices about how each shape is calculated and stored. When a user punches a hole through an object it is the kernel that intersects the cylindrical surface of the hole with the existing object to calculate the new edge curves; for complex shapes this needs serious mathematics. Differences between kernels make translation of CAD models a challenging problem: translation often requires approximation and reparameterisation of geometry, so a translated model is seldom exactly the same. Ways to exchange CAD data: remodel parts from scratch, use direct translators, exchange exported kernel files, or (often best) use a neutral format.

Table 6.1: Available modelling kernels

Kernel	**Notes**
Convergence Geometric Modeler	Dassault Systemes.
Romulus	Released in 1982; licensed by Siemens and HP.
Parasolid	Siemens (UGS Corp). Model creation and edit, feature modelling, advanced surfaces, direct model editing and automated regeneration. Handles only 3D solids and or surfaces, not 2D lines or arcs. File formats .X_T, .X_B. All major CAD/CAM/CAE software use Parasolid as a kernel.
ACIS	Spatial Corporation, part of Dassault Systemes; used in CAD/CAM/CAE/CMM; based on C++; wireframe, surface and solid; rich geometric operations, mass and volume calculations, 3D visualisation, advanced surfaces. File formats .SAT, .SAB. AutoCAD uses ACIS. The most significant commercial kernel; competes with Parasolid.
ShapeManager	A fork of ACIS developed by Autodesk since 2001.
Granite	Parametric Technology Corporation.
Open CASCADE	Freely available modelling kernel.

Dual kernel CAD systems. A dual kernel system has two differing kernels built into one system. IronCAD, formerly ACIS only, was twinned with Parasolid to become the first dual kernel system and uses both simultaneously; CAXA (Parasolid and ACIS) is another. The benefit is the ability to work on models developed under either kernel, but such systems are extremely difficult to build and, with numerous kernels and formats, the approach only partly solves the problem.

6.2.1 Direct Translators versus Neutral Format

With direct translators, exchange among n proprietary systems needs $n!/(n-2)! = n(n-1)$ translators. With a neutral file format and *half translators* (native to neutral, and neutral to receiving system) only $2n$ are needed.

Table 6.2: Number of translators required

Number of systems n	**Direct translators** $n!/(n-2)!$	**Half translators** $2n$
4	12	8

Table 6.2: Number of translators required

Number of systems n	**Direct translators** $n!/(n-2)!$	**Half translators** $2n$
8	56	16
12	132	24

In a typical enterprise CAD data are used throughout the organisation: product design (CAD), engineering and analysis (CAE), process planning (CAPP), production planning and control (PP&C), manufacturing (CAM) and quality (CAQ), all connected through a neutral interface and product data management.

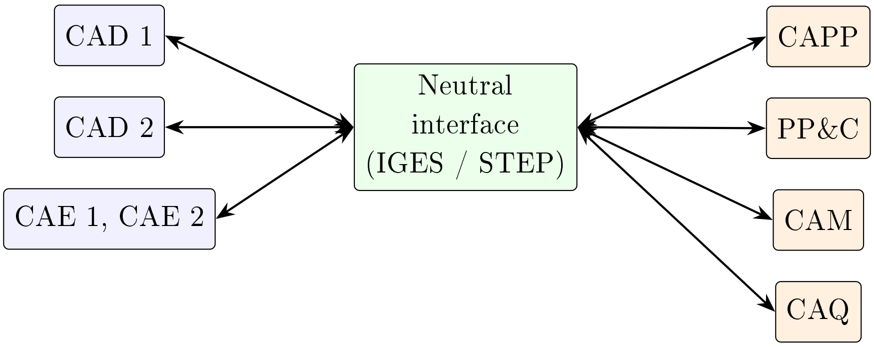


Figure 6.1: Data interchange through a neutral interface

6.3 Neutral File Formats

6.3.1 IGES

IGES (Initial Graphics Exchange Specification) was the first specification for CAD data exchange, published in 1980 as a National Bureau of Standards (NBS) document. It is supported by all important CAD vendors and remains the most widespread standard for CAD data exchange. It was originally developed for drafting data such as 2D and 3D wireframe models, text, dimensioning data and a limited class of surfaces, and has been extended in entities, syntax, clarity and consistency. Version 5.2 offers:

- **Geometry**: 2D and 3D wireframes, curves and surfaces, CSG (since version 4.0 in 1988) and B-Rep (since version 5.1 in 1991);
- **Presentation**: drafting entities for technical drawings;
- **Application dependent elements**: piping and electronic schematics, AEC elements;
- **Finite element modelling**: elements for FEM systems.

The specification defines the file format, language format and the product definition data (geometric, topological and non-geometric). An IGES file has six sections: Flag, Start, Global, Directory Entry, Parameter Data and Terminate. Each entity instance consists of a directory entry (an index with attributes describing the data) and a parameter data entry (which defines the specific entity), with bidirectional pointers between the two.

Problems with IGES. File size and processing time are practical problems; it was designed to exchange geometrical information only, so feature and other design or manufacturing information within a proprietary model is ignored; it is not an international standard; it provides no means of checking the accuracy of translators; it is continually updated, requiring compatibility between versions; its scope is limited to CAD data. In practice IGES translation is unreliable, geometry is corrupted and much cleanup is needed.

6.3.2 STEP

Definition 6.1 (STEP)

STEP (Standard for the Exchange of Product model data) is a collection of standards for representing and exchanging product model information, aiming to cover all aspects of a product (geometry, topology, tolerances, materials and so on) during its whole life, from product conception to the end of useful life. Its objective is a system independent mechanism to describe product information in computer aided systems. It incorporates geometry (B-Rep, CSG, features), process plans, manufacturing information, function information and tolerances.

STEP defines not only geometric shape but topology, features, tolerance specifications and material properties necessary to define a product completely for design, analysis, manufacture, test, inspection and product support. Most CAD vendors have implemented or are implementing STEP pre and post processors. Currently STEP translation is not yet widespread and existing translators lose information such as parametrics and constraints, features and history trees.

Table 6.3: IGES versus STEP

Aspect	IGES	STEP
Scope	Limited to CAD (mainly geometry and drafting) data	Whole product life cycle: geometry, topology, features, tolerances, materials, process plans, manufacturing
Standard status	Not an international standard (NBS 1980)	ISO 10303
Translator checking	No means to verify translator accuracy	Conformance testing defined
Versions	Continually updated; version compatibility needed	Layered architecture with application protocols
Feature information	Ignored	Included
Practical state	Widespread but unreliable, cleanup needed	Not yet widespread; translators recently available

Table 6.4: Which format for which representation

Representation	File formats
Solid models	STEP (some IGES translators)
Surface models	STEP, IGES
Wireframe models	STEP, IGES, DXF
Tessellated models	STL

6.3.3 STEP Architecture

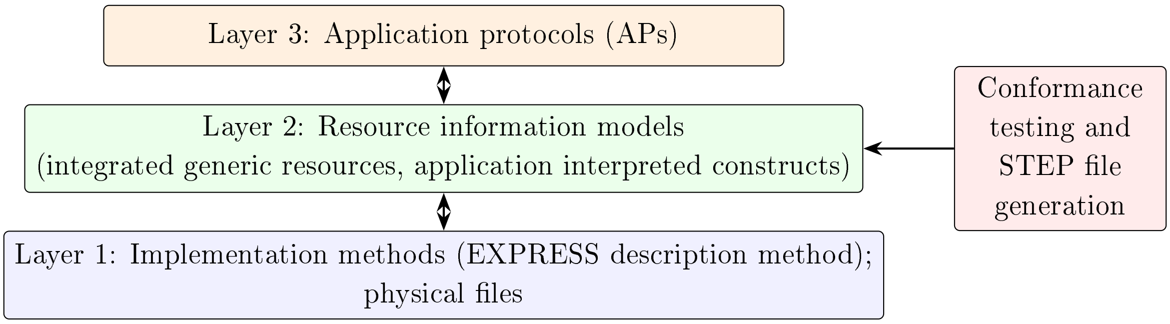


Figure 6.2: Layered architecture of STEP

The STEP documentation is organised into: description methods, implementation methods, conformance and tools, integrated generic resources, application information models, application protocols and application interpreted constructs.

6.3.4 EXPRESS and the Physical File

EXPRESS is the description language of STEP. The following schema defines a Cartesian point:

```
SCHEMA TEST_SCHEMA;
    ENTITY CARTESIAN_POINT;
        x_coordinate: REAL;
        y_coordinate: REAL;
        z_coordinate: REAL;
    END_ENTITY;
END_SCHEMA;
```

After processing by an EXPRESS compiler, a STEP physical file (ISO 10303 Part 21) can look like:

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION((''), '1');
FILE_NAME('CARTESIAN-POINT',
    '2007-07-10T09:19:11-04:00',
    (''),
```

```
    (''),
    'STEP INTERFACE',
    'STEP DESIGN SYSTEM',
    '');
FILE_SCHEMA(('TEST_SCHEMA'));
ENDSEC;
DATA;
#1=CARTESIAN_POINT(10.0,20.0,30.0);
#2=CARTESIAN_POINT(5.0,10.0,15.0);
#3=CARTESIAN_POINT(30.0,10.0,6.0);
ENDSEC;
END-ISO-10303-21;
```

The HEADER section describes the file, its name and the schema; the DATA section instantiates entities, each with a number (#1, #2, #3) and its attribute values.

6.3.5 STEP Application Protocols

Table 6.5: STEP application protocols in the course material

AP	Title
AP 201	Explicit drafting
AP 202	Associative drafting
AP 203 (ISO 10303-203)	Configuration controlled 3D designs of mechanical parts and assemblies
AP 204	Mechanical design using boundary representation
AP 205	Mechanical design using surface representation
AP 206	Mechanical design using wireframe representation
AP 207	Sheet metal dies and blocks
AP 213	Numerical control process plans for machined parts
AP 214	Core data for automotive mechanical design processes
AP 215	Ship arrangement
AP 219	Dimensional inspection information exchange
AP 223	Exchange of design and manufacturing product information for cast parts
AP 224	Mechanical product definition for process planning using machining features
AP 238	Application interpreted model for computerised numerical controllers
AP 240	Process planning for mechanical products

Key Examination Points

- Three approaches: eliminate, translate, exchange (neutral file).
- Direct translators $n(n-1)$; half translators $2n$ (4 systems: 12 versus 8).
- IGES 1980: six sections (Flag, Start, Global, Directory Entry, Parameter Data, Terminate); geometry only; version 5.2.
- STEP: entire life cycle, EXPRESS, Part 21 physical file, three layers (implementation methods, resource models, application protocols).
- AP 203, 213, 224, 238, 240 are the manufacturing relevant protocols to remember, with AP 214 for automotive.

6.4 Integration of CAD, CAPP, CAM and CNC

The conventional CAx data model interface is a chain in which each stage is joined to the next by a file format. Interface 1 (CAD to CAM) uses IGES, SET, VDA or STEP; interface 2 (CAM to CNC) uses APT or CL data, then a post processor, and the machine format ISO 6983 (RS274D). This chain carries low level information, has limited scope and needs vendor specific extensions, and information flows only in one direction (CAD to CNC). Requirement analysis for an intelligent CNC system gives a picture of the needed integrated chain:

Table 6.6: Requirement analysis for developing an intelligent CNC system

Stage	**Requirement group**	**Requirements**
CAD	Functional	Autonomy; minimised human decision and interaction; reconfigurable; quality control; adaptive control; resource management; high speed machining; self learning
CAPP	Data interface	Direct interface with CAD data; comprehensive; use of standard data; internet based support; seamless data exchange; virtual machining
CAM	Implementation	Software based CNC; open and modular architecture; user configurability
CNC	Machine control	HMI (input device, display unit); machine control unit (data processing unit, control loop unit); driving system; machine tool; feedback device with feedback data; motion data and miscellaneous function

The integration of CAD/CAPP/CAM/CNC through STEP standards replaces the file chain with a single database: STEP-NC entails a single database containing all CAD/-

CAM/CNC data that each CAx system or NC controller can access. Implementation into the CAD-CAM-CNC chain allows *bidirectional* information flow, so knowledge gained on the shop floor is fed back to the initial stages of the process, reducing production cycle time and cost. The STEP-NC file carries high level information, an entity description, an object oriented and generic structure. Interoperability within the CAD-CAM-CNC line is vital for adaptability of manufacturers using CNC machinery extensively.

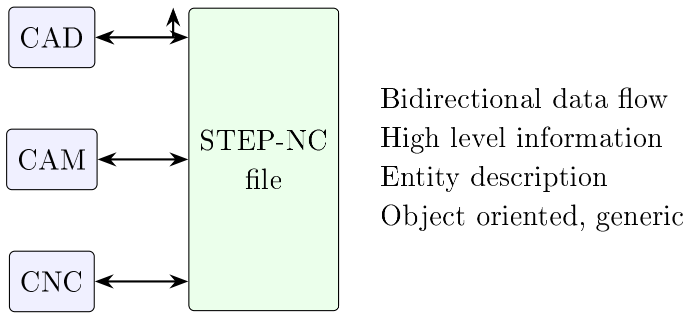


Figure 6.3: Bidirectional data flow through a STEP-NC file

Table 6.7: Conventional CNC programming (G code chain) versus STEP-NC

Aspect	**Conventional (ISO 6983 G code)**	**STEP-NC (ISO 14649)**
Information level	Low level: tool motions, G and M codes	High level: features, workplans, workingsteps
Scope	Limited; geometry of tool paths only	Complete product and process data, entity description
Vendor dependence	Vendor specific extensions; post processor for each machine	Generic, controller independent
Data flow	One way: CAD, CAM, post processor, CNC	Bidirectional; shop floor knowledge fed back
Data model	Files chained through interfaces (IGES, APT, CL)	Single database for CAD, CAM and CNC
Programming effort	Post processor and manual editing needed	Time to program a CNC about 35 percent shorter
Drawings	Many drawings sent to manufacturing	About 75 percent fewer drawings; STEP-NC self documents
Machining time	As programmed	Small to medium job orders about 50 percent faster because the controller can optimise

6.5 ISO 14649 STEP-NC in Machining

Definition 6.2 (STEP-NC)

STEP-NC (ISO 14649) is the Standard for the Exchange of Product data for Numerical Control. It allows a complete database of machining information to be built around the digital product model and ultimately makes it possible for this enhanced product model to serve as machine tool input. Part features are linked to specific *working steps*, generic descriptions of machining operations, which are roughly equivalent to machining commands formatted as traditional M and G codes.

STEP-NC (ISO 14649) provides high level data containing information such as machining features, set up, workplan and workingsteps. A machine tool control unit with an internet connection can in concept access a database of models, download a selection to memory and execute it as a part program to machine the corresponding workpiece from the appropriate stock, provided the control is STEP-NC compliant and the machine capable of the operations. This is the "hard copy" manufacturers have long wanted. Model Driven Intelligent Control of Manufacturing (the "Super Model" project, led by STEP Tools, Inc.) aims to include product geometry and the other information necessary for manufacturing in neutral format data files. Automatic feature recognition is a key enabling technology because product geometry has to be linked with STEP-NC working steps exactly as machinable features are linked with proven machining routines in knowledge based machining.

Structure of a STEP-NC program. A STEP-NC file (Part 21 physical file) consists of a project, a workplan (main workplan) containing workingsteps, each workingstep referring to a machining feature, the machining operation and the tool. The example given in the source is:

```
#1 = PROJECT('EXECUTE PIECE_TEST',#2,(#4),$,$,$);
#2 = WORKPLAN('MAIN WORKPLAN',(#10,#11,#12,#13,#14,#15),$,#8,$);
#10 = AM_WORKINGSTEP('WS ADDITIVE MANUFACTURING',#5,#17,#27,$);
#11 = MACHINING_WORKINGSTEP(
      'WS ROUGH GENERAL OUTSIDE PROFILE',#5,#19,#29,$);
#12 = MACHINING_WORKINGSTEP(
      'WS FINISH GENERAL OUTSIDE PROFILE',#5,#19,#29,$);
```

A shorter workpiece example lists `#1=WORKPIECE(...)`, `#6=MATERIAL('TiCN','TiCN')` and `#101=GENERAL_OUTSIDE_PROFILE`, with features such as Hole 1 (drilling, counterbore), Hole 2, Hole 3, Boss 1 and Pocket 1 (roughing, finishing) attached to workingsteps.

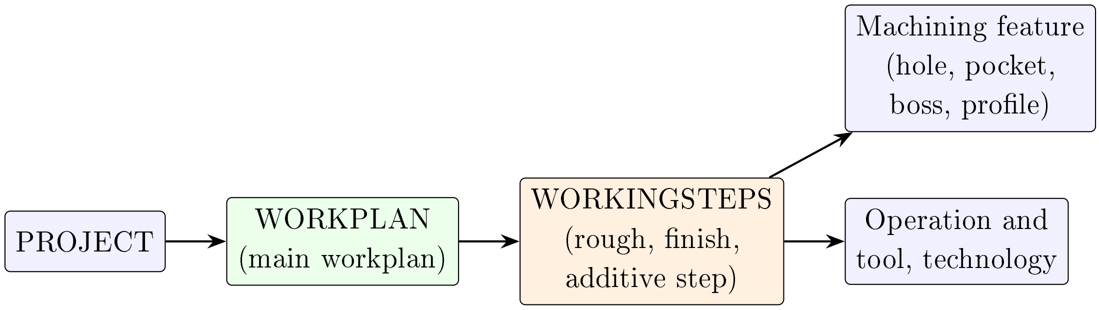


Figure 6.4: Hierarchy of STEP-NC data: project, workplan, workingsteps, features and operations

6.5.1 STEP-NC and Function Blocks

Definition 6.3 (Function block)

A function block is a software unit that encapsulates algorithms and can be designed to behave like an electronic device or circuit. The unit, designed for a specific purpose, contains the process algorithm and control state machines needed to accomplish a specific task. There are three types: basic function blocks, composite function blocks and service interface function blocks.

Function blocks (IEC 61499) are an emerging International Electrotechnical Commission standard for distributed industrial processes and control systems design. The STEP-NC controller investigates a future model of CNC controller that operates directly on STEP-NC (ISO 14649). The STEP-NC data model represents a common standard for NC programming; function blocks encapsulate machining data such as machining features and algorithms. Both demand higher level input data and share rich information for CNC. STEP-NC provides the high level data (features, set up, workplan, workingsteps); function blocks act as the tool that executes CNC machining and connects high level data with low level data, software and hardware.

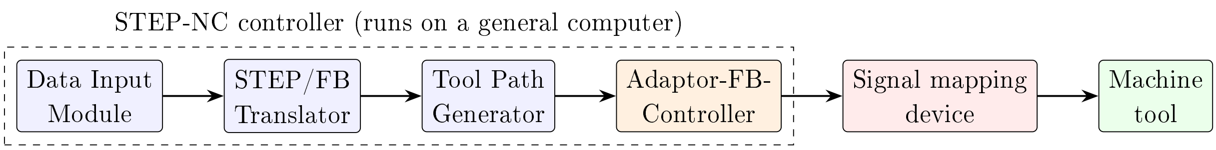


Figure 6.5: Architecture of the feature based STEP-NC controller

The controller software fits any general computer independent of controller vendors and relies on a communication standard among computer hardware, an operating system (Windows or Linux) and application software (Function Block Development Kit, FBDK, open source from Holobloc). Modules:

Data Input Module. Works with two types of data: *generic data* (machining method, machining sequence, machine tool data) and *native data* (cutting tool data, cutting conditions, tool path generation). The main source is the STEP Part 21 physical file.

STEP/FB Translator. Input: Part 21 physical file (ISO 10303-21). Four parts: data extractor, main processor, converter and FB editor. Supporting software: Java Development Kit, Java Runtime Environment and Java 3D.

Tool Path Generator. Generates the tool path (TPG file) from the workingsteps and features.

Adaptor-FB-Controller. Executes the function block application and interfaces with the machine through the signal mapping device.

The graphical user interface loads a STEP file, generates, edits and saves a TPG file (a text file) and launches the STEP-NC function block application on line. The STEP-NC FB system application in the FB (IEC 61499) editor consists of four devices: Interface3D, Machining Data, STEP-NC Controller and Signal Mapping. Its functions: develop visual information, model machine behaviour, operate machine parameters and execution, and send control signals. The prototype was demonstrated by machining workpieces through the STEP-NC controller with 3D simulation and machining data editor boxes.

6.5.2 Intelligent CAM and CNC Environment with STEP-NC

In the intelligent CAM and CNC environment the CAD/CAM environment (CAD software, CAD reconstruction module, CAM model with feature recognition and process data) writes a STEP-NC file; the file is also linked to a cloud manufacturing knowledge database and to industrial machines equipped with advanced CNC. In the CNC environment STEP-NC reading feeds visualisation and tool path generation, connected to simulation and optimisations; the physical process (for example additive manufacturing) is followed by inspection, whose feedback returns to the optimisation.

Key Examination Points

- STEP-NC (ISO 14649): high level, feature based, workplan and workingsteps, bidirectional, single database.
- Claimed gains: programming time 35 percent shorter, drawings 75 percent fewer, machining of small and medium orders 50 percent faster.
- Function blocks (IEC 61499): basic, composite, service interface; STEP-NC controller modules: Data Input Module, STEP/FB Translator, Tool Path Generator, Adaptor-FB-Controller.
- Translator parts: data extractor, main processor, converter, FB editor.

6.6 Artificial Intelligence and Machine Learning in CNC Machining

Definition 6.4 (AI machining)

AI machining refers to integrating artificial intelligence technologies with traditional machining processes to enable enhanced precision, efficiency and automation. By using AI algorithms machining operations become smarter: they predict maintenance, optimise tasks and make real time adjustments, thereby improving productivity, reducing waste and lowering costs.

AI enhances precision, efficiency and automation; algorithms optimise cutting paths, predict machine maintenance and enable real time adjustments, giving reduced waste, faster production and lower costs, and smarter, adaptable machining solutions for complex, custom projects. Artificial intelligence encompasses machine learning, deep learning, natural language processing (NLP), visual recognition (vision), speech, expert systems, robotics, predictive analytics and big data.

6.6.1 Machine Learning and Big Data in CNC

CNC machines generate expansive datasets: sensor measurements, operational parameters, metrology data and machine status information. Machine learning models applied to these data refine machining for higher throughput, superior quality and reduced cost. Examples:

- Analysing patterns of tool wear and sensor readings gives optimal tool change schedules, reducing downtime (AI based tool wear prediction).
- Feed rate and spindle speed are intelligently adjusted to optimise material removal rate while maintaining precision (adaptive machining).
- Big data analytics on production data identifies correlations between production variables and defects, predicting quality issues before they occur (AI based quality control).
- Early detection of potential machine failures is possible by multivariate analysis, paving the way for predictive maintenance.
- Deep learning extracts insights from unstructured data (customer orders, drawings, production logs) to support demand forecasting and production planning.

6.6.2 Benefits of AI in CNC Machining

Table 6.8: Benefits of AI in CNC machining

Benefit	**Explanation**
Increased efficiency and productivity	Continuous analysis of large datasets (historical production runs, material characteristics, physics simulations) fine tunes tool paths, feeds, speeds and other parameters for every new part program; machines perform at peak efficiency, shortening cycle times and minimising material waste. Tedious tasks such as tool path creation and post processing are automated, freeing skilled programmers for strategic tasks.
Improved quality control	AI powered sensors, computer vision and imaging inspect parts in real time during machining; if deviations from tolerance or surface defects are found the system adjusts or compensates the process, minimising scrap. Defect data analysis gives root causes for preventive and corrective action.
Enhanced flexibility and adaptability	Machine learning lets machines learn and adjust programs on the fly to new designs, materials, tooling or requirements with minimal changeover downtime, and supports more autonomous operation.
Reduced operating costs	Less need for less skilled shop floor labour to operate machines; predictive maintenance forecasts repair and replacement needs well before they become urgent, reducing downtime and prolonging equipment life.

6.6.3 AI in the CNC Machining Program (G Code and M Code)

AI algorithms analyse 3D models, material specifications and historical production data to simulate tool paths and determine optimal machining parameters such as feed rates and spindle speeds, which are components of the G code and M code. Cycle times and material waste fall. Machine learning refines the process by evaluating outcomes and continuously improving programming accuracy and efficiency. During operation AI adjusts G code and M code instructions in real time to maintain quality and precision, and AI powered sensors and computer vision inspect machined parts and flag defects. The result is data driven programming and self optimising machining processes.

6.6.4 AI Optimisation of Management Processes in CNC Machine Shops

AI collects and analyses data from CNC operations (machine performance, production rates, maintenance schedules) to support decisions on resource allocation and scheduling that maximise utilisation and minimise downtime. Predictive analytics foresees machine failures and maintenance needs. AI driven systems automate inventory management by tracking material and component usage in real time, avoiding overstocking and stock outs. In quality control AI identifies patterns or anomalies that indicate quality issues, reducing rework and scrap. In strategic planning AI evaluates market trends, customer feedback and production capabilities to identify opportunities for new products or process improvements.

6.6.5 Machine Learning Paradigms Used in Manufacturing

Table 6.9: Supervised, unsupervised and reinforcement learning

Paradigm	**Principle**	**Data needed**	**Manufacturing use (examples)**
Supervised learning	Uses labelled instances to predict future events. A known training data set is evaluated and the algorithm infers a function to predict outputs; it compares its output with the correct output, finds errors and corrects the model. After adequate training it can give targets for any new input	Labelled (input, correct output) pairs	Classify defect versus good part from images; predict tool wear or surface roughness from sensor signals; fault classification
Unsupervised learning	Used when training data are neither classified nor labelled; infers a function that describes hidden structure. It does not find the right output but explores the data and draws inferences	Unlabelled data	Cluster machine operating states, anomaly detection in sensor data, discovering hidden structure in production logs

Table 6.9: Supervised, unsupervised and reinforcement learning

Paradigm	Principle	Data needed	Manufacturing use (examples)
Reinforcement learning	Learns by interacting with its environment through actions and finding errors or rewards; characterised by trial and error search and delayed reward; agents determine the best behaviour automatically to maximise performance	Reward signal from the environment (no labelled answers)	Control and scheduling policies, adaptive setting of process parameters, robot control (general examples)

Machine learning permits massive amounts of data to be analysed, usually faster and more precisely, but may need additional time and resources for proper training. (The manufacturing use column is explanatory; the source gives the definitions only.)

6.6.6 Deep Learning versus Traditional Machine Learning

As manufacturing processes and data grew more complex, traditional machine learning, which relies on subjective, high quality feature engineering, proved inherently limited. Modern data such as images for surface defect detection and sequences underlying machine degradation need progressive, hierarchical representation that decomposes complexity into smaller pieces. The fourth industrial revolution therefore moved machine learning toward *deep learning*, which uses advanced computing infrastructure such as cloud computing and a cascade of network layers corresponding to different levels of abstraction of machine condition related patterns, improving defect diagnosis and RUL estimation.

6.6.7 Hierarchical View of Manufacturing Systems and AI Based Optimisation

The source uses a hierarchical, three level decomposition of the manufacturing system: system, process and material, with parallels to the ISA-95 framework. The low level is the physical processes (material property modelling matters); the mid level is sensing and monitoring of the processes (the foundation of maintenance and prognostics and health management, PHM); the top level is managing manufacturing operation (system level modelling and performance analysis). The hierarchy is valuable when judging whether an AI technology suits the system requirements, to ensure global goals are met and sub-optimisation is avoided.

A manufacturing system here comprises machines, robots, conveyors and supporting activities such as maintenance and material handling. Factory operations are highly nonlinear and stochastic because of countless uncertainties and interdependencies, and system performance, hence global competitiveness, depends critically on optimal control of material flow through the work cells; AI is applied to this optimisation problem.

6.6.8 AI Case Studies in Manufacturing

Table 6.10: Industrial examples of AI in manufacturing

Company	**Application and result**
General Motors (Dreamcatcher)	Machine learning based generative design of a seatbelt bracket: 150 designs, including a single piece replacing an eight piece part that was 40 percent lighter and 20 percent stronger; an engineer still decides which design suits the company's needs and manufacturing capabilities.
Continental	AI based virtual simulation that can generate 5,000 miles of vehicle test data per hour, against more than 20 days for physical testing; supports ADAS testing.
Audi	Machine learning with smart cameras and image recognition to find tiny cracks in sheet metal parts automatically in seconds, in line without slowing production. It replaces optical crack detection that needs reconfiguration for every new component and gives false detections because simple algorithms depend on lighting and surface properties. Sample images are marked to pixel level.

Drivers for adopting AI listed in the source: high revenue volatility, need for continual cost savings, short production times, increased regulation and inspection, need for learning and adaptability on the factory floor, capacity and supply chain demands, and increased need for small batch or customised goods. Factories of the future can detect defects throughout production, deploy predictive maintenance, respond to real time demand changes, validate intricate goods, cut the cost of single run goods and shift mundane tasks to machines.

Key Examination Points

- Four benefits: efficiency and productivity, quality control, flexibility and adaptability, reduced operating costs.
- ML uses tool wear patterns (tool change schedules), feed and spindle speed optimisation, defect correlation (quality prediction), multivariate analysis (predictive maintenance), deep learning on unstructured data (forecasting).
- AI in G and M code: simulate paths, choose feeds and speeds, adjust in real time, vision inspection.
- ML paradigms: supervised (labelled data), unsupervised (hidden structure),

reinforcement (trial and error, delayed reward); deep learning replaces manual feature engineering.

6.7 Intelligent Fully Autonomous CNC Machine Tools

6.7.1 Levels of Intelligence: Conventional versus Intelligent CNC

The source describes the flow computer, CNC controller, actuator, machine tool and cutting process, with four feedback levels. Current CNC machine tools, with or without adaptive control, use the inner loops; future intelligent CNC machine tools with adaptive and process control also use the outer loops that include knowledge, know-how and skill (key issues for the future).

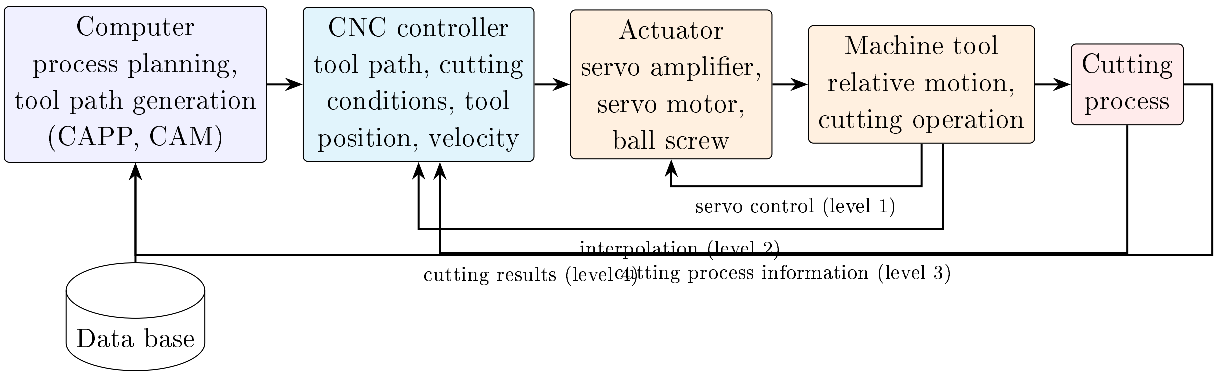


Figure 6.6: Intelligent CNC machine tool: four levels of feedback

Table 6.11: Feedback levels of an intelligent CNC machine tool

Level	Meaning
Level 1: servo control	Position and velocity feedback between machine tool and actuator (servo amplifier, servo motor, ball screw).
Level 2: interpolation	Interpolation feedback to the CNC controller (tool position, tool velocity).
Level 3: cutting process information	Cutting force, temperature, deformation, vibration and noise measured from the cutting process and fed back.
Level 4: cutting results	Machining accuracy, surface roughness and tool condition fed back to the computer (process planning and tool path generation).

Disturbances noted: deformation, vibration and noise at the machine tool; temperature and vibration at the cutting process. Knowledge, know-how and skill stored in the database are the key issues for future intelligent CNC machine tools.

6.7.2 Architecture of an Intelligent CNC Machine

The hardware architecture of the intelligent CNC in the source: a PC (Windows NT with real time support) handles man machine interface, communication and CAD/CAM functions; several DSP boards run functions. *DSP board 1 (motion control module)* performs NC code decoding; interpolation (linear, circular, spline); axis control functions (PID, PPC, CCC, ZPETC); sensor functions (velocity, torque, position, force); machine tool operating functions (feed, speed, offsets). *DSP board n (intelligent machining module)* performs sensor data collection, filtering, FFT and FRF, adaptive control, tool wear monitoring, tool breakage detection, chatter avoidance, thermal deformation compensation, probing and manipulation of machine tool operating functions. Motor power, velocity feedback and position feedback pass through a servo amplifier and I/O box to the machine; a tacho generator and encoder are mounted on the motor.

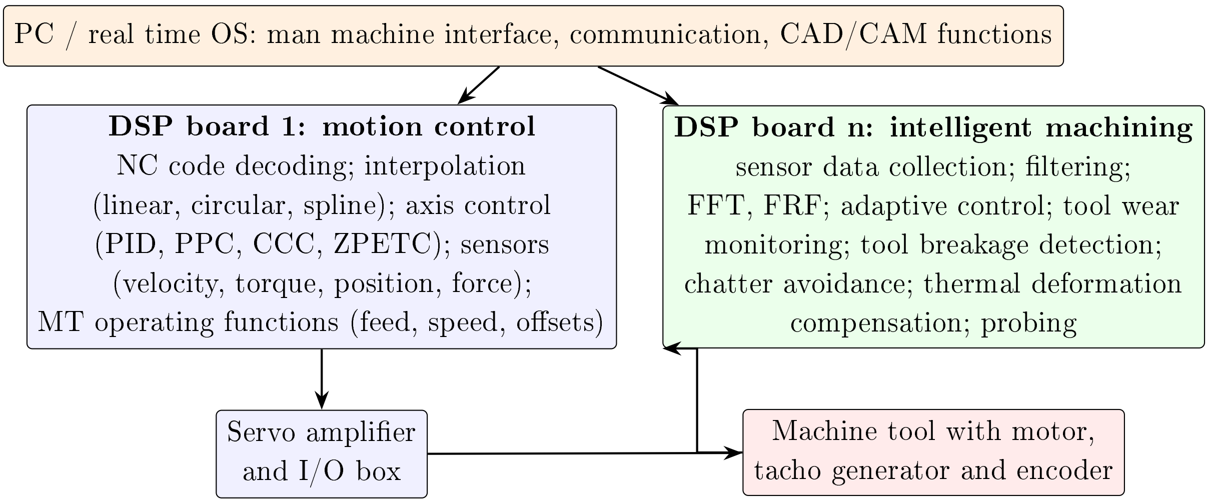


Figure 6.7: Hardware architecture of an intelligent CNC machine

6.7.3 Management, Prediction, Strategy and Observation Framework

A second view of the intelligent machine divides the system into design, management, prediction, strategy and observation around real machining.

Table 6.12: Modules of the intelligent machining framework

Module	**Content**
Design	Workpiece model and CAD data (raw material); resource and machining data supplied by the user.
Management	Planning (process, tool list, machining sequence such as face T1, pocket T2 and T3, hole T5 to T7, step T2 and T4, operations OP 1 and OP 2) and database (resource: machine tool data and tool data; machining know-how: machining feature and cutting condition), with database generation and database maintenance.

Table 6.12: Modules of the intelligent machining framework

Module	Content
Prediction	Real time process simulation of the machining process, cutting force, machining error, chatter vibration, cutting temperature and tool wear.
Strategy	Cutting condition maintenance (depth of cut, step over, feed rate, spindle speed) and tool path generation (CL data).
Observation	Monitoring (feed rate, spindle speed, spindle load, cutting force, vibration, temperature, tool wear) and process diagnosis (machining status, machining trouble such as chatter vibration and tool breakage).
Real machining	Machine tool executing the CL data, whose monitored signals return to observation.

Data flow: design output goes to planning; the database supplies real time process simulation; the prediction feeds cutting condition maintenance and planning; tool path generation sends CL data to real machining; monitoring signals feed process diagnosis and the database maintenance, closing the loop.

6.7.4 Intelligent Autonomous CNC Concepts

The intelligent CNC combines the CAD/CAM, simulation, process, machine, geometry and surface chain with intelligent components (controller and monitoring) that exchange information with simulation and CAD/CAM. The process and machine determine geometry and surface quality, which are fed back to the intelligent components. Requirements for a fully autonomous machine: autonomy; minimised human decision and interaction; reconfigurability; quality control; adaptive control; resource management; high speed machining; self learning; direct interface with CAD data; standard data; internet supported seamless data exchange; virtual machining; software based CNC with open modular architecture and user configurability.

6.7.5 From IIoT and CPS to the Intelligent Machine

An intelligent CNC is the end point of a chain of technologies. Sensors and machines (IoT) are given IP connectivity and embedded computation so that each becomes a cyber-physical entity (CPS); the data streams are stored and analysed (analytics, big data); AI and machine learning turn the analysis into diagnosis, prediction and decisions; and these are fed back into the controller as adaptive control, tool wear compensation and self optimising machining (the intelligent machine).

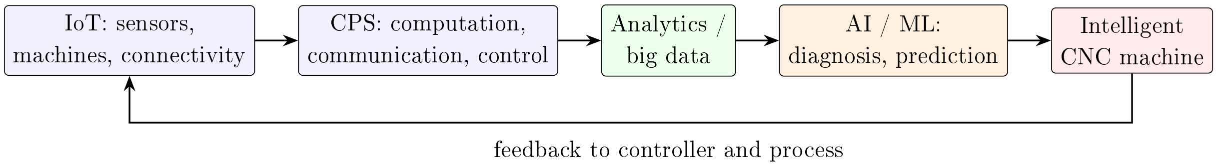


Figure 6.8: IoT to CPS to analytics to AI to intelligent machine

Key Examination Points

- Intelligent CNC adds outer loops (levels 3 and 4: cutting process information and cutting results) to conventional servo and interpolation loops.
- Two DSP boards: motion control module and intelligent machining module (adaptive control, tool wear monitoring, breakage detection, chatter avoidance, thermal compensation, probing).
- Framework: design, management (planning and database), prediction (real time simulation), strategy (cutting condition maintenance, tool path generation), observation (monitoring, process diagnosis).

6.8 Real-Time Machine Monitoring

Real time machine monitoring uses sensors that continuously send parametric data from machines. Monitored signals in the intelligent CNC: feed rate, spindle speed, spindle load, cutting force, vibration, temperature and tool wear; process diagnosis interprets machining status and machining trouble (chatter vibration, tool breakage). On the shop floor, machines (press, weld arm, cutting, CNC) connect through a gateway to a database and enterprise, with a shop floor dashboard (machine ID, target, set pressure, current, pressure) and mobile access; reports include downtime summary, downtime by reason, tabular uptime efficiency, production summary and OEE reports. The point is that one cannot improve what one does not measure; sensors convert conventional non intelligent machines into sources of data that integrate the smart machine with the enterprise (SCM, ERP with logistics, production and distribution, CRM with marketing, sales and customer service).

Table 6.13: Uses of real time monitoring data

Use	**Detail**
Tool condition	Tool wear monitoring, tool breakage detection, tool change scheduling.
Process stability	Chatter detection and avoidance, cutting force and vibration monitoring, thermal deformation compensation.
Quality	Real time inspection, tolerance deviation compensation, defect flagging.
Machine health	Predictive maintenance from multivariate sensor analysis.
Management	Downtime by reason, OEE (quality problem, control faults, scrap clearance, no operator, clean up, waiting, set up/tool replacement), event logs.

6.8.1 Intelligent Manufacturing Systems: From Sensors to Maintenance Decision

Definition 6.5 (Intelligent manufacturing system)

An intelligent manufacturing system applies AI through a system characterised by autonomous intelligent sensing, interconnection, collaboration, learning, analysis, cognition, decision making, control and execution of human, machine, material, environment and information across the whole system and life cycle.

The history of manufacturing is closely tied to improving the reliability of machine equipment to reduce unexpected downtime. Research on machine condition monitoring, defect root cause diagnosis and remaining useful life (RUL) prognosis establishes the core competence of maintenance. Machine learning learns data patterns inductively and relates them to processing condition and performance, on the principle that data reflect the physical machine characteristics.

Table 6.14: Monitoring, diagnosis, prognosis and maintenance decision

Stage	**Question answered**	**Typical output**	**Typical methods**
Monitoring	What is the machine doing now?	Signals, alarms, condition status (spindle load, vibration, temperature, tool wear)	Sensors, data acquisition, thresholds, dashboards
Diagnosis	What is wrong and why (root cause)?	Fault type and cause; performance assessment of degradation level	Classification, pattern recognition, ML and deep learning
Prognosis	How will it degrade, and when will it fail?	Predicted degradation trend and remaining useful life (RUL)	Data driven degradation models, health index, one step ahead prediction
Maintenance decision	What should be done and when?	Recommended maintenance strategy and timing	Decision support system, optimisation (computational intelligence)

6.8.2 Framework of Fault Diagnosis and Prognosis in Machine Centres

The source presents a framework of five modules for fault diagnosis and prognosis in machine centres.

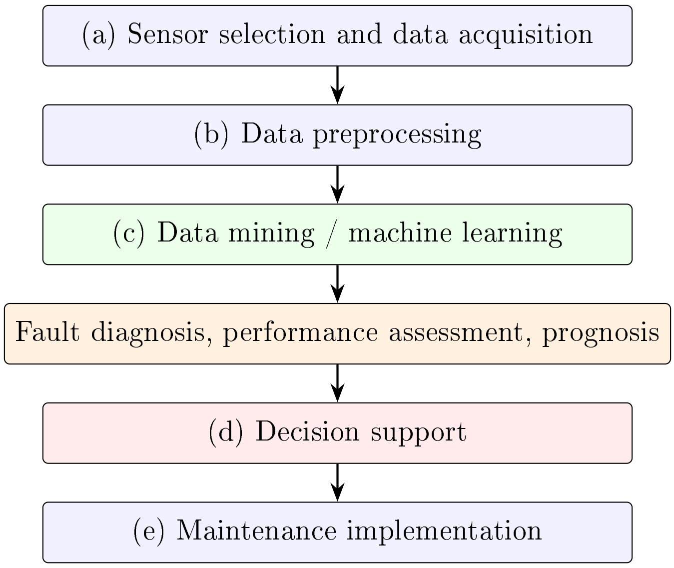


Figure 6.9: Framework of fault diagnosis and prognosis for machine centres

Sensor selection and data acquisition. Choose sensors that capture the condition of the machine and acquire the signals from the plant floor assets.

Data preprocessing. Clean and prepare the data before analysis (general description; the source names the module without detail).

Data mining and computational intelligence. Data mining is applied to discover failure information; computational intelligence is applied to optimise the solutions.

Decision support. Uses fault diagnosis, performance assessment of the degrading level and fault prognosis models to formulate the recommended maintenance strategy from the prediction or evaluation result.

Maintenance implementation. Executes the maintenance decision to prevent occurrence and development of failures, ensure equipment and personnel safety and reduce economic loss.

The system monitors plant floor assets, links production and maintenance operations systems, collects feedback from a remote customer site, integrates into upper level enterprise applications, discovers hidden information about impending failures and generates maintenance knowledge. It aims at near zero breakdown performance. After fault prediction, results are recorded and published on an online dashboard for remote condition monitoring and information sharing, and all data and analysis results are stored in an online data warehouse for exchange and transport.

6.8.3 Prognosis, Machine Health Index and Remaining Useful Life

Definition 6.6 (Prognosis and remaining useful life)

Prognosis predicts the temporal progression of machine performance degradation from its current state to final functional failure. The *remaining useful life* (RUL) is the time remaining until the machine (or component) reaches functional failure; reliable RUL estimation contributes to timely maintenance.

AI based prognosis is part of the data driven approach: it establishes a machine performance evolution model to predict future performance from current and past status. To estimate RUL, a one step ahead prediction is carried out iteratively until the predicted value passes a *failure threshold*; the number of steps multiplied by the step interval gives the RUL. If the machine performance is difficult to measure directly, an artificial *health index* (HI) is created from the sensor data to represent performance; HI based prognosis then tracks the HI trend towards the threshold. As explanatory material (general), for a monotonic health index $HI(t)$ with failure threshold HI_f, the estimate is $\mathrm{RUL}(t) = \hat{t}_f - t$ where $\hat{t}_f$ is the predicted time at which HI reaches HI_f.

6.8.4 Link to Monitoring Infrastructure

The sensors, PLCs and SCADA historians of Section 5.16 supply the data for this framework: SCADA data remain a data source for industrial IoT analytics, and rising alarm frequency is an early indicator that starts proactive maintenance. Together with tool wear monitoring, breakage detection and chatter avoidance of the intelligent machining module (Section 6.7) this completes the path from monitoring to maintenance decision.

Key Examination Points

- Monitoring, diagnosis, prognosis and maintenance decision are successive stages; RUL is the time to reach a failure threshold; a health index represents performance that cannot be measured directly.
- Five module framework: sensor selection and data acquisition, data preprocessing, data mining, decision support, maintenance implementation.
- Shop floor sources of the monitoring data: PLC, SCADA and MES (Section 5.16).

6.9 Real-Time CAM Simulation for Digital Manufacturing

The real time process simulation module predicts machining process, cutting force, machining error, chatter vibration, cutting temperature and tool wear, and provides these predictions to the strategy module to maintain cutting conditions (depth of cut, step over, feed rate, spindle speed) and update tool path generation. Simulation in machining is used before real machining to verify the geometry of the finished part and to predict physical factors so that cutting parameters can be optimised. Virtual CNC software (VERICUT, Predator Virtual CNC, DELMIA Virtual CNC, NC Simul) simulates machine, fixtures, stock and controller behaviour so that crashes, broken tools and scrap are avoided. In the digital manufacturing sense, real time CAM simulation is coupled with monitoring data so that the simulated model tracks the actual machining and tolerances are checked continuously.

6.10 Digital Twins in Manufacturing

Definition 6.7 (Digital twin in manufacturing)

A digital twin in manufacturing is a virtual model of a factory, production line, machine or process that mirrors its real world counterpart. It integrates IoT sensors, AI, simulation and real time analytics to monitor, simulate and optimise manufacturing operations.

6.10.1 Implementation of a Digital Twin

1. **Data collection**: machines, robots and production lines send live data via IoT sensors.
2. **Digital model creation**: a virtual replica of the system is built (CAD, PLM, simulation).
3. **Simulation and prediction**: AI/ML algorithms test scenarios such as machine breakdowns, design variations or workflow changes.
4. **Optimisation**: insights are used to reduce downtime, optimise scheduling and improve quality.
5. **Feedback loop**: improvements are fed back into the physical system.

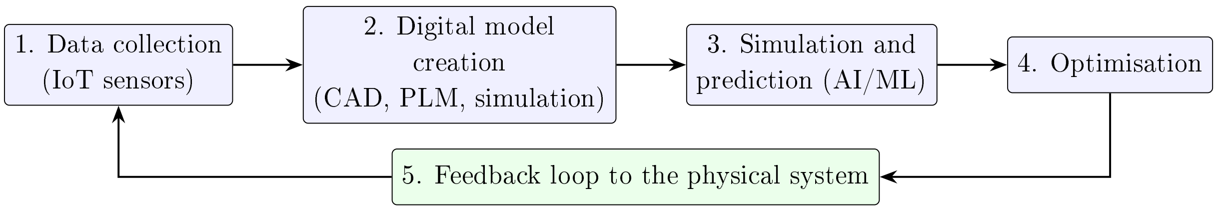


Figure 6.10: Digital twin implementation cycle

6.10.2 Applications in Manufacturing

Table 6.15: Applications of digital twins

Application	**Description**
Product design and development	Virtual prototyping reduces cost and time before physical manufacturing; example: testing material performance digitally.
Production process optimisation	Simulates assembly lines to minimise bottlenecks; dynamic scheduling and resource allocation.
Predictive maintenance	Predicts machine failures using sensor data; minimises downtime and maintenance cost.
Quality control	Real time monitoring to detect defects early; simulates how changes in input parameters affect product quality.
Supply chain management	Models logistics, inventory and demand variations; enhances resilience and flexibility.
Worker safety and training	VR based digital twins train operators in simulated environments.

Benefits. Reduced costs (predictive maintenance, fewer physical prototypes); increased efficiency (optimised workflow and production); improved product quality (real time monitoring); faster time to market (virtual testing shortens the design cycle); sustainability (resource optimisation, energy efficiency).

6.10.3 Digital Twin in Bottleneck Resolution

1. **Data collection and visualisation**: IoT sensors collect real time data (machine cycle times, work in progress, downtime, operator performance); the twin creates a virtual model of the entire assembly line, making bottlenecks visible.

2. **Bottleneck identification**: the model highlights stations with higher cycle times than upstream or downstream processes, detects idle times (machines waiting for parts or operators) and uses heat maps or flow simulations to show congestion points.

3. **Scenario simulation**: the twin allows "what if" analysis without disturbing real production. Example simulations: adding an extra workstation, reallocating workers, adjusting buffer sizes, rescheduling operations.

4. **Optimisation and predictive insights**: AI/ML algorithms predict future bottlenecks from demand or workload fluctuations and suggest optimal resource allocation and layout changes.

5. **Feedback and continuous improvement**: once solutions are tested virtually, the best option is implemented in the real factory; continuous data flow keeps improving the model.

Example 6.1 (Bottleneck at Station 3)

If Station 3 in an assembly line takes 40 seconds per unit while others take 20 to 25 seconds, the digital twin flags Station 3 as a bottleneck; simulates whether adding an extra operator, rebalancing tasks or upgrading machinery reduces the bottleneck; and recommends the most cost effective solution before changes are made physically. Since the line output is governed by the slowest station, the cycle time of the line is 40 seconds per unit until Station 3 is improved.

6.10.4 Digital Twins for Predictive Maintenance, Quality Control and Production Optimisation

For predictive maintenance the twin combines sensor data from spindle, axes and drives with failure models to predict failures. For quality control it monitors in real time and simulates the effect of input parameter changes on quality. For production optimisation it runs dynamic scheduling and resource allocation scenarios and the bottleneck workflow above.

Key Examination Points

- Five steps of digital twin implementation: data collection, digital model creation, simulation and prediction, optimisation, feedback loop.
- Six applications; five benefits.
- Bottleneck resolution: collection and visualisation, identification, scenario simulation (what if), optimisation and predictive insights, feedback.
- Station 3 example: 40 s versus 20 to 25 s.

6.10.5 Digital Twins as the Integration Layer for Prediction and Prognosis

The digital twin closes the loop between the sensing, prediction and maintenance stages: sensor data update the virtual model, the model is used to predict degradation and test maintenance or scheduling scenarios, and the best option is fed back to the physical system. The prognosis framework of Section 6.7 and the real time process simulation of the intelligent machine are therefore two views of the same idea, with the twin acting as the shared model. Blockchain based twins (traceable, tamper proof records) are noted as an emerging extension in Section 6.13.

6.11 Virtual Manufacturing and Virtual Reality

Definition 6.8 (Virtual reality)

Virtual reality (VR) is a computer generated environment with and within which people can interact. It provides an environment generally unavailable because of cost, safety or perception restrictions. VR ranges from text oriented on-line forums and multiplayer games to complex simulations that combine audio, video, animation or 3D graphics; 3D glasses, helmets and sensors create more realistic effects. Applications: simulators, manufacturing, automobile, entertainment, gaming, medical field, science and technology, space research.

Definition 6.9 (Virtual manufacturing (VM))

Virtual manufacturing is manufacturing in the computer. It is "an integrated, synthetic manufacturing environment exercised to enhance all levels of decision and control", or "an aggregation of computer based information that provides a representation of the properties and behaviours of an actualised product". It is a computer based model representing the behaviour of the real manufacturing system. A VM system does not produce any physical output, but information about it. Virtual reality is a tool that offers visualisation for VM.

The words of the definition: *environment* supports construction and provides tools, models, equipment, methodologies and organisational principles; *exercising* means constructing and executing manufacturing simulations using real and simulated objects, activities and processes; *enhance* means increasing value, accuracy and validity; *levels* run from product concept to disposal, from factory equipment to the enterprise, from material transformation to knowledge transformation; *decision* means understanding the impact of change (visualise, organise, identify alternatives).

6.11.1 Approaches and Classification

Three paradigms define product, process and resources within cost, weight, investment, timing and quality constraints:

- **Design centered VM**: provides manufacturing information to the designer during the design phase.
- **Production centered VM**: uses simulation to model manufacturing processes for inexpensive, fast evaluation of many processing alternatives.
- **Control centered VM**: adds simulations to control models and actual processes for seamless simulation for optimisation during the actual production cycle.

Classification of VM: (1) by type of integration system, real physical systems (RPS) become virtual physical systems (VPS) and real information systems (RIS) become virtual information systems (VIS) for processing and decision making; (2) by type of product and process design: design centered, production centered, control centered; (3) by functional usage: virtual prototyping, virtual machining, virtual inspection, virtual assembly and virtual operational systems.

6.11.2 Methods, Tools, Significance and Obstacles

Core activities are modelling and simulation: representing the model in a computer environment and correlating its response with the real system to a certain accuracy and precision. Tools: *core technology* (fundamental, represents what VM can do), *enabling technology* (necessary to build a VM system), *show stopper technology* (without it a VM system cannot be built) and *common technology* (widely used and important). Methods: manufacturing characterisation; modelling and representation; visualisation and environment construction; verification, validation and measurement; multi discipline optimisation; flow simulation; graphical 3D kinematics simulation; finite element analysis.

Significance: validate feasibility and accuracy of production and process plans; optimise the production process and performance; enhance production feasibility and reduce fixed cost; support management decision making and strategic planning. Applications: online training, checking NC programs and tool paths, virtual sheet metal processing, virtual machine tools, virtual inspection, automotive and aerospace quality. Case study: Boeing simulates the entire assembly process; it saves more than 2,000,000 dollars annually; a VM center built in Ohio in 2008 compressed the development cycle, allowed all what-if scenarios in the 3D environment and improved collaboration.

Table 6.16: Obstacles and benefits of virtual manufacturing

Obstacles	**Benefits**
Technical: operational speed of the system; human understanding of manufacturing, modelling and presentation; artificial intelligence; unifying data standards; database management; evaluation techniques and standards. Social: not suitable for short term planning	Enhanced risk measures and control (predict product development cost and problems, information on production process); shorter product development cycle (computer based product models and prototypes, no physical prototype needed); enhanced competitive edge (customers can take part in product development, designers respond faster)

6.12 Knowledge Based Engineering

Definition 6.10 (Knowledge based engineering (KBE))

KBE is the process of capturing product design knowledge and re-using it, by defining the parameters and rules of the design process and integrating them into the design. The output is a dynamic product that lets designers adapt the design in successive iterations without complete re-design.

KBE is a discipline with roots in CAD, a complementary development to CAx (computer aided technologies) in the 1980s; early roles were support tools for design engineers; the success of early prototypes led to KBE being considered the basis for generative design. Components of KBE: artificial intelligence, expert systems, knowledge based systems and CAD. An *expert system* is "a computer system that uses knowledge and inference procedures to solve problems that would require the expertise of a number of human experts". Modern CAD adds parametric design, parts libraries, non geometry based constraints and embedded rules to the drafting aid, consistent drawings and quick editing of traditional CAD. The architecture has a knowledge base (domain expertise as facts, relationships and rules that can be positive or negative, include or exclude, logical or procedural, geometric, configuration, marketing, cost), an inference engine (searches the knowledge base and applies rules; the driver of the application) and a user interface (keyboard and mouse input, screen output), and a user who brings design ability, manufacturability, saleability and profitability (capturing design rules, manufacturing capability, marketing constraints and cost models).

Successful implementations: of 20000 main parts of the Boeing 747, 777 parts designed with KBE gave a 73 percent reduction in design time; parametric CAD and analysis took one year to design rib feet once but KBE design and analysis took 10 hours. Case study: Jaguar XK8 headlamp design, where feasibility rules distributed between disciplines gave a response time of about four weeks that KBE reduced to minutes and allowed more design iterations.

6.13 Industry 4.0 and Integration of Intelligent Systems into CIM

6.13.1 Industrial Automation and Industrial IT; IT plus OT equals IIoT

Industrial IT comprises control and signal processing; communication and networking; simulation, design, analysis and optimisation; real time computing; and database. Information technologies (IT: mobility, ERP, analytics, database, cloud; IT systems) and operation technologies (OT: SCADA and DCS, controllers and PLC, field devices and sensors; field data) converge: **IT + OT = IIoT** (Industrial Internet of Things). IIoT elements: machines and sensors, tools, processing, user, analytics, connectivity and alerts. A cyber physical system (CPS) is a system of collaborating computational elements controlling physical entities; CPS are physical and engineered systems whose operations are monitored, coordinated, controlled and integrated by a computing and communication core (computation, communication, control).

6.13.2 Industry 4.0

Industry 1.0 was mechanisation using water and steam power; Industry 2.0 complicated machinery for mass production using electrical power; Industry 3.0 automation, computers and the internet (quality, reliability, repeatability); Industry 4.0 is connected manufacturing and smart factories: a combination of cyber controlled, monitored physical systems, complex automation and IoT. Today's factory is a rigidly sequenced production line (workstations 1 to 4); tomorrow's factory is decoupled, fully flexible and highly integrated (workstations that can be reached in any order).

Table 6.17: Six design principles of Industry 4.0

Principle	**Meaning**
Interoperability	Ability of cyber physical systems (workpiece carriers, assembly stations, products), humans and smart factories to connect and communicate through the Internet of Things and the Internet of Services.
Virtualisation	A virtual copy of the smart factory created by linking sensor data with virtual plant models and simulation models.
Decentralisation	Ability of cyber physical systems in smart factories to make decisions on their own.
Real-time capability	Capability to collect and analyse data and provide insights immediately.
Service orientation	Offering of services of cyber physical systems, humans and smart factories via the Internet of Services.
Modularity	Flexible adaptation of smart factories to changing requirements of individual modules.

Building blocks of Industry 4.0: autonomous robots, simulation, horizontal and vertical system integration, industrial Internet of Things, cyber security, additive manufacturing, augmented reality and big data analytics. The smart factory combines virtual reality, simulation, flexible manufacturing system, augmented reality, cyber physical systems, artificial intelligence, cloud computing, big data and additive manufacturing. The digital enterprise digitises and integrates the entire value chain (product design, production planning, production engineering, production, services, with suppliers) from the digital world to the real world using simulation, product lifecycle management (PLM), manufacturing execution systems (MES), totally integrated automation and lifecycle and data driven services.

6.13.3 Intelligent Automation

Intelligent automation (IA) is a combination of robotic process automation (RPA) and artificial intelligence technologies that together empower rapid end to end business process automation and accelerate digital transformation. Smart automation combines a deep understanding of workflow with robust granular data to allow efficient, accurate and controlled decisions; for example "automation" is the standard cruise control that maintains a car's speed, whereas "smart automation" is the autopilot or enhanced cruise control using cameras, radar and sonar to read the environment. IA is based on business process management (BPM) to orchestrate users, tasks, systems and robots (RPA), with analytics and AI (especially machine learning) for automated and intelligent decisions, case management for flexibility, and integration between systems to prevent duplication of data. Technologies: artificial intelligence, machine learning, computer vision (for example OCR), natural language processing and process mining. Objectives: improve customer and employee experience, boost productivity, save time and cost, reduce human intervention and errors and reduce paper use. Business benefits: end to end processes for flexible, resilient business models; freeing employee time; collaboration between IT and business functions; involvement of employees in the transformation.

Table 6.18: Roadmap for digital next intelligent manufacturing

Stage	**Content and outcomes**
1. Connect the plants and enterprise (data foundation)	IT-OT convergence; sensor and connectivity; gateway solutions; multi device support; M2M connectivity; data integration. Outcomes: real time visibility, capacity optimisation, improved quality.
2. Conduct predictive and prescriptive analytics (advanced analytics and AI/ML)	Real time data management; data lakes; predictive modelling; AI/ML models; interactive dashboards. Outcomes: overall equipment efficiency and reduced inventory, decreased lead time, cost savings.
3. Drive cognitive intelligence (cognitive)	Operationalise AI/ML; contextualisation; self tuning algorithms; digital feedback loop automation; automated decisioning. Outcomes: improved service levels, optimised pricing, responsive and flexible value chain, proactive risk management.
4. Augment intelligence on edge (augmented and edge analytics)	AR/VR enablement; AI/ML models on edge; connected products and services; autonomous decisioning. Outcomes: productivity and efficiency, margin improvements, product or service innovation.

Table 6.19: Top 10 skills relevant in Industry 4.0 (World Economic Forum)

Rank	**In 2020**	**In 2015**
1	Complex problem solving	Complex problem solving
2	Critical thinking	Coordinating with others
3	Creativity	People management
4	People management	Critical thinking
5	Coordinating with others	Negotiation
6	Emotional intelligence	Quality control
7	Judgment and decision making	Service orientation
8	Service orientation	Judgment and decision making
9	Negotiation	Active listening
10	Cognitive flexibility	Creativity

Manufacturers expect from a connected factory (survey categories): increased production and output, reduced staff cost, reduced overall cost of production, improved quality, improved accuracy, flexibility of production, energy savings, reduced production cycle time, better working environment, improved health and safety, new orders won through improved competitiveness, and other.

6.13.4 Cyber-Physical Systems in Manufacturing

Systems that combine computing, communication and control with physical processes and elements are cyber-physical systems (CPS), used in manufacturing, transportation, healthcare and energy. They usually integrate sensors, actuators and other devices with control systems and networks for real time monitoring and control of physical processes. Observability, the ability to monitor and understand system behaviour and find bottlenecks, is essential. Seen from a manufacturing viewpoint, a CPS is an internet enabled physical entity such as a pump or compressor, embedded with computers and control components consisting of sensors and actuators; it is assigned an IP address, can self monitor, generates information about its own functioning, communicates with associated entities or outside, and operates in a self regulating and autonomous manner. Examples include industrial control systems, smart grids and autonomous vehicles; CNC machines, CMMs, data acquisition systems and SCADA networks are cyber-physical entities in a manufacturing plant.

The development of CIM and intelligent manufacturing followed computing and communications: computers gave numerical control of machine tools and robots; the microprocessor is the heart of CNC; computer graphics gave CAD; computer networks and databases enabled CIM; AI and machine learning contributed to intelligent manufacturing systems; the internet enabled concurrent engineering, extended enterprises and supply chains; multi agent systems supported agent based and holonic manufacturing; and wireless communication, sensor networks and IoT made high resolution manufacturing systems and tracking and tracing possible.

Table 6.20: Future trends of CPS (source)

Trend	Meaning
Integration with IoT	Sensors and devices connect physical objects to the internet, giving more data sources.
AI and machine learning	More advanced data analysis and decision making; adaptation to changing conditions.
Cloud based CPS	Greater scalability and flexibility than on-premises systems.
Security and privacy	Secure communication protocols and measures against cyber threats.

6.13.5 Architecture of an Intelligent Manufacturing System

The source describes new *models* (Internet based, service oriented, collaborative, customisable, flexible and socialised intelligent manufacturing systems), new *means* (human machine integrated smart manufacturing systems featuring digitalisation, IoT, virtualisation, service, collaboration, customisation, flexibility and intelligence) and a new *form* (an intelligent manufacturing ecology characterised by ubiquitous interconnection, data drivenness, cross border integration, autonomous intelligence and mass innovation). Their deep integration forms an intelligent manufacturing ecosystem. The application of AI only makes sense inside an intelligent manufacturing system.

Table 6.21: Layers of an intelligent manufacturing system

Layer	Role
Resources and capacities layer	Human, machine, material and other manufacturing resources and capabilities
Ubiquitous network layer	Interconnection of resources
Service platform	Common services
Intelligent cloud service application layer	Applications delivered as services
Security management and standard specification system	Security and standards spanning all layers

6.13.6 Emerging Technologies: CPS Security, Big Data and Blockchain

These topics are supplementary to the core syllabus and are summarised briefly.

Cyber-physical security. Cyber-physical attacks go beyond intellectual property theft and affect the physical world: changed product designs, manipulated manufacturing equipment and altered final products. Examples: tool path files modified in a subtractive process and design files altered in an additive process. Undetected outcomes include defective products; consequences include delayed launch, ruined equipment, higher warranty cost, loss of customer trust and physical harm. IoT, cloud and SaaS based CAE tools enlarge the attack surface; manufacturing security is often treated as pure IT, which cannot mitigate cyber-physical attacks. Vulnerability types listed: hardware, software, technical (human weakness), network, platform and management (absence of security policy standards). The vulnerability assessment approach holds that vulnerabilities occur at intersections of four kinds of entities.

Table 6.22: Entities in the manufacturing vulnerability assessment approach

Entity	**Meaning**	**Examples**
Cyber	Pre-processing, saving, transferring, managing or post-processing digital information	MRP, PLM, ERP, CAE tools, data management, data mining, inspection reporting
Physical	Tangible and not fully governed by automation	Manufactured parts, manual machines, raw material, manual inspection equipment
Cyber-physical	Cyber and physical elements interacting autonomously, with or without supervision	CNC machines, CMMs, DAQ systems, SCADA networks
Human	Anyone who can interact with other entities	IT support, designers, manufacturing engineers, machinists, quality engineers, maintenance crew

Impact metrics: loss of information, inconsistency, relative frequency, lack of maturity and time until detection.

Big data analytics. Big data in manufacturing is generated by production and process machines (pumps, motors, compressors, meters, sensors, controllers, conveyors) and must be analysed and visualised to give insight. Its value lies in predicting and actively solving problems, and moving from predictive to *prescriptive* analytics that plan and prevent future disruption. Uses: streamlining resource management, improving efficiency, optimising product development, new growth and smart decisions; near real time root cause determination of failures, issues and defects; anomaly detection faster and more accurately than the human eye. This is the data basis of the machine learning methods above.

Blockchain. A blockchain is a shared database of records stored in blocks linked chronologically and cryptographically, identical copies being held by the nodes of a network, so an altered copy can be verified against the others. A transaction authenticated by the nodes is added as a block carrying the information, timestamp and cryptographic hash, so a recorded block cannot be altered or deleted and duplicate entries are impossible.

Manufacturing uses: supply chain transparency and traceability of parts before, during and after manufacture (for a recall the source of the part, the affected vehicles and the part's journey are found at once), prevention of fraud and counterfeit components by admitting only trusted participants, automated payments through *smart contracts*, ethical sourcing and compliance verification, and provenance for consumers. Combined with IoT and AI analytics, orders and payments can execute automatically from forecast demand. Challenges: decentralisation is hard to achieve together with scalability; immutability and transparency conflict with privacy; integration across established supply chains is difficult. Blockchain based digital twins are mentioned as a further direction.

6.13.7 Integration of Intelligent Systems into CIM

Intelligent systems fit into the CIM structure at each level: CAD with knowledge based engineering and generative design; CAPP with feature recognition and knowledge based planning; CAM with automatic feature recognition, knowledge based machining and real time simulation; CNC with AI, machine learning, monitoring and adaptive control; shop floor with IIoT sensors and dashboards; quality with vision based inspection and closed loop feedback; and planning with digital twins for scenario simulation. The STEP and STEP-NC data model gives the common information base that lets these intelligent components exchange data bidirectionally.

Key Examination Points

- A manufacturing CPS is an internet enabled physical entity with embedded computation, sensors and actuators that self monitors and communicates; cyber-physical attacks affect the physical world.
- Layers of an intelligent manufacturing system: resources, ubiquitous network, service platform, intelligent cloud service application, security and standards.
- Emerging topics (big data, blockchain) are supplementary: prescriptive analytics, traceability and smart contracts.

6.14 Product Lifecycle Management

Definition 6.11 (Product lifecycle management (PLM))

PLM refers to the management of data and processes used in the design, engineering, manufacturing, sales and service of a product across its entire lifecycle, from idea through retirement, and across the supply chain. It provides the capabilities to manage information and to facilitate communication and collaboration across that lifecycle.

PLM was first built for engineers to collaborate on the latest designs on on-premises technology; it has since grown to include customer service, marketing, sales, suppliers and partner channels, and cloud based PLM gives a single source of truth updated in real time across sites. A "product" is anything offered on the market to meet a demand, tangible or a service. Its integration with CAD (parametric models), CAE (virtual simulation and test) and CAM (virtual manufacturing to find problems before a facility is built) is the link to this book's CIM framework, and it supports concurrent engineering by performing tasks in parallel.

6.14.1 Phases of the Product Lifecycle

Table 6.23: Beginning, middle and end of life

Phase	**Content**
Beginning of life (BOL)	Design and manufacture: conceptualisation, development, prototypes, requirements, concepts and testing; specifications, production process and supply needs.
Middle of life (MOL)	Distribution (logistics), use, service and maintenance; data on failures, maintenance rates and user experience feed immediate fixes and future development.
End of life (EOL)	Products are recollected, disassembled, remanufactured, recycled, reused or disposed; reverse logistics; recovering information on parts and materials still of value.

A second, marketing oriented view of the lifecycle has four stages: introduction (costly and risky), growth, maturity and decline. Care is needed not to confuse the two: BOL, MOL and EOL describe the engineering and service life of an individual product, while introduction, growth, maturity and decline describe the market life of a product line. In the engineering process, stages named in the source are design, prototype, production, inventory and obsolescence.

6.14.2 PLM Functions, PDM and Engineering Change

PLM systems (for example Teamcenter and Windchill) support the ECR/ECO process (engineering change request and engineering change order), updating the bill of materials (BOM), managing change notifications and problem reports, and communication with vendors and suppliers; they log raw materials and parts, track manufacturing procedures, keep version history and control, and record changes in customer demands, regulations and cost. Two development strategies are recognised: starting from scratch and modifying an existing product.

A *product data management* (PDM) system stores CAD models, drawings, BOMs and other product documents, controlling access by checking files out and in. PDM sits within the PLM system and within the ERP environment.

Table 6.24: PDM versus PLM (synthesised from the text; the source slide comparing them is a picture)

Aspect	PDM	PLM
Scope	Storage and control of product data: models, drawings, BOM, documents	Data and processes over the entire lifecycle, from idea to retirement, across the supply chain
Function	Check in and check out, vault, versions	Change management (ECR/ECO), BOM, collaboration, portfolio and compliance across departments
Relationship	A component (repository) inside PLM and ERP	Enterprise level system that includes PDM and integrates CAD, CAE, CAM, ERP and supply chain
Users	Mainly engineering and design	Engineering plus manufacturing, service, sales, suppliers

6.14.3 Design Approaches, Benefits and Challenges

Design approaches supported by PLM: concurrent engineering workflow design (tasks in parallel), bottom-up design (CAD centric, parts assembled and the BOM generated), top-down design (high level requirements first), both ends against the middle (combining the two), front loading (gathering downstream design and data before the product is defined, with stage gates) and design in context (designing within the assembly). Benefits: faster time to market, lower development cost, more transparent lifecycle, data driven decisions, improved product quality, reduced compliance risk and a single source of alignment. Challenges: keeping records current when specifications change, disconnect between manufacturing and operations, mismanaged supply chain, difficulty tracking metrics, training time and multiple sources of specifications. Evaluation metrics include innovation (sales from new products), lead time, cycle time, product life, waste, reliability and warranty claims. PLM extends beyond sales through IoT: product monitoring, product performance and proactive or reactive maintenance data close the loop between design, development and operation. Leading PLM vendors named are Autodesk, Dassault Systemes, PTC and Siemens (Teamcenter).

Key Examination Points

- PLM manages product data and processes from idea to retirement; PDM is the data vault within it.
- BOL (design, manufacture), MOL (distribution, use, service), EOL (recycle, reuse, dispose).

- PLM functions: ECR/ECO, BOM management, change notifications, supplier communication.
- PLM integrates CAD, CAE and CAM and supports concurrent engineering.

6.15 Chapter Summary and Examination Points

Key Examination Points

1. Data exchange approaches (eliminate, translate, exchange); translator count formulas; kernels ACIS and Parasolid.
2. IGES (six sections, 5.2) versus STEP (life cycle, EXPRESS, Part 21, three layers, APs 203, 213, 214, 224, 238, 240).
3. STEP-NC (ISO 14649): features, workplan, workingsteps; bidirectional; 35, 75 and 50 percent gains; function blocks IEC 61499 and controller modules.
4. AI and ML in CNC: tool wear prediction, adaptive machining, quality prediction, predictive maintenance; four benefits; AI in G and M codes.
5. Intelligent CNC: four feedback levels; DSP motion control and intelligent machining modules; management, prediction, strategy, observation.
6. Digital twin: five implementation steps; six applications; bottleneck resolution five steps; Station 3 example.
7. Virtual manufacturing: three paradigms, classifications, technology types, obstacles and benefits; KBE components and architecture.
8. Industry 4.0: six design principles, IT plus OT equals IIoT, smart factory, roadmap stages.
9. Prognosis: monitoring, diagnosis, prognosis and maintenance decision are distinct stages; RUL and health index; five module framework; supervised, unsupervised and reinforcement learning.
10. PLM (Section 6.14): BOL, MOL, EOL; PDM inside PLM; ECR/ECO and BOM management.