Chapter 2
Design Process and Its Fundamentals

In this chapter, we describe the basics of the development process for electronic systems. We will see how service-proven standards and norms along with standard drawings and computer technology can be used to break down the design process into separate activities, which are then more easily performed. Every research and development engineer needs to be familiar with these design activities and with the requisite technical documentation (technical drawings, circuit diagrams, CAD models) in order to produce successful electronic products.

2.1 Life Cycle of Electronic Products

The electronics industry is an exciting place to be these days, and indeed most technical innovations today come from this industry. Fifty percent of many companies' sales come from products that are less than five-years old. These products need close attention not only just when they are first designed but also through their entire life cycle. Figure 2.1 shows the typical life cycle of a product from a business standpoint. The product life span can vary. In order for a product to be economically viable, a business must establish whether the development costs can be recouped and, if yes, over what time period.

The different stages in a product's life, also known as the product life cycle, typically consist of development (*development stage*), use (*marketing stage*) and disposal (covered in Chap. 7). The development stage consists of the following steps:

- product planning,
- design and development, and
- first production run with prototype build and pilot series.

As we can see in the middle of Fig. 2.1, the growth rate is high at the beginning of the marketing stage and then the product matures. The market for the product and

J. Lienig and H. Bruemmer, *Fundamentals of Electronic Systems Design*,
DOI 10.1007/978-3-319-55840-0_2

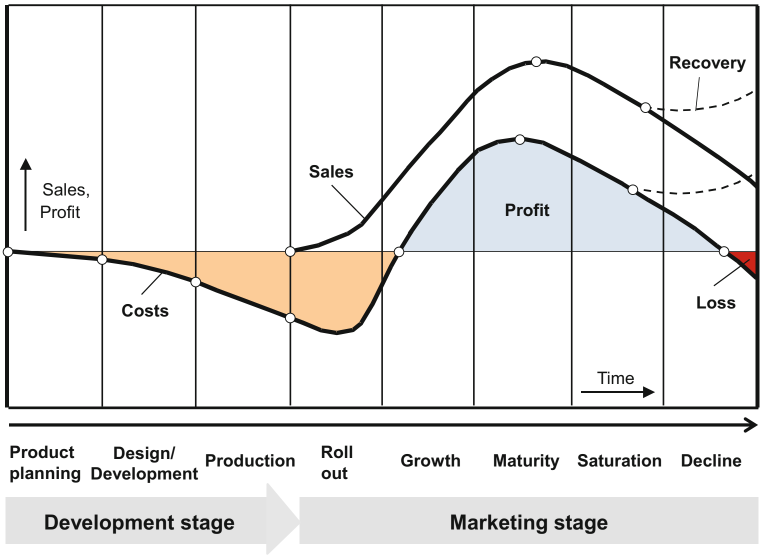


Fig. 2.1 Development and marketing stages for a product [1, 2]

the competition are known. The maturity stage is superseded by the saturation stage, where there is only low or no growth. As more players enter the market, the sales price comes under increasing pressure. Direct costs are driven down by increased productivity. But, despite this, the relationship between sale price and direct costs is increasingly reduced. Finally, during a period of decline, the product is typically forced from the market by the competition or by the other substitute products.

The development process for an electronic system described in this book begins with the *product planning* stage. During this initial planning period, ideas for products are generated and assessed, and project tasks are formulated. The subsequent *design and development* stage uses as input the functional specification developed with the product proposal. At the end of the design and development stage, the complete product documentation containing all instructions for product fabrication, use, maintenance, and disposal (including recycling) is produced.

2.2 Design and Development Process

The design and development of an electronic system or device is a key element of the preparations for manufacture; in essence, function, quality, and costs are defined at this stage. *The design process encompasses all creative, manual, and technical activities necessary to define the product and which need to be carried out to* *convert a system definition to a sufficiently detailed system design specification for product manufacture and deployment.*

Before we proceed, it is helpful to clarify a bit of terminology. The term "design" typically covers the first prototype for a system or a device, while the term "development" generally includes the production of the documentation (e.g., circuit diagrams for electronic modules, drawings for mechanical components) necessary for fabrication. As it is often difficult to differentiate these two terms, they are considered as synonyms throughout this book.

Design and development can be divided into four stages, each with different definitions [1]:

– task definition (informative definition),
– conceptual stage (cardinal definition),
– design stage (formative definition), and
– implementation stage (manufacturing definition).

Information is gathered on the requirements for the product to be developed during *task definition*. The result is an informative definition in the form of a requirements specification. These requirements are then transformed into an optimal technical principle at the *conceptual stage*. The cardinal definition of a solution is thus formulated here along with proof of functionality.

A system design based on the conceptual solution is presented at the *design stage*. The objective is to determine the best overall design for the product taking technological and economic constraints into consideration. Finally, fabrication and usage details are set out at the *implementation stage* to facilitate manufacturing and deployment.

Engineering guidelines usually define seven steps for the design and development process, which add further details to the four stages above:

1. Detailing the system definition and the requirements specification	Task definition
2. Establishing functions and their structures 3. Searching, assessing and selecting solutions	Conceptual stage
4. Splitting the systems to be developed into individual modules 5. Designing the individual modules 6. Designing the entire system, including installation plan	Design stage
7. Developing the fabrication, use, maintenance and disposal details, building a prototype (optional).	Implementation stage

Figure 2.2 shows the steps involved in the overall product design process. A number of variants, which subsequently need to be optimized and streamlined, are produced in several of the steps. Design teams often use creativity techniques, such as brainstorming, to produce these variants. Variants are optimized and eliminated with the help of selection and assessment methods, such as weighted lists [1].

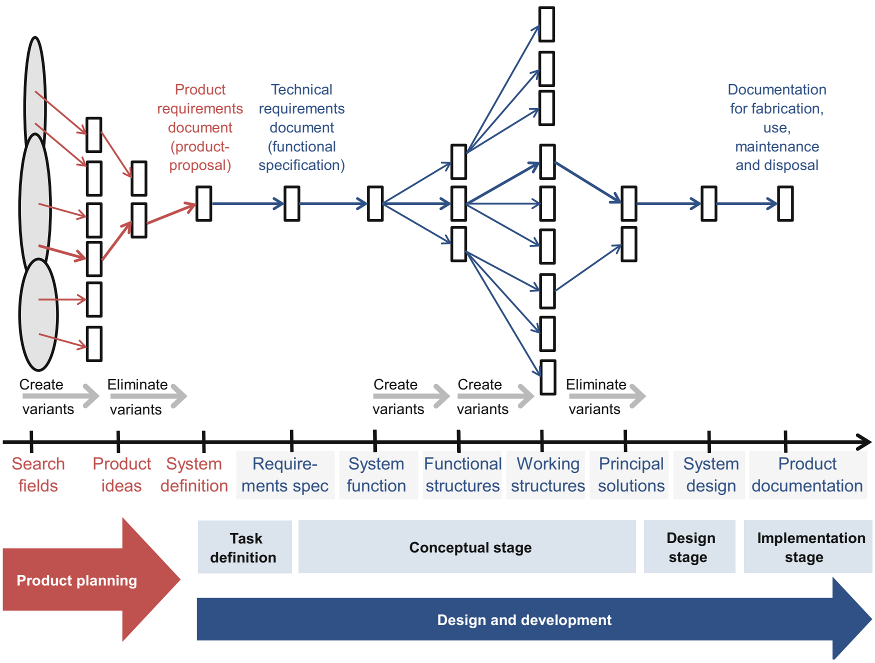


Fig. 2.2 Steps and results of the product planning and development processes. Development effectively begins with the system definition based on the product proposal. The development work is divided into the task definition and the conceptual, design and implementation stages. How variants are produced and eliminated is also shown

At the end of the design and development process, a prototype is typically produced. This first prototype should be tested under conditions that replicate as closely as possible the conditions that will be encountered later in real-world scenarios.

2.3 Guidance for Product Planning, Design and Development

In today's consumer-driven industry, development departments must respond rapidly to market changes. They need to draw up a work schedule along with cost estimates to ensure optimal use of resources. This is particularly critical for more complex systems, where knowledgeable specialists from a wide range of backgrounds are often involved. As a prime example of a complex system, we shall examine a "vending machine" whose block diagram is shown in Fig. 2.3. The vending machine example will illustrate key concepts and steps that design teams will use for a broad range of consumer and industrial products.

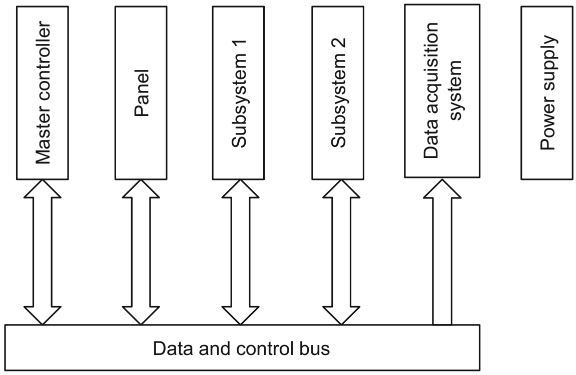


Fig. 2.3 Block diagram of a vending machine

In our vending machine example, the user selects an item via a panel with keyboard and screen which also displays the price. Coins are identified and their values determined by the cashier unit (subsystem 1). The coins are stored in the device. The machine gives change if needed on a command from the master controller.

The development of an electronic coin validator with high detection reliability for counterfeit coins and other currencies requires a team of qualified electronic, mechanical, and test engineers, drawing upon their specialized knowledge. The system definition needs to be precisely formulated; duties and responsibilities should be clearly set out so that there is very little wasted capacity. The same applies to the merchandise storage space (subsystem 2) and associated output unit with actuator.

The master controller checks all operations in the vending machine. When a button is pressed and an item is selected, the price is read from an electronic price register and displayed. When payment is received, the controller calculates the difference between the sale price and the value of the inserted coins and displays it on the screen. If change needs to be given to the customer, the cashier unit returns the largest coins possible. The master controller controls all drives in the vending machine, calculates the values for data acquisition, and outputs an alarm message in the event of a fault in a module.

The data acquisition system stores sales transactions over a longer period and gathers statistics. The power supply unit powers circuits, screens, and actuators.

All functions and subfunctions must be accurately defined when designing such a system, to ensure correct operation. Even in this introductory example of the vending machine, we can see how complexity grows as our requirements become more detailed and specific. We manage this complexity by defining high-level functions, subfunctions, and interactions with which we will implement these requirements. The specification also depends on the type of components and circuits to be used; for example, conventional digital logic or a computer could be deployed as master controller. The project engineer is therefore required to investigate and define the system early in the project. As the number of people involved in the project grows, so too do costs and coordination issues. It is therefore imperative from a business perspective that joint activities within the team be effectively scheduled.

2.3.1 Planning Development Work

The argument is often made that development work cannot be planned. However, this is not correct, as the truly creative work only takes on average 10–15% of the total project time, with the remaining time taken up with well-defined tasks that will transform these creative ideas into a viable product. Workflows and planned times for such tasks should be defined by the engineers involved in the project, as they are the only people with the necessary knowledge.

First, as much information as possible should be collected on the project. The mission should then be formulated. Subsequently, different solutions will be developed and evaluated. Once the best solution based on the requirements specification has been selected, the system to be developed is defined, the engineering work is set out, and projected costs are calculated.

The timetable for the design process should not be seen as a restriction on the work of the development engineers or as pressure on them. Rather, its purpose is to make the complex decision process more transparent and get superiors involved. Indeed, management has to take responsibility for decisions. More often than not they have a better understanding of how the overall system fits in the marketplace and of customer concerns, but have less detailed system knowledge.

Project planning increases the likelihood of success by preventing false starts; it permits the development engineer to start a project only after all solutions have been examined and assessed; and the design criteria established as per the requirements specification. The project engineer should study all technical details before starting any development work.

He/she should document all his/her reviews and investigations as well. This is very important, so as to allow others to participate in decision making for certain solutions or for the entire project, and to understand later why and how certain decisions have been made. In addition, as new engineers join the team, they can quickly come up to speed by reviewing such documentation and understanding key decisions.

2.3.2 Information Flow

The development engineers should work closely and continuously with the customer right from the start. The customer is the company's customer in the case of stand-alone, customized devices. Sales, i.e., the marketing department, often assumes the role of the customer for standard, mass-produced products. Typically, they know the problem to be solved, but not the deployed technology. The opposite is true for development engineers. Hence, unnecessary delays and costs will be avoided when there are competent contact persons on both sides, working closely together in an iterative, collaborative manner.

Knowledge of certain merchandise groups and understanding the requirements restricting the system to these merchandise groups is pivotal in the case of the vending machine example introduced earlier. For example, overall costs can likely be significantly reduced if very large or heat-sensitive items of merchandise are not offered by the machine. Customer options to adjust the price register, the statistical data gathered, the UX/UI (User Experience/User Interface), and the system design are other topics that must be considered.

Direct contact between customer and the company's development engineers will facilitate a speedy and seamless system handover. Costs for the proposed solutions should only come from sales people as they are the only people who know the valid cost plan. The development team should be in close contact with the fabrication department, as the manufacturing technology used in a company typically has a major impact on product design.

Companies are often organized as per Fig. 2.4. The term "development" often stands for the development department with its own R & D laboratory and design departments and may be used to describe research laboratories as well.

All electrical, electronic, physical and chemical investigations, and experiments needed for system development are typically carried out in laboratories in many companies. Circuit diagrams and documentation, used, for example, for IC and

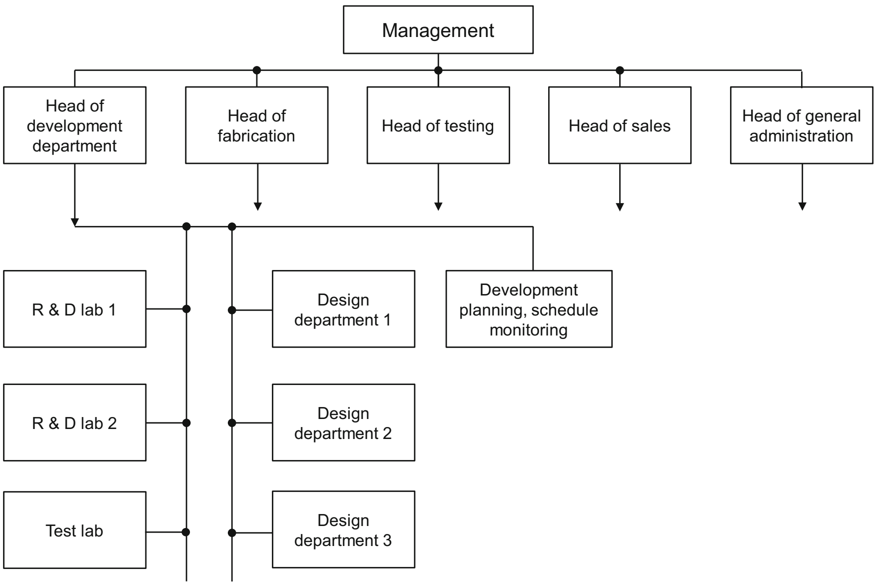


Fig. 2.4 Organization chart of a company

PCB layout design and packaging, and technical drawings for mechanical components are produced during these tests. Prototypes are also assembled and tested in test areas at this time.

2.3.3 Feasibility Study During Product Planning

The system to be developed is first roughly described in a *feasibility study* during product planning, in preparation for system development. It may also be possible at this stage to determine the features a system should not have if system characteristics cannot yet be clearly defined. The feasibility study typically contains details on the application and the customers, an analysis of competitor systems (rival products), and a rough estimate of the sales targets and required total investments.

The purpose of the feasibility study is to help the company decide whether or not to go ahead with the project. The feasibility study need not be carried out if the company executives decide to go ahead with the project for other reasons, such as marketing.

A typical feasibility study for a vending machine, for example, would address the following questions:

- What customers and what merchandise groups should the system be designed for?
- What characteristics does the system need to have?
- Is the system an economically viable proposition for the customer considering price and performance? Has the customer expressed maximums or minimums for price or performance, respectively, and/or trade-offs between them?
- What is the expected sales volume for each customer group?
- How high are the maximum development and manufacturing costs for the assumed price?
- What capital investments are required to manufacture the system?
- What is the competition, and what advantages and benefits should the system offer over competing products?
- Are some of the system components, like the coin validator or the fully assembled cashier unit in the example, available as purchase items? This might be an option to eliminate development risks or cut development costs.

2.3.4 Task Definition and Conceptual Stage

The development process starts when a *product requirements document* is produced during product planning or there is a definite customer order. First, the system definition will be detailed (*task definition*) by collecting all necessary data from all available sources. Among the topics covered are: establishing the state of the art of the given technology, standards and norms, legal issues, the protection of utility models and patents, and any necessary authorizations.

The task definition reduces the level of abstraction for the design solution, and it becomes more concrete. At the same time, it defines the functional specifications for the development process.

The task definition produces the *functional specification* based on the customer *product requirement document* (Sect. 2.3.5). This *functional specification* thus contains, from the developers' view point, the company's specifications for building the system.

An intermediate report should be produced after the task definition. A decision will be taken about continuing the project based on this report. Solution options will then be identified for the proposed system, and standards for selecting the optimal solution will be formulated as well. The following tasks will then be carried out:

- Drafting the definitive functional specification (Sect. 2.3.5).
- Further detailing of the selected solution for identifying any major problems and drawing up a project structure plan (Sect. 2.3.6).
- Defining the reliability data and the maintenance intervals.
- Performing trials for components that have not been proven yet. These tests may continue until the end of the development period. Alternative solutions may be needed.
- Drawing up a training program for service personnel and producing service documentation for the course.
- Determining criteria (e.g., functions, costs, and time) whose compliance is required for securing the project. These criteria should be reviewed regularly during the project.
- Drawing up a development time schedule (Sect. 2.3.6) and a report with recommendation (or disapproval) to continue with the project.

The design stage should only start if the solutions submitted after the conceptual stage have been successfully verified. Junior engineers often make the mistake of ending the conceptual stage too soon, i.e., they start designing components too quickly before all solutions have been identified and assessed.

The project work described here is often affected by modifications to the system definition after the functional specification has been drafted and before the system is deployed. These changes could be caused by market changes, by rivals, or by other technical requirements from the customer. The impact of such changes on delivery dates and costs can be more accurately estimated if sufficient design and engineering documents are available when such a situation occurs.

Executing the project as described above brings transparency to planning and decision making for higher management, and increases the chances of technically and commercially successful project completion.

2.3.5 Functional Specification

The design and development process starts with a functional specification, typically outlined in a *technical requirements document*. This is a list of requirements the system has to comply with, along with the technical definitions of the operating environment. These requirements are the technical response to a matching requirements document, the *product requirements document*, created from a user's point of view by the contracting party or the product planning department. Hence, the functional specification contains engineering details for developing the system based on the product requirements document from the customer.

Requirements should be ordered according to their priority:

- Requirements that the system must fulfill for the targeted customers,
- Features that could attract a wider group of customers if the price is not increased at all or is only marginally increased, and
- Low-priority requirements that do not necessarily have to be complied with ("nice to have").

A typical functional specification will contain the following:

- Precise description of the system definition (what is the purpose of the product or system?),
- Description of the interfaces to the environment, such as other technical systems and humans,
- Size and weight definitions and installation conditions,
- Definition of the operational and environmental conditions,
- Standards of precision,
- Functional safety,
- Service life duration,
- Maintenance and repair requirements,
- Standards and regulations, such as mandatory standards,
- Storage conditions, transportation requirements, and packaging,
- Sales volumes,
- Approved development and manufacturing costs, sales price, running costs at customers, and
- Deadlines.

By asking product-specific questions, such as "Does the customer have experience with similar devices or components?" "What characteristics should the product *not* have?", additional information is often obtained that should be defined *a priori*.

2.3.6 Scheduling

Scheduling the design process can be a major challenge. There is no question that it is needed; management rightly demands specification and compliance with deadlines and costs from engineering departments. Proper scheduling helps avoid frustration and costly delays, which can adversely affect both management and engineering "enthusiasm" for the project. As the saying goes, *plan the work, then work the plan*.

A number of requirements must be fulfilled to create a useful planning system. The activities of different employees and departments in product development must be coordinated and scheduled. A *network plan* is particularly useful in this regard for large projects. This type of time schedule shows all project activities and their mutual dependencies in graphic form, as will be explained later. The following conditions need to exist for a successful network plan:

– The planning system must match the organizational structure;
– A competent member of staff should be responsible for planning and scheduling;
– All employees should be familiar with the planning system and should endorse it; and
– Sufficient time is allocated for the preparatory study and planning.

There is a difference between the planning system and its data. A lot of publications are available on different planning systems and techniques [1, 3].

A *project structure plan* (Fig. 2.5) is typically drafted when preparing a proposed development solution. This is a simple and self-explanatory schematic

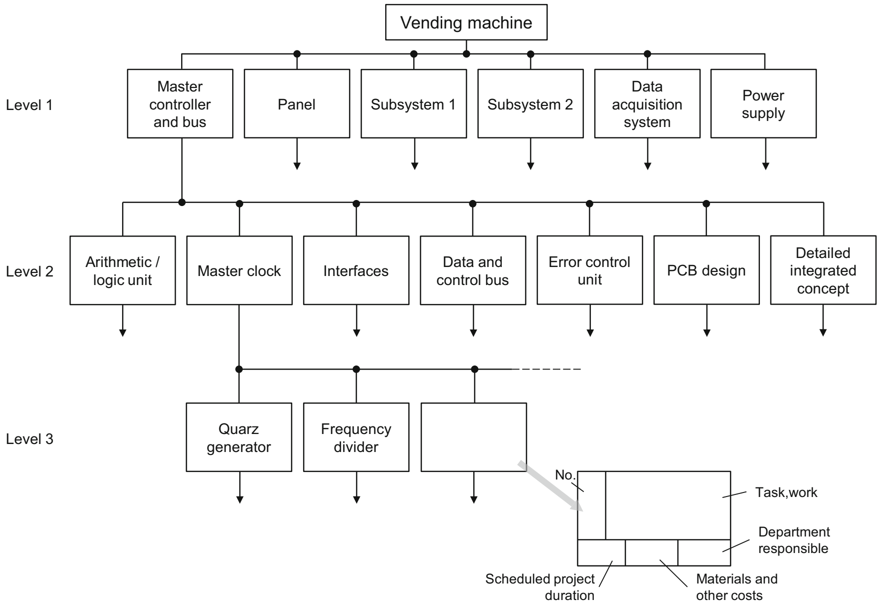


Fig. 2.5 Section of a project structure plan for the vending machine depicted in Fig. 2.3

representation of the work needed to complete the project. The individual levels in the project structure plan indicate the different subtasks and their relationships to higher- and lower-level tasks. The subtasks, representing the final items of the schedule, are called *work packages*.

The project structure plan is not yet the network plan. The former should always be drafted to ensure that no critical items are overlooked in the latter.

The work packages should be small enough to allow accurate *time estimates* for their completion. Practicable time units, such as hours, working days or weeks, should be used. The more detailed the work description, the more accurate the scheduling. And the more detailed the information available on the tasks, the better the resulting plan will be. One of the purposes of the preparatory study is to examine the less clearly defined topics and to gather as much information as possible on them. The work will need to be divided into smaller packages if some issues remain unresolved. A review will then be carried out at the end of each work period, and work rescheduling may be necessary.

When estimating times, remember that development engineers are often sidelined from their design work such as they may spend up to 40% of their time at meetings, talking on the phone, writing e-mails, and doing other works.

When the project structure plan has been drawn up, the work packages are divided into *processes*. A process is a time-consuming activity with a defined start and finish. Another process attribute is that it incurs costs. Additionally, a process is performed without a break from start to finish.

Individual processes in a project cannot be performed in any order. It may be necessary to start a given process only when other processes are complete. This gives rise to *relationships* and *dependencies*.

The logical project plan is defined and the *network plan* can be drawn up when the predecessors and successors for individual processes have been established (Fig. 2.6). It is sometimes easier to draw up the network plan starting with the project end date and "work backwards." This is because it is often simpler to

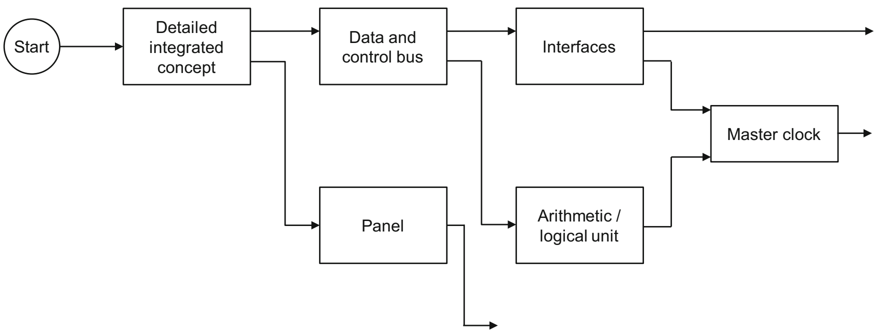


Fig. 2.6 Section of a network plan (activity-on-node network) based on the project structure plan for the vending machine in Fig. 2.5

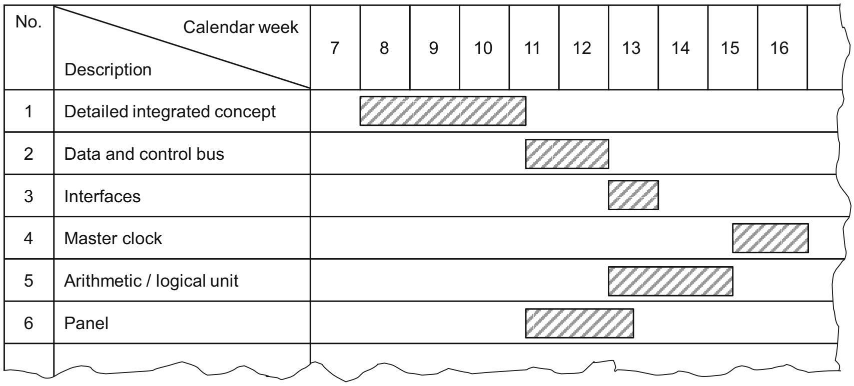


Fig. 2.7 Section of a bar chart

establish what needs to be done before a process is started than to determine the steps needed after its completion.

The earliest and latest start and end times for individual processes are calculated from the relationships and time estimates. There will be many free periods (buffer periods) available for many processes. The path from the start to the end of the network plan, where there are no buffer periods, is called a *critical path*.

Finally, the time units used for planning are converted to calendar dates to build a *bar chart* or *Gantt chart* (Fig. 2.7). The following must be known for calendar planning:

- the project start date (calendar date),
- holidays during the project period, and
- the planning unit (days, weeks).

A bar chart can be produced without creating a network plan with projects that have very few processes and very few mutual dependencies.

2.4 Technical Drawings

All parts have to be uniquely and adequately described for fabrication in the *product documentation* generated at the end of the design and development process [4]. Freehand sketches of components, modules, and systems in compliance with standards should be read and produced at the conceptual stage. Typically, development engineers are required to deliver complex, often computer-based, *technical drawings*.

In order to communicate all needed information from the development to the fabrication process, a technical drawing must include the following critical information:

- geometry, i.e., the shape of the object, how the object will look when it is viewed from various angles,
- dimensions, i.e., the size of the object in accepted units,
- tolerances, i.e., the allowable variations for each dimension,
- material, i.e., what the object is made of, and
- finish, i.e., the surface quality of the object.

Using such standardized illustrations ensures global understanding and portability of the specifications and avoids misconstructions. A set of drawings for a system is made up of a number of items:

- main or general drawing (mandatory),
- group or module drawings (if required),
- detail part drawings (mandatory for all parts to be manufactured),
- bill of components (contains all system parts, also standard or purchased parts; it is mandatory), and
- assembly drawing (optional).

The *right-angled parallel projection* as per ISO 128-30 [5] has established itself as the benchmark for parts drawings. It has the advantage that all views can be displayed undistorted and true to scale with all dimensions. The front and side views are produced by the *orthographic projection*, which is the standard technique for drawing a physical object in different plan views (Fig. 2.8).

An object can have six views, whose defined location and orientation to one another are derived from the orthographic projection (Fig. 2.9):

	Bottom view		
Side view from the right	Front view	Side view from the left	Rear view
	Top view		

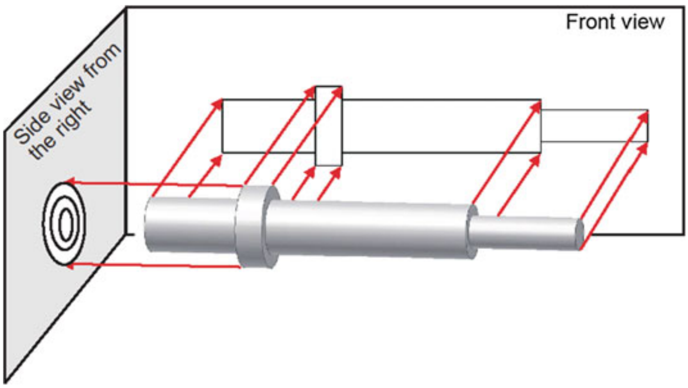


Fig. 2.8 Orthographic projection of an object for representing the front and side views. All visible object edges and the silhouette are shown in the viewing angle for each view

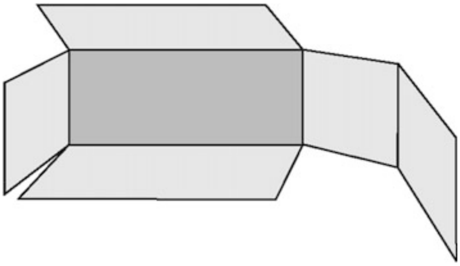


Fig. 2.9 Six views of an object are generated by projecting in the respective planes of a rectangular expandable projection box. The side view from the right, for example, is located on the *left* of the front view (see Fig. 2.8), and the bottom view is located *above* it

For engineering documentation, the views drawn are typically a subset of all six, namely only the ones needed to uniquely render the object. As an example, Fig. 2.10 pictures an object in all six views and Fig. 2.11 shows only the necessary views to be drawn.

The size of a drawn object when it is increased or reduced in size with respect to the original is indicated with a *scale*. While a scale of 1:1 represents the original size, enlargements may be represented by scales of 2:1 (20:1, 200:1, etc.), 5:1 (50:1, 500:1, etc.) and 10:1 (100:1, 1000:1, etc.). The same applies to size reductions that

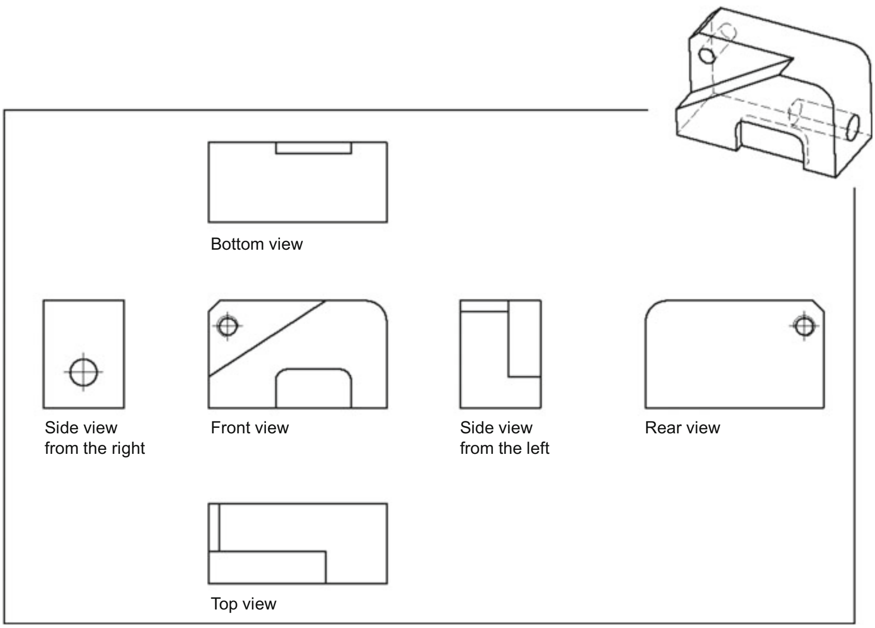


Fig. 2.10 Six possible views of an object and their orientation to one another

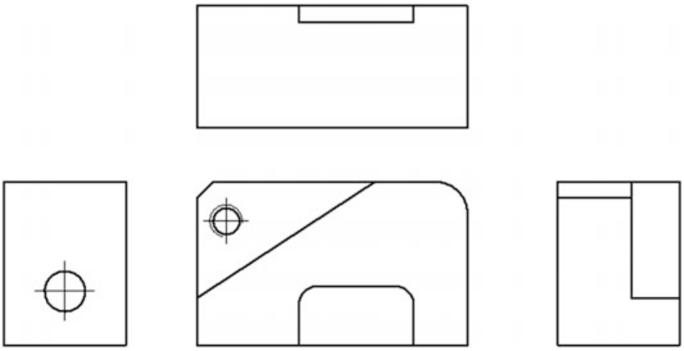


Fig. 2.11 Object in Fig. 2.10 can be uniquely shown in four views (front view, side views from left and right, bottom view)

can be drawn in the scales 1:2 (1:20, 1:200, etc.), 1:5 (1:50, 1:500, etc.) and 1:10 (1:100, 1:1000, etc.) [6]. The scale is specified in the title block on the drawing [7].

Sectional views, also known as sections, are used to illustrate features inside an object or to reduce the number of views (Fig. 2.12) [8]. Cut surfaces are shaded, and cut surfaces for the same part are shaded in the same way. Every sectional view is based on a non-sectional view of an object. The location of the sectional view on the unsectioned object may be indicated by a dash-dotted line for clarity purposes. The viewing direction is marked with arrows. Note that complete workpieces, such as shafts, pins, and screws, remain unsectioned in a sectional view [8].

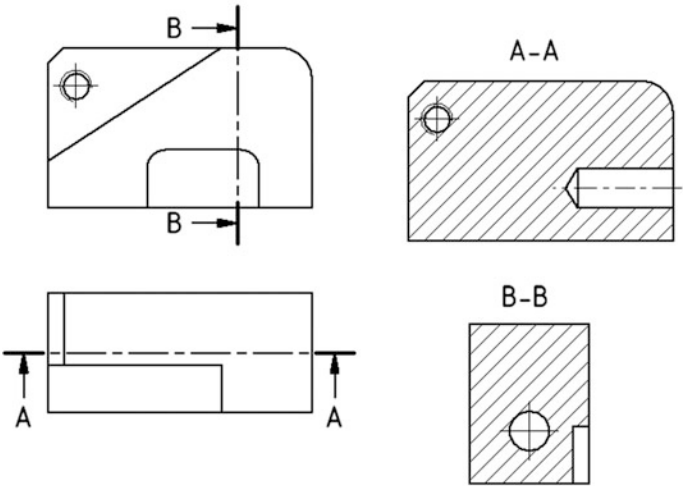


Fig. 2.12 Sectional views from front and top views for the object in Fig. 2.10 illustrate the locations and orientations of the holes, in particular

If the depicted object is drawn for manufacturing or testing, it must be uniquely *dimensioned* (Fig. 2.13). There are different types of dimensioning: function-based (functional dimension, i.e., the dimension is essential to the function of the piece or space), production-related (the dimension is relevant for manufacturing), and test-related dimensioning (inspection dimension). A functional dimension is specified only when needed. Functional dimensions should be specified first in a

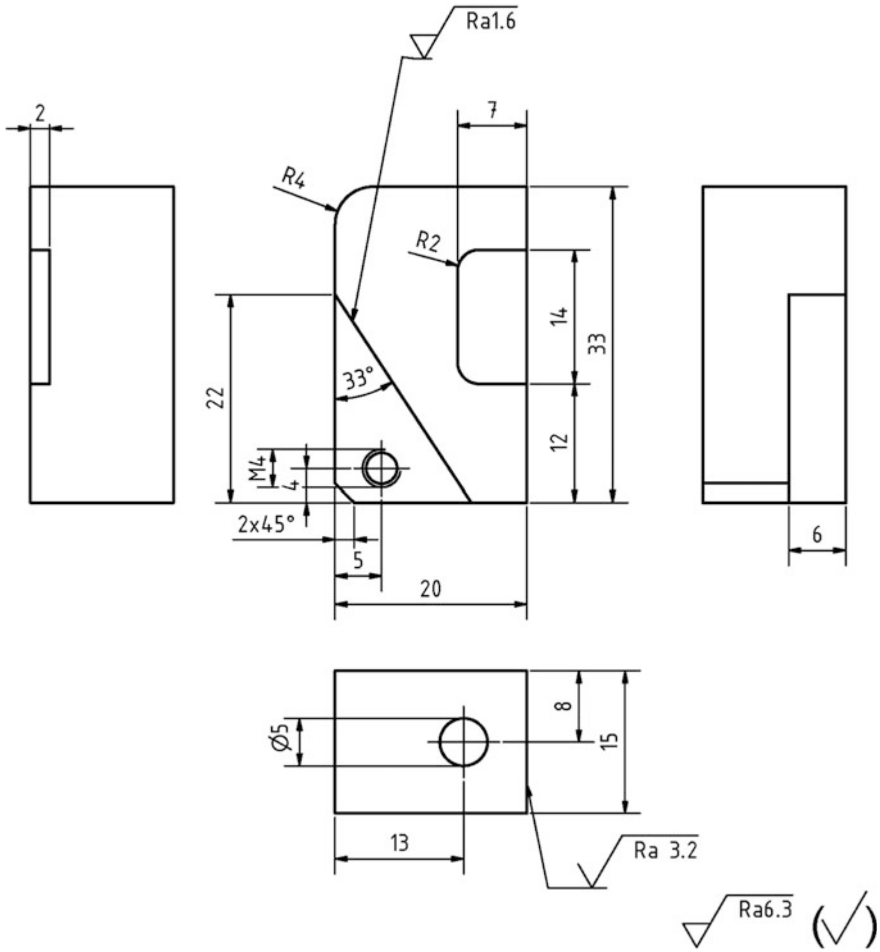


Fig. 2.13 Object in Fig. 2.10 with production-related dimensioning. There is no need to specify tolerance for the individual dimensions as general tolerances are specified in the title block ("ISO 2768–mK", not shown in the figure here). Solid surface specifications that specify an average surface roughness of 6.3 µm to be achieved by material abrasion are given *bottom right*. The bracketed expression indicates that there are also surfaces with different tolerances (individually labeled in the drawing)

drawing, followed by the production-related dimensions and the inspection dimensions. Inspection dimensions are typically used in inspection drawings only.

Every dimension should be quoted with a *tolerance value* as no part can be perfectly manufactured. General tolerances, which apply to all dimensions in the drawing and which are specified in the title block, should be preferred (see Fig. 2.13). Other options include (1) the ISO system of limits and fits which is a worldwide coordinated system of hole and shaft tolerances, and (2) manually specifying tolerance limits (see Chap. 8.2).

In recent decades, technical drawings have increasingly been prepared using computer technology with its CAD systems. Software programs for 2D drawings that provide different views of an object were initially developed, which essentially replaced paper-based drawings with digital ones. Today's CAD packages add new functionality to technical drawings; not only can objects be represented in 3D, they can also be optimized with comprehensive 3D modeling (Sect. 2.6).

Chapters 8.1 and 8.2 contain the most important instructions and guidelines for technical drawing; Chap. 8.3 contains preferred numbers and dimensions to be used in the design process.

2.5 Circuit Diagrams

Modern systems contain mechanical and electronic components. Circuit elements and their interactions must be represented in *circuit diagrams*, also known as *circuit schematics*. Development and design engineers need to understand circuit diagrams and know how to produce them. A circuit diagram is a symbolic description of a circuit showing the circuit elements and the links between them. The layout of the components in the diagram is not the same as the position of the elements and the connections in the actual implementation. The main purpose of the circuit diagram is to show the logical and electrical functions and interconnections of the circuit. It contains the following elements (Figs. 2.14 and 2.15):

- symbols
- device labels (ID letter with consecutive numbers),
- device type or rating,
- electrical connections (interconnects, bus systems),
- back-annotation data (optional), and
- frame and title block.

The *symbols* found in a circuit diagram can be either elementary symbols for resistors and other component symbols, or block symbols, e.g., subcircuits.

Each symbol has a unique *label* composed of a letter, which defines the component type, followed by a serial no. (*C4*—capacitor no. 4, *D1*—digital gate no. 1, *R12*—resistor no. 12). The component *type* (e.g., gate *74ACT04D*) or *value* (e.g., resistance 10 k) should be written beside the symbol as well. The engineering unit

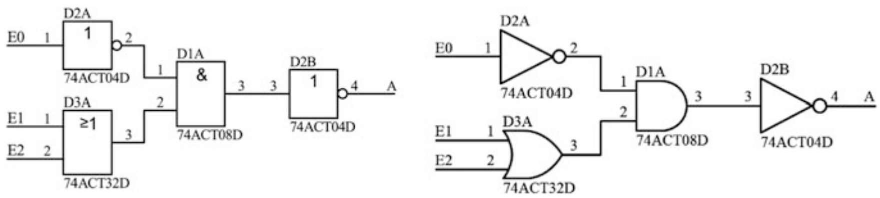


Fig. 2.14 Section of a circuit diagram with *digital* components (AND gates, OR gates, two inverters) in IEEE/ANSI/IEC standard format (*left*) and traditional schematics format (on the *right*). The letters *A* or *B* added to the labels *D1*, *D2,* and *D3* indicate copies (instances) of a library element. Elements *D2A* and *D2B* are copies *A* and *B* of inverter *74ACT04D* in the library in this example. Both gates are contained within the chip package *D2*

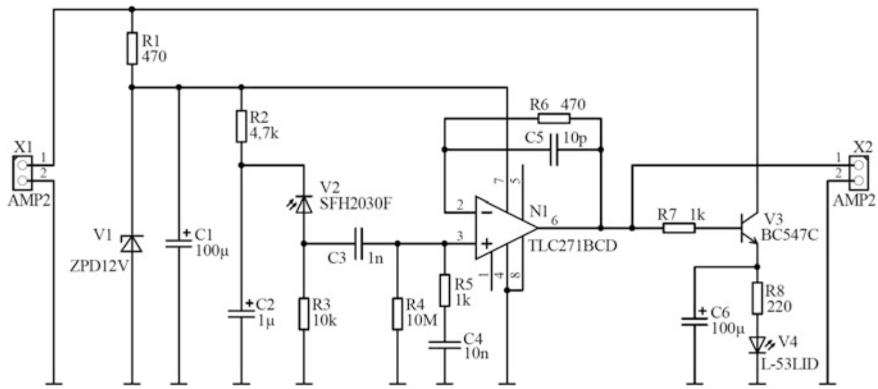


Fig. 2.15 Exemplary circuit diagram (with no frame or title block) with an operational amplifier and different *analog* components (resistors, capacitors, connectors, photodiode, Zener diode, LED, transistor). Each symbol is followed by a letter with a serial no. and the type and/or value of the component. The engineering unit (e.g., ohm or Ω) is not cited

(e.g., ohm or Ω) is not written with the values. *Pin numbers* are specified where necessary to avoid ambiguities (e.g., with integrated circuits, connectors) and when they are not already defined by the symbol (as with the transistor).

Digital elements are typically located in a common IC package despite being drawn individually in the circuit diagram. This assignment can be indicated in the schematic view. For example, *D2A* and *D2B* are two copies *A* and *B* of a type *74ACT04D* inverter, which are contained in package *D2* (Fig. 2.14 and Chap. 8.4).

The following *connections* are deployed in a circuit diagram:

- lines: of no electrical significance, for decorative purposes only, e.g., borders;
- wires: pin-to-pin interconnects of signal paths, signal name is optional; and
- bus systems: bundling many signal paths, signal name is obligatory, every signal has the same name and a different index.

Back-annotation data are details gathered during the design steps that follow circuit design, e.g., layout design, which you "write back" in the circuit diagram to consider them in the future. Current values for specific interconnects calculated in the layout could be inscribed in the circuit diagram, for instance.

A completed circuit diagram is used in the design process for electronic circuits to generate a net list. This net list, along with device information loaded from libraries and technology information, is then used to produce the implementation arrangement, the *layout*, for the integrated circuit (IC) or the printed circuit board (PCB).

Please refer to the end of the book (Chap. 8: Appendix) for more information on circuit diagrams and components. Nominal values for devices based on the preferred numbers are defined in Chap. 8.3. Chapter 8.4 contains a list of symbols for devices for use in the circuit diagram, and Chap. 8.5 introduces labeling options of electronic components (colors, characters).

2.6 Computer-Aided Design (CAD)

Computer-aided design (CAD) is the application of information technology to aid in the creation, modification, analysis, and optimization of a design. CAD features include:

- producing and using computer models for different disciplines (e.g., mechanics, electronics, heat transfer, fluid mechanics, magnetics) to find suitable and/or optimized solutions;
- deriving direct production information (e.g., rapid prototyping, CNC (Computerized Numerical Control) fabrication); and
- generating product documentation (e.g., technical drawings, production documents, manuals).

A *CAD model* is typically seen as a computer model of a device or a system that includes its geometric and material properties. That said, it can also reflect and optimize the different requirements that need to be considered in the design and development process of a device or a system (Fig. 2.16).

The CAD model represents the system design status at a specific time in the design process. The first version of a CAD model is typically produced at the end of the conceptual stage in the design process, modeling the geometric and material properties of the conceptual solution.

Additional numerical "special-purpose models" cover selected requirements based on the current state of the CAD model. Among these extra models are finite-element models for heat issues, structural mechanics, and electromagnetism as well as electrical and timing circuit models.

For many decades, computer-aided design has played a key role in the design of complex VLSI circuits with millions (and now, billions) of gates.

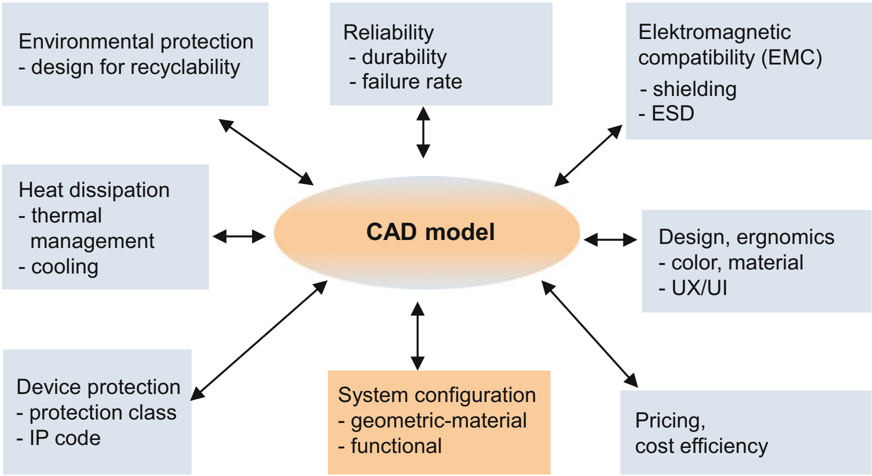


Fig. 2.16 CAD model at the heart of the system design process. Besides modeling the geometric, material and functional device/system configuration, other special-purpose models, such as finite-element models for optimizing heat dissipation, can be built for specific requirements in the design process

Geometric-material aspects of the electronic circuit are taken into consideration in the CAD model, e.g., as external interconnect and pad dimensions for the associated electronic devices. "Special-purpose models" form the basis for the following activities:

– producing a circuit diagram from a digital logic description,
– simulating and verifying the circuit function(s),
– automated design of the layout for the substrate (IC and PCB), and
– verifying the layout against the design constraints.

Simulation runs based on these "special-purpose models" are typically used to analyze the system for improvement with respect to individual requirements. The results of these analyses lead iteratively to an optimized CAD model for the system to be developed, whereby the geometric and material design characteristics are modified. CAD models thus assist the engineers in iteratively optimizing the technical solution throughout the design process.

We will show how a detailed analysis can be performed based on the geometric-material CAD model with an example of a simple mechanical rubber stop. The rubber stop consists of two carbon steel disks, with a rubber sleeve between them, acting as a resilient and damping element (Fig. 2.17).

The CAD model of this rubber-stop assembly is made up of the CAD models of two identical carbon steel disks and a rubber sleeve (Fig. 2.18). The relationship of the components to one another is described in the assembly model with parameterized assembly dependencies (e.g., "Insert" for parallel and concentric orientation of circular surfaces).

Fig. 2.17 Rubber-stop assembly made of a rubber sleeve with two carbon steel disks at each end

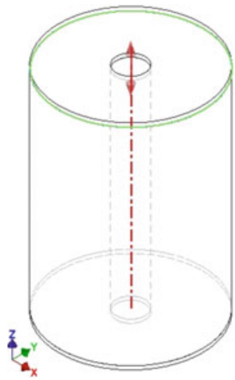


Fig. 2.18 Assembly dependencies in the CAD model of a rubber stop

Suppliers usually provide CAD models for purchase parts. The CAD models are available in the libraries of CAD systems for standard parts (e.g., screws and nuts). CAD models need to be generated for custom-designed components only.

The user can gradually produce three-dimensional (3D) geometrical models of parts from two-dimensional (2D) sketches with geometrical operations (e.g., extrusion or rotation about an axis). The sketched contour of the circular cross section is extruded to create this rubber sleeve model (Fig. 2.19). The model could also be created by rotating a rectangular sketch about the z-axis.

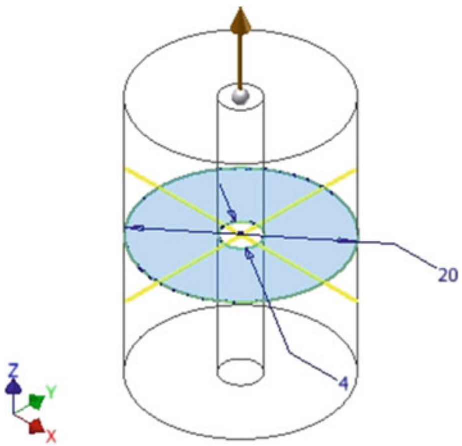


Fig. 2.19 Producing the 3D-geometric model from 2D sketches (volume of rubber sleeve)

In addition to this example, CAD systems feature the following options:

- Complex geometrical shapes can be created from basic elements with Boolean operators (e.g., union, difference, average).
- Standard form elements (e.g., holes, screw threads, bevels) are supported by features in CAD systems. All you need to do then is place a form element (e.g., bore hole) at the desired position and configure it (e.g., with a screw thread or counterbore).
- Dimensions used in the geometrical operations and form elements in the 2D sketches can be modified at any time in the design process. All dependent dimensions will automatically change as well, including the component arrangements in the assembly, for example.

In addition to utilizing these capabilities for creative design, development engineers still also use CAD systems for generating sets of drawings from the 3D geometric models. The required dimensioned, scale views can be semi-automatically generated in the part drawings (Fig. 2.20).

So-called exploded views can be produced as well. They are especially suitable for highly complex structures at the component level to illustrate the assembly of parts and subassemblies. Figure 2.21 shows an example of the basic rubber-stop assembly with a semi-automatically generated parts list.

Besides using CAD systems to produce a set of drawings, they are also applied in *finite-element simulations* (*FE simulation*) to analyze the mechanical characteristics of components. The finite-element models are automatically generated based on the 3D-geometrical models. The example in Fig. 2.22 shows the vonMises equivalent stress distribution in the rubber sleeve when external compressive forces are applied only on the hole edges of the carbon steel disks. We can determine where the maximum stress occurs by means of mechanical finite-element simulation. The shape of the part can thus be optimized at these critical locations based on these simulation results.

A dynamic simulation of mechanical assemblies can be carried out as well as the static loading on the parts with finite-element models. The behavior of elastic materials in real time can be investigated. This typically requires long computation times. Simplified analogous models based on network analogies are often used for this purpose:

- Body masses are modeled as point masses placed at the center of gravity of the real object.
- Body elasticities are modeled as springs with a parameter for the spring rigidity.
- Damping characteristics of bodies are modeled as damper elements with a damping constant.

The decay response of the oscillation after an excitation can be analyzed for the exemplary rubber stop in Fig. 2.17 with the following dynamic analogous model

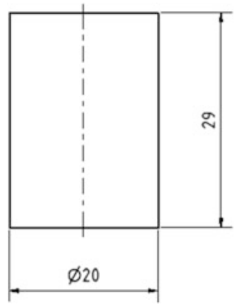
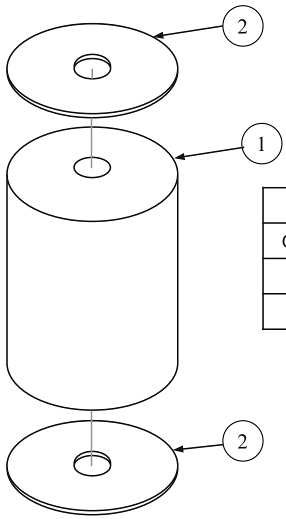


Fig. 2.20 Dimensioned drawing view of a part (rubber sleeve)



PARTS LIST			
OBJECT	QUANTITY	PART NUMBER	DESCRIPTION
1	1	Rubber 128713	Rubber sleeve
2	2	Disk 357518	Carbon steel disk

Fig. 2.21 Exploded view of an assembly (rubber stop) with parts list

(Fig. 2.23). The carbon steel disks are simplified as rigid elements and the rubber sleeves as massless elements. Some of the required concentrated parameters for the network elements, such as the masses of the individual objects, can be taken from the CAD model. Other parameters, such as the spring rigidity as a quotient of the pressure force and deformation, can be determined from the static finite-element simulations. The damping constant of rubber has to be obtained from the material data or the measurements.

The geometric and material properties of the CAD model are iteratively modified with the results of these simulations. The future characteristics of the entire

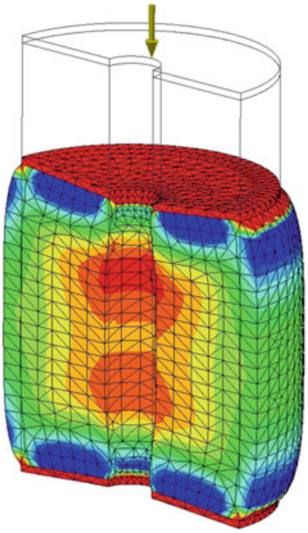


Fig. 2.22 A finite-element simulation where the stresses in the rubber sleeve are determined for compressive forces at the hole edges

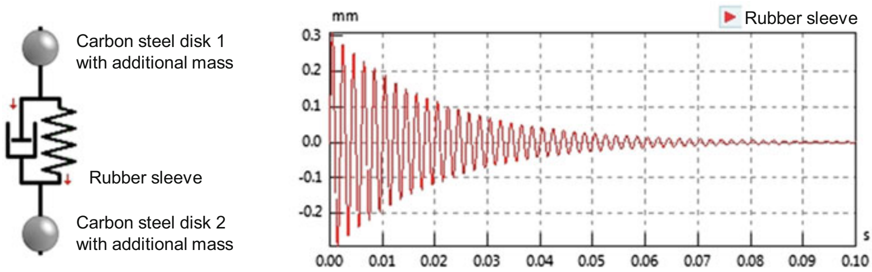


Fig. 2.23 Dynamic analogous model of the rubber stop in Fig. 2.17 with simulation results, showing the decay response of its oscillations after an excitation

electronic/mechanical system can thus be determined using the CAD models, which enables us to develop and simulate the desired optimal solution before undertaking its actual physical implementation.

References

1. G. Pahl, W. Beitz, J. Feldhusen, K.-H. Grote, *Engineering Design: A Systematic Approach*, Springer, 3rd edition. 2007
2. F. Kramer, *Innovative Produktpolitik, Strategie—Planung—Entwicklung—Einführung*, Springer, 1987

3. H. Kerzner, *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, Wiley, 11th edition, 2013
4. ISO 29845:2011, *Technical product documentation—Document types*, online at www.iso.org
5. ISO 128–30:2001, *Technical drawings—General principles of presentation—Part 30: Basic conventions for views*, online at www.iso.org
6. ISO 5455:1979, *Technical drawings—Scales*, online at www.iso.org
7. ISO 7200:2004, *Technical product documentation—Data fields in title blocks and document headers,* online at www.iso.org
8. ISO 128-40, 44, 50:2001, *Technical drawings—General principles of presentation—Part 40: Basic conventions for cuts and sections, Part 44: Sections on mechanical engineering drawings, Part 50: Basic conventions for representing areas on cuts and sections,* online at www.iso.org

http://www.springer.com/978-3-319-55839-4

Fundamentals of Electronic Systems Design
Lienig, J.; Brümmer, H.
2017, XIII, 241 p. 149 illus., 114 illus. in color.,
Hardcover
ISBN: 978-3-319-55839-4