OREGON STATE
UNIVERSITY
COMPUTER
SCIENCE
DEPARTMENT

Preliminary Results of an Experimental Study of the Mechanical Design Process

David G. Ullman and Larry A. Stauffer
Department of Mechanical Engineering
Oregon State University
Corvallis, Oregon 97331

Thomas G. Dietterich
Department of Computer Science
Oregon State University
Corvallis, Oregon 97331

86-30-9

Preliminary Results of an Experimental Study of the Mechanical Design Process

David G. Ullman
Larry A. Stauffer
Department of Mechanical Engineering

Thomas G. Dietterich
Department of Computer Science

Oregon State University
Corvallis, OR 97331

Abstract:

This paper presents the initial results from an empirical study of mechanical design engineers. In this study, six engineers were video-taped solving real mechanical design problems. Preliminary analysis of this protocol data has yielded several important findings, including (a) mechanical designers progress from systematic to opportunistic behavior as the design evolves, and (b) drawings and sketches are not merely tools for recording the final design, but instead they play a critical role in organizing the design process itself. The paper describes these and other findings and compares them to similar published data on software designers and to current theories of design methodology. The process of selecting the problems, taking the data, and reducing it is fully described. This paper is the first concerning this data and covers primarily the early stages of the design process and observations on the overall strategy of the subjects.

1 Introduction

There has been little effort to develop a coordinated approach to the study of the mechanical engineering design process. Consequently, few patterns have emerged that indicate, with any degree of certainty, how the effectiveness of mechanical engineering design may be improved. Additionally, to develop computer aided design (CAD) tools beyond their current abilities, an understanding of the human design process is needed. The study reported on in this paper is an effort to understand the mechanical engineering design process used by practicing engineers and from that understanding develop tools to improve the usefulness of CAD systems. The study consists of developing two design problems and video recording engineers as they solve those problems and verbalize their thoughts. This protocol analysis technique is heavily used in psychological studies and has been the basis for similar work in the area of software engineering. The recorded data have been transcribed and are now being analyzed to construct a detailed history of the engineering methods, knowledge, and heuristics employed by our subjects. From this history, we seek to identify opportunities and implications for the development of future design tools.

This paper is organized as follows. Section 2 describes our reasons for undertaking this study. Section 3 then presents an overview of the method of protocol analysis and presents the results of previous protocol-based studies of software designers. In Section 4, the exact methods that we have employed are described in detail, including the process of problem selection, data collection, and data analysis. Some particularly interesting sections of the protocol data are given to illustrate the methods. Finally, Section 5 lists our principal observations and assesses their implications both for design theory and for the development of AI-based CAD tools.

2 Rationale for the Study

There are four basic motivations for undertaking a study of the mechanical design process.

First, the process of mechanical design is not well understood. While there are many popular textbooks on mechanical design methodology (e.g., Love, 1980; Ostrofsky, 1977) as well as a significant body of more detailed work published in Europe (Hubka, 1980, (translation 1982); Pahl and Beitz, 1977 (translation 1984)), none of these books is based on any empirical data. These texts each encourage a systematic design strategy based on the development and elimination of parallel solutions to a problem or subproblem. They present theories of how mechanical design *should* be accomplished. However, no experiments have been performed to determine whether the methods espoused in these texts are in fact effective in producing good designs.[1] There are two other symptoms of the lack of understanding of mechanical design—namely, (a) mechanical design is difficult to teach and (b) there are no objective methods for determining whether someone is an expert designer. For these basic intellectual reasons alone, it is worthwhile to study the mechanical design process.

The second motivation for the study is the desire to improve the efficiency of the design process and the quality of the resulting designs. It has been conjectured that manufacturing productivity has increased many fold since the turn of the century but that design productivity has increased only slightly over the same period. Thus, the goal of understanding the process of mechanical design and, from that understanding, developing more efficient design methods, is seen as a critical research area (Rabins, et al., 1986).

The third motivation for studying human design engineers is our desire to apply and extend the methods of artificial intelligence (AI) to the development of intelligent CAD tools. AI tools show promise of being able to assist designers in many phases of the design process where existing CAD tools are inapplicable. For example, AI techniques may be useful during the conceptual design phase—a time when crucial design decisions are made. Another possible application of AI is to the problem of design history recording and replay. At present, the primary record of a design is the set of final drawings produced by the engineer. The various alternatives, trade-offs, and design constraints that contributed to the design are not recorded, and are therefore lost. This means that when it is necessary to modify the design or to design a similar device, the design must be reconstructed from scratch.

To explore and develop these AI methods, it is essential to study the methods employed by people. People are remarkably flexible and robust problem solvers, whereas computers are notoriously rigid and fragile. By studying human designers, we may be able to improve the flexibility and robustness of computer-based methods.

Finally, the fourth motivation for studying engineers as they solve real problems is that any CAD tool must provide a natural conceptual interface to the designer. If the engineer cannot understand what the CAD system is doing or how it is approaching the design problem, then the engineer is unlikely to make effective use of the system. Similarly, if the AI-based CAD system cannot understand what the engineer wants to do, it will be unable to provide assistance or record the decision-making processes of the engineer. In short, intelligent CAD tools must employ human-like methods if they are to be useful.

With these considerations in mind, we are conducting an empirical study of the mechan-

---

[1] The behavior of actual designers has received a small amount of study (e.g., Marples, 1960; Mitroff, 1967), but these studies have been informal and almost invariably based on a single subject.

ical design process based on the method of protocol analysis. The next section discusses this method of analysis and our reasons for selecting it.

3 The Use of Protocol Analysis in Engineering Design Studies

3.1 The Protocol Analysis Method

The technique of *protocol analysis* involves presenting a subject with a problem and asking him/her to "think aloud" while solving the problem (Ericsson and Simon, 1980). With a little practice, most people can get to a point where thinking aloud is virtually automatic. The goal is to obtain the subject's stream-of-consciousness thoughts, no matter how insignificant or incoherent the subject may think they are. This verbal protocol is then analyzed in an attempt to obtain a "trace" of the problem-solving steps taken by the subject. Through careful analysis, it is possible to develop a coherent explanation of the problem solving methods employed by the subject.

This approach is quite different from other approaches to learning about the design process. Consider, for example, the alternative method of *retrospective reporting*, where the subject explains, at some later time, what was done. Studies have shown that in retrospective reporting, people have a strong tendency to not report exactly what happened but what they *perceived* happened; there is often a major difference between the two (Ericsson and Simon, 1980).

Another alternative method is *informal reporting*, where an observer simply watches a person solve a problem and asks questions. Observational techniques of this kind, while they involve a minimum of disruption, only record what is written down or retained in other tangible form.

The chief advantage of protocol analysis, when compared with these other methods, is that it gathers all the information the subject can verbalize and communicate through drawings and gestures as it occurs. For many classes of problem solving, protocol analysis is recognized as being superior to these other methods of gathering data.

The protocol analysis method does have its drawbacks. One is that the requirement to "think out loud" tends to slow subjects down. Research has shown (Ericsson and Simon, 1980) that this does not affect the order or content of their problem solving steps. Hence, it was not considered to be an important drawback in the present study.

Another drawback of verbal protocols is that they cannot record what happens during incubation, when the subject "mulls over" the problem for a few days and thinks up ideas.

Psychologists are fairly divided over the issue of incubation in ordinary problem solving, and it certainly is a central question in creative design (Weisberg, 1986). This was a problem for the current study, and we attempted to control for incubation in our subjects (see Section 4.2).

A third problem with protocol analysis is that it is best suited to studying subjects solving problems by themselves. Engineers, however, often design in teams or at least pass their problems around for others to review. We could not think of any practical way of accommodating this aspect of design in the present study.

Protocol analysis methods have been applied to many problems in psychology and artificial intelligence. The method itself has been extensively studied and tested to understand its limitations. Ericsson and Simon summarize this research and conclude that protocol analysis does not make the problem solving process artificial and that it is the best known method for extracting someone's thoughts. As discussed in the next section, the method has been employed to study the design process in another engineering discipline.

3.2 Previous Applications of Protocol Analysis in Engineering

Two large-scale studies have been conducted that apply protocol analysis to learn about the design process in software engineering. One, by Adelson and Soloway (1984) studied professional software engineers designing an electronic mail system, and the other, by Kant, Newell, and Steier (Kant, 1985; Kant and Newell, 1982; Steier and Kant, 1985) studied graduate students designing algorithms for computational geometry. Adelson and Soloway also studied the difference between expert and novice designers working on familiar and unfamiliar objects and in familiar and unfamiliar domains. Similar results were found by both groups. Important among these are the following:

- The designers rapidly developed a kernel idea and refined it during the design process.
- The designers kept a current mental model of the state of the design that was transformed, as the design progressed, from abstract to concrete.
- The designers made a strong effort to keep the development of the design balanced. They would focus their effort on parts of the design that were most abstract; attempting to keep all parts of the design at the same level of detail.
- They spent about half of their time simulating the behavior of their programs. The simulation process served many functions: it helped the designer integrate constituents from several parts of the design; it served as a kind of agenda to keep

track of subtasks that require attention; it encouraged a kind of balanced, methodical refinement of the software system; and it allowed for comparison to the design goal. The design goal in software design is, in essence, a data flow behavior. In these studies, simulation means modeling the data flow through some part of the program at the current level of abstraction. This is the "working model" of the current design. Thus simulation allows the behavior of the current state to be modeled so that a comparison with the goal can be made. Kant also reports that simulation helps the designer identify interesting opportunities for improving the design.

- The designers took both mental and written notes on things to remember later in the design. These included constraints, partial solutions, potential inconsistencies or other concepts that arose during the design process. These were not handled immediately, since they were at a greater level of detail than the current state of the design. Note-taking was not observed if the designer was working in a familiar domain but on an unfamiliar object.

It was the success of these previous studies that encouraged us to undertake the present study. In the sections below, we describe our observations and compare them with the results of these software engineering studies.

4 Experimental Procedure

Now that we have presented the motivations for studying human mechanical designers using the method of protocol analysis, we turn our attention to the actual procedure that we followed. We have included substantial detail about our efforts, as there is little in the literature about such studies and we want to establish a clear picture of what we did as a basis of comparison for future studies.

4.1 Selecting the Design Problems for the Study

The two problems that we have developed were selected with several goals in mind. First, we wanted to observe all phases of the design process. Hence, we provided our subjects with incomplete, high level specifications, and we followed their progress until they produced detailed working drawings for at least some parts of the final design. Second, we wanted to explore the difference between product designs and one-off designs, because the constraints on the problem of designing for a product are much different from those for designing a one-of-a-kind device. We have achieved this goal by developing one product-oriented problem and one "one-off" problem. Third, we wanted to explore the relationship between the engineer's knowledge and skills and the requirements of the problem. To achieve this, we have taken protocols from graduate students with limited design experience as well as from experienced mechanical designers. We have also selected problems for which our experienced subjects can be expected to have a high degree of expertise.

Appendix I of this paper presents the two design problems that we have used. One, which we call the "flipper-dipper," involves designing a machine to grasp and position a thin aluminum plate onto the surface of a water bath. The machine must dip both sides of the plate, one at a time. This problem requires simple knowledge of kinematics and some actuation technology, such as manual manipulation, pneumatics, or small electromechanical transducers. It is a "one-off" problem, since only three of these machines are to be constructed. This problem is based on a consulting contract completed a number of years ago by one of the authors.

The second problem is product-oriented. It involves designing the contacts and compartment for the batteries in a small portable computer. The contacts must be designed so that a robot can easily install them in the computer during assembly. This problem requires knowledge of metal springs, molded plastic materials, and robot assembly constraints. Over the expected lifetime of the product, approximately 1.8 million units will be produced. This problem was developed with the cooperation of a major computer manufacturer.

Each problem was designed to take about 10 hours to complete. This is extremely long by the standards of ordinary protocol-analysis studies[2]. This length of time was dictated by the need to use realistic design problems and by our desired to obtain data on all phases of the design process. Ideally, even longer protocols would be taken, but these 10-hour protocols present formidable data analysis problems, and longer protocols would be unanalyzable in practice.

The size of these design problems forced us to take the protocol over a series of separate sessions. To account for the breaks and the potential incubation that might occur during this time, we had the subjects summarize the state of their design at the end and beginning of each session. This is discussed further in the next section.

We presented the problems to a total of six subjects, three subjects for each problem. Of the subjects, four are experienced, practicing design engineers in areas that are similar to the problem they were assigned. Two experienced engineers were assigned to each problem. The other two subjects were graduate students who had some limited design experience. There was one graduate student assigned to each problem. The distribution

---

[2] The software design studies, for example, were based on sessions lasting only two hours.

Table 1: Distribution of Subjects

Problem	Grad Student	Professional
battery contacts	S1	S2 and S3
flipper dipper	S4	S5 and S6

of subjects is shown in Table 1.

4.2 Protocol Data Collection

To record only the verbal protocol for a mechanical design engineer would be to miss all of his hand gestures, sketching, and pointing to earlier sketches. Thus, the protocols were recorded on video tape to capture all data possible. The camera was focused on the subject during the initial conceptual stages of the design when there was much gesturing and little drawing. Later, the camera was aimed directly down on the engineer's sketch pad as the problem solving became more detail-oriented. The camera was refocused as needed during the protocol in an effort to always capture the major communication of the subject. The visual behavioral report was used with the verbal report to better understand the subject's thoughts.

The problems used in the protocols and reproduced in Appendix I were refined through the following process. First, they were used in a short, one hour session, where a graduate student subject (not one of our six experimental subjects) read the problem and developed a rough conceptual design. The purpose of this was to test the problems to make sure that all essential information was available and to identify any aspects of the written description or the initial sketches that might confuse the subject. We also wanted to see what kind of questions the subjects might ask, so that the examiner could be prepared to give good answers during subsequent data taking.

The second step was to take a full 10-hour protocol for each problem from a graduate student subject with limited industrial experience (subjects S1 and S4). The purpose of this step was to collect data on less experienced subjects and to make sure the problem statements were clean before taking data from the practicing engineers. We found no major problems at this stage of the problem development.

After refining the problems by this procedure, we then administered them to experienced engineers who are presently working as designers. Each subject's background was matched to the problem as closely as possible.

All protocol sessions for the professional engineer, "expert" subjects, were conducted at the subject's place of employment. This was desireable because all of their reference materials were there and the protocol sessions involved a large time commitment on the subject's part. Once at the plant, we would meet in a convenient conference room instead in the subject's office. None of the engineers had a private office, and conducting the protocols in a public setting would have been too distracting and inhibiting.

In the conference room, a video tape recorder and camera (and a back-up audio recorder) were set up for use. All equipment was set up and tested prior to the subject's arrival in order to deemphasize it's presence and help the subject relax. Most of the subjects forgot the camera was there after working a while. This was evidenced by comments from the subjects (e.g. "Oh, is that still on?" at the end of a session) and the perception of the examiner. The main considerations were to get the microphone close to the subject without getting in the way and keeping the equipment controls turned away from the subject so he/she wasn't distracted when tapes were changed. An audio recording as well as a video recording was made as a backup. It helped to have two audio tape recorders to prevent delays during tape changes.

Interruptions were not always avoidable, such as those between sessions. A subject could verbalize his/her thoughts for two, sometimes three hours. After this time, fatigue set in, and the subject lost concentration and became too tired to continue. Since the solution time for each problem was about 10 hours, the protocol sessions were conducted over the course of 2–4 days depending on the schedule of the subject. At the conclusion of a session, the subject was encouraged not to think about the problem until the next session. Though they all claim to have made an effort not to think about the problems, some thoughts inevitably took place. To help us account for these thoughts, the subjects were asked to summarize their designs at the end and beginning of each session. In this way, we attempted to detect any problem-solving that may have occurred outside of the protocol sessions. To prevent further loss of information, the sessions were held on consecutive days. To expect the subject to avoid thinking about the problem for more than one day seemed unrealistic.

During the entire protocol session, the examiner was in the conference room with the subject. The purposes for this were to monitor the video and audio equipment, change the tapes, to operate the video camera (zooming in and changing positions when necessary), and to answer the subject's questions. The subjects always had questions about the problems that could not be anticipated ahead of time. The subjects were sometimes unsure about the scope of the problem. The examiner tried to keep them from delving into the fringe areas. It is extremely important for the examiner to have a technical background, because many of the questions that arose where technical in nature (the examiner was a mechanical engineering Ph.D. student with 4 years of industrial design experience).

Another function of the examiner was to keep the subject verbalizing. Sometimes the subjects would get so involved in their thoughts that they would forget about the experiment and not verbalize. The examiner would simply say "keep talking please" and nothing more.

At the beginning of the first protocol session, the examiner presented the subject with a set of written instructions (Appendix 2) to ensure consistency among the subjects. The subject read them to himself/herself, and then the examiner answered any questions about the verbalization process.

The first session continued with the presentation of a practice problem to get the subject accustomed to thinking aloud, check out the equipment, and generally put the subject at ease. Other researchers have used practice problems such as hands-on model building exercises (Bailey and Kay, 1986). We wanted a realistic problem that would be similar to the type of problem they would be solving for the protocol data. We also wanted a problem that would require no rigorous analytical treatment but be solvable with only basic mechanical engineering knowledge. A problem was developed that required the subject to design a toilet bowl for a mobile home with unique space requirements (see Appendix III). All the subjects felt somewhat knowledgeable about the task and therefore could work the problem with confidence. After working on the problem for 10–20 minutes, the examiner had the subject abandon the solution when the verbalization technique seemed routine. The subject was encouraged on his or her performance, suggestions made, and questions answered. Then the real problem solution was begun.

The subject was given the written problem statement and asked to read it out loud. This was requested to reinforce the verbalization needed in the remainder of the sessions. The subject was encouraged not to rush but to work as naturally as possible. All subjects worked with a sketch pad. One subject even brought in drafting supplies and paper to construct a full size drawing of his machine. This type of behavior was encouraged, if it made his problem solving in this experimental environment more similar to his everyday work environment.

Two of the engineers occasionally used CAD systems in their everyday work. We decided not to let them use their systems in this study as we did not know what effects CAD would have on their problem solving process and we did not want to introduce another variable. This is an area for further study. Additionally, all of the workstations were in public places and were therefore not conducive for taking data. It was not practical to move the systems to the conference rooms. Neither of the two engineers said they did the bulk of their work on CAD, as both had only begun using CAD within the last year.

We supplied the subjects with pen and paper for their sketches. They often complained about using pen, but we didn't want to loose information that was erased. The pages were numbered so that we could keep track of them while analyzing the protocols. It is essential to have copies of the sketches while watching the video tapes, as a large amount of the protocol data is in the drawings.

4.3 Analysis of the Protocol Data

In analyzing the protocols, it is important to have a realistic perspective of what the protocol data actually contains. The protocols give a verbalization of a portion of the contents of the subject's short term memory during the design procedure (Newell and Simon, 1972). The subject can only verbalize one thing at a time and thus cannot verbalize all that he/she is thinking. Additionally, everything that is verbalized is not necessarily pertinant to the design. In Figure 1, we show a representation of the relationship between the engineer's problem solving effort and the protocol itself. Some of the information in the protocol is irrelevant ("I'm ready to take a break now") and can be easily eliminated. What is left we refer to as the relevant protocol, that which captures a portion of the subject's problem-solving effort.

The engineer's problem-solving effort not only contains a description of the form and function of the mechanism being designed but also other information such as the engineer's strategy and agenda (his/her design process). Statements such as "I think I'll look at a side view first" don't correspond to any change in the evolving mechanism, but certainly give us an indication of the subject's design strategy. So we view the subject as having two very basic types of thoughts: those which pertain to the form or function of the mechanism being designed and those that refer to the problem solving process itself. With the protocol anlysis method we have captured part, but not all, of the form, function, and process via the subject's verbalizations. These are are represented by the hatched region in Figure 1. The remainder of the design process must be inferred. But, as shown below, careful analysis of the protocol can give meaningful results with minimal inference.

4.4 The Coarse Breakdown

The first step in analyzing the protocols was to develop a *coarse breakdown*. This is done by watching the video tape and reading the transcript at the same time.[3] This breakdown was undertaken to understand the flow of the protocol, to familiarize us with the design,

---

[3] Each audio protocol was transcribed immediately after the protocol sessions. This is a very time consuming process, typically requiring 70 man-hours per protocol to type and correct.

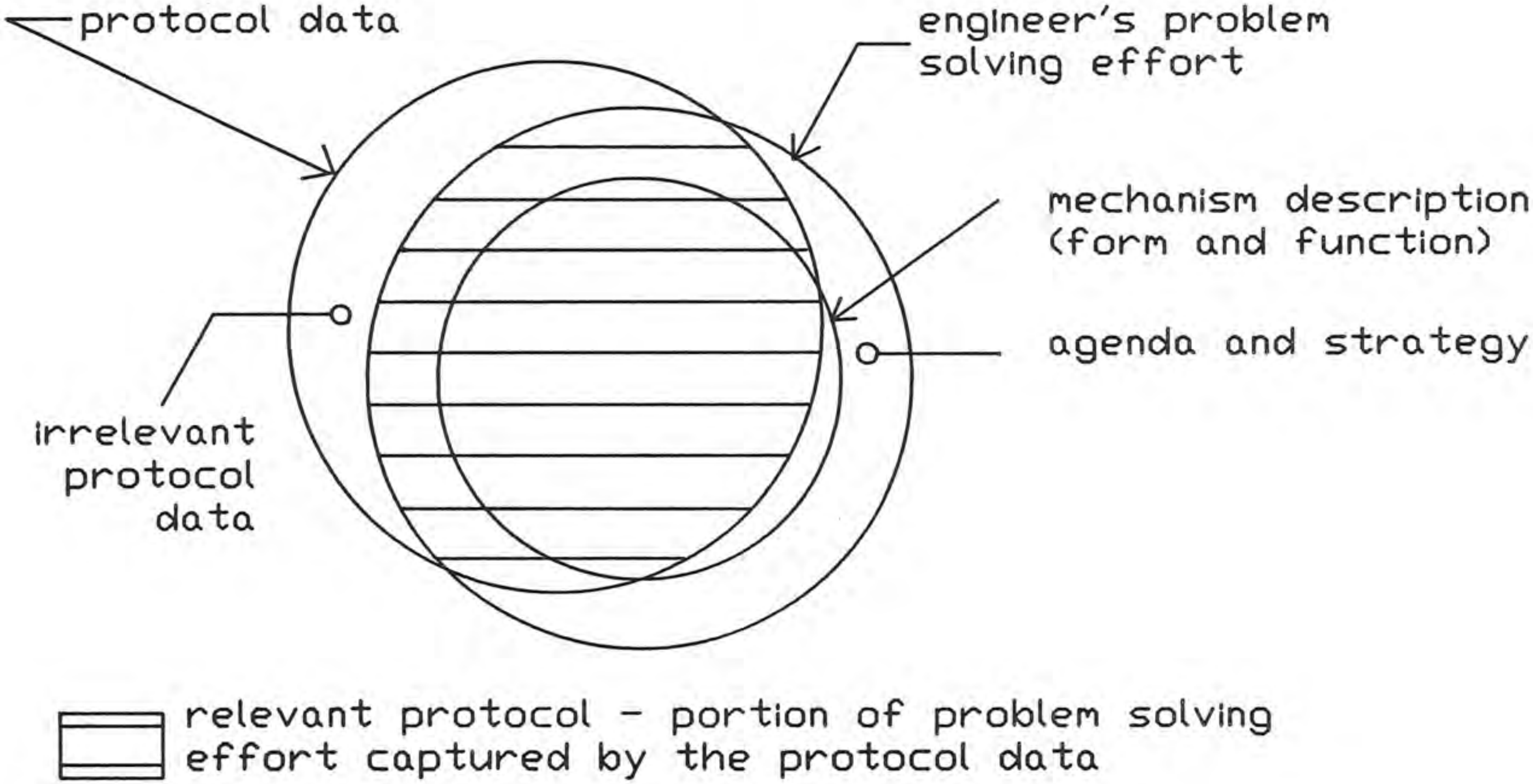


FIGURE 1. Model of Relevant Protocol Data

and to identify interesting areas for further analysis. Our general procedure was to have two researchers independently prepare coarse breakdowns and then compare and combine their results for a final version. With practice we were able to obtain a 75% agreement between researchers. Sometimes it took 3 or more viewings of a specific section of the protocol by each researcher, followed by vigorous discussion to identify what the designer was doing at that time. This kind of detailed analysis is not permitted by nonrecorded observational techniques such as direct observation and anecdotal data collection.

We have tried various notations for representing the coarse breakdown in an effort to capture both the chronology of the design and the development of form and function. Many researchers have used a tree to represent the design process (Pahl and Beitz, 1984, Newell and Simon, 1972, Mitchell, Steinberg and Shulman, 1984). We have found the mechanical design effort, complete with form, function, agenda, strategy, and chronology, to be too complex to represent in such a simple manner. A stand-alone tree of the final form or of the function does not convey the chronology and the complex interrelationships of design decisions sufficiently. The final result of this search for a notation is given below for two of our subjects' designs.

A coarse breakdown for subject S6, an experienced engineer solving the "flipper dipper" problem, is shown in Figure 2. The three main headings in the figure correspond to the three basic functions that needed to be considered (as they were conceived by this subject): "operation," "power," and "environment." The vertical axis represents time, and it advances down the page. The protocol is divided into a sequence of events. In each event, the subject is focused on some entity: a form, a function, or a strategy for carrying out the problem-solving process. Each event is represented in the figure as a block. Blocks with rounded ends represent events in which the subject worked on some "function." Rectangular blocks represent "form" events. Blocks with triangular ends designate events that dealt with neither form nor function, but instead with the "strategy" of the problem-solving process or with identifying new "agenda" items to consider. The numbering of the blocks gives the page and line number in the transcript where the event appears and thus, the sequence of events.

As in all the protocols, S6 began by reading the problem statement. (He spent 11 minutes here.) After gaining an understanding of what was required, he began to think in terms of the three main functions: "operation," "power," and "environment." The "operation" function was subdivided into two secondary functions: "input and output of the machine" and "handling of the plates." The subject had no form for the machine in mind at this time, but was concerned with the functions that the mechanism needed to perform. After addressing each of these main funtions, he began concentrating on the

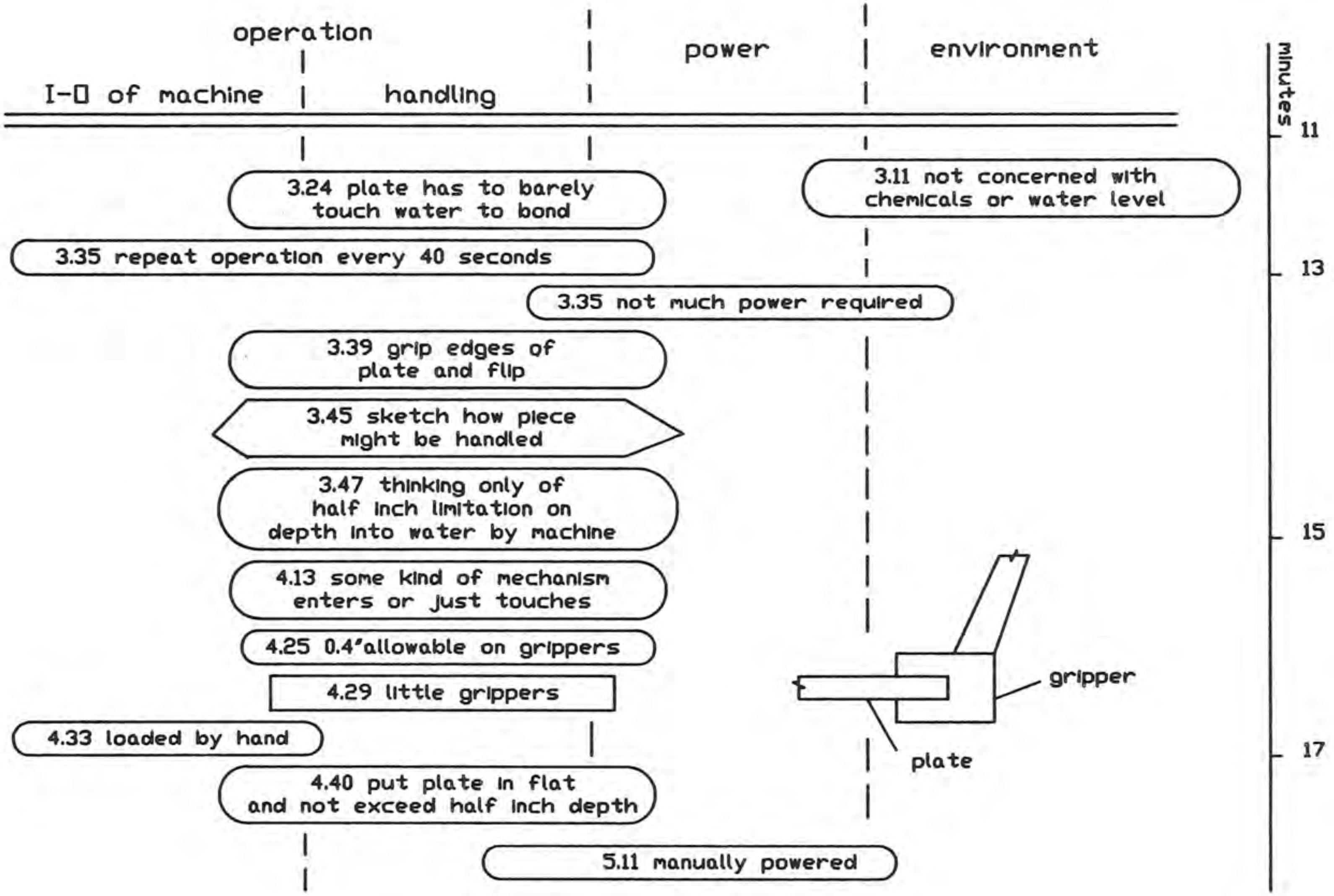


Figure 2. S6 Functional Coarse Breakdown

function of handling the plates. In event 3.45 he began to develop his first form when he states "So, basically, I'm going to just sketch mechanically how the piece might be handled." He talked vaguely in event 4.13 about "some kind of mechanism" for handling the plates and eventually, in event 4.29, he sketched his first form, which he called "little grippers." These were still very conceptual and nebulous. After this brief functionally-oriented thought process (represented in Figure 2), the subject sketched his first conceptual design that would perform the functions he had been considering.

Once S6 made his first conceptual design, he never considered other ideas. His original design remained and evolved into a final detailed design. After the first conceptual design was developed, the subject generally organized his thoughts around individual components (forms), not around the functions shown in Figure 2. To reflect this change in focus that occurred at the time of the first conceptual design, we have re-arranged the columns of the coarse breakdown so that they now denote forms instead of functions. This can be seen in Figure 3, which shows a later section of the S6 coarse breakdown. In this form-oriented scheme, a function sometimes spans two or more columns when the function includes more than one form (e.g., 18.45 and 20.31). A form block can also span two or more columns if it was later evolved into different forms. However, this is not seen in the example and rarely happened in any of the protocols.

At this point in the problem solving session, S6 was still finalizing the relationship between the forms. The figure begins with the subject stating a function that will eventually affect several forms (event 18.35): "it would only take an altitude of about 8 inches to clear the water and to rotate the part, then come down and register on the sides of the tank ..." He then developed refinements to pre-existing forms to achieve this function. Configuration A in Figure 3 is a graphical representation of the mechanism at this point.

Next, he restated the need to raise the plate to clear the water bath and rotate it, but this time in event 20.31 he quickly calculated a distance of 5 1/2" instead of 8" as in event 18.35. In actuality, the 5 1/2" distance would cause physical interference in the machine's operation. The erroneous constraint was established as a new function. He then generated Configuration B in Figure 3, which basically switches the positions of the support and the spring. It appears that the 5 1/2" constraint serves as a trigger for this second design.

Rather than analyze both configurations to a greater level of detail, he simply chose the second idea over the first (for apparently minor reasons, event 20.41). He never again used the 5 1/2" distance for raising the plate and eventually designed a machine that raised the plate 8" above the water bath.

The protocol for subject S5, another expert engineer working on the same problem, was analyzed in the same manner (See Figure 4). While S6 arrived at his first conceptual

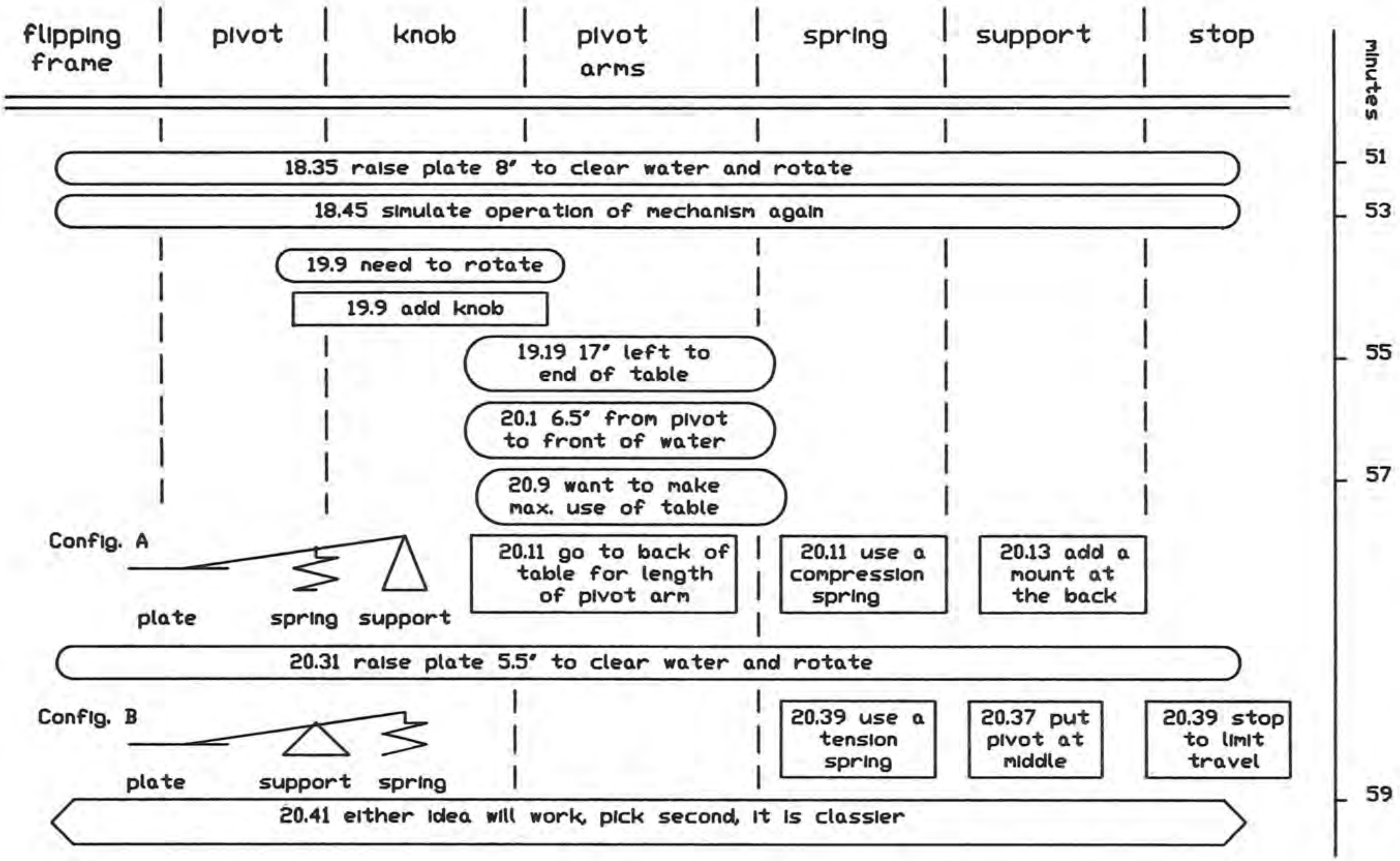


Figure 3. S6 Coarse Breakdown – Form Selection

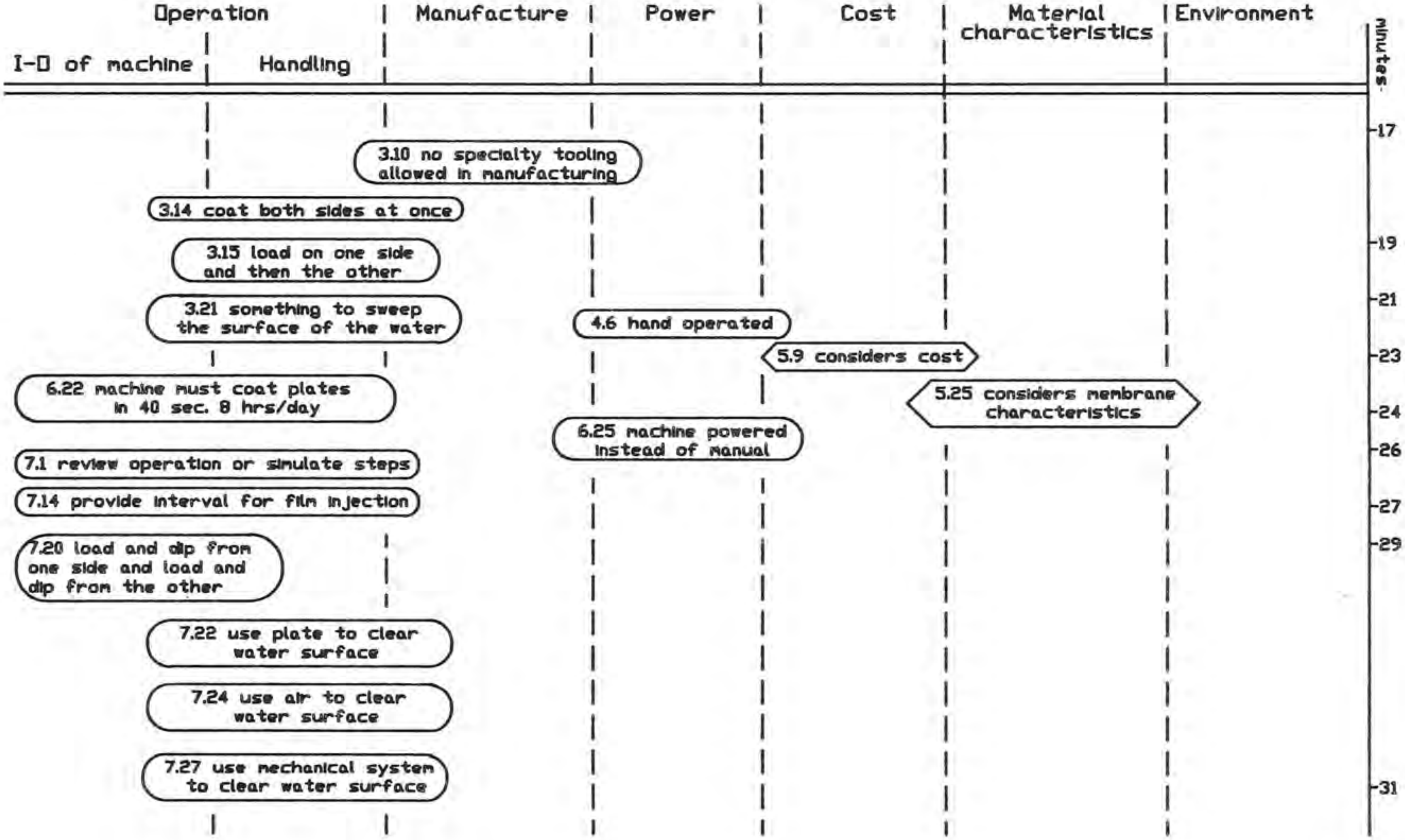


Figure 4. S5 Functional Coarse Breakdown page 1 of 3

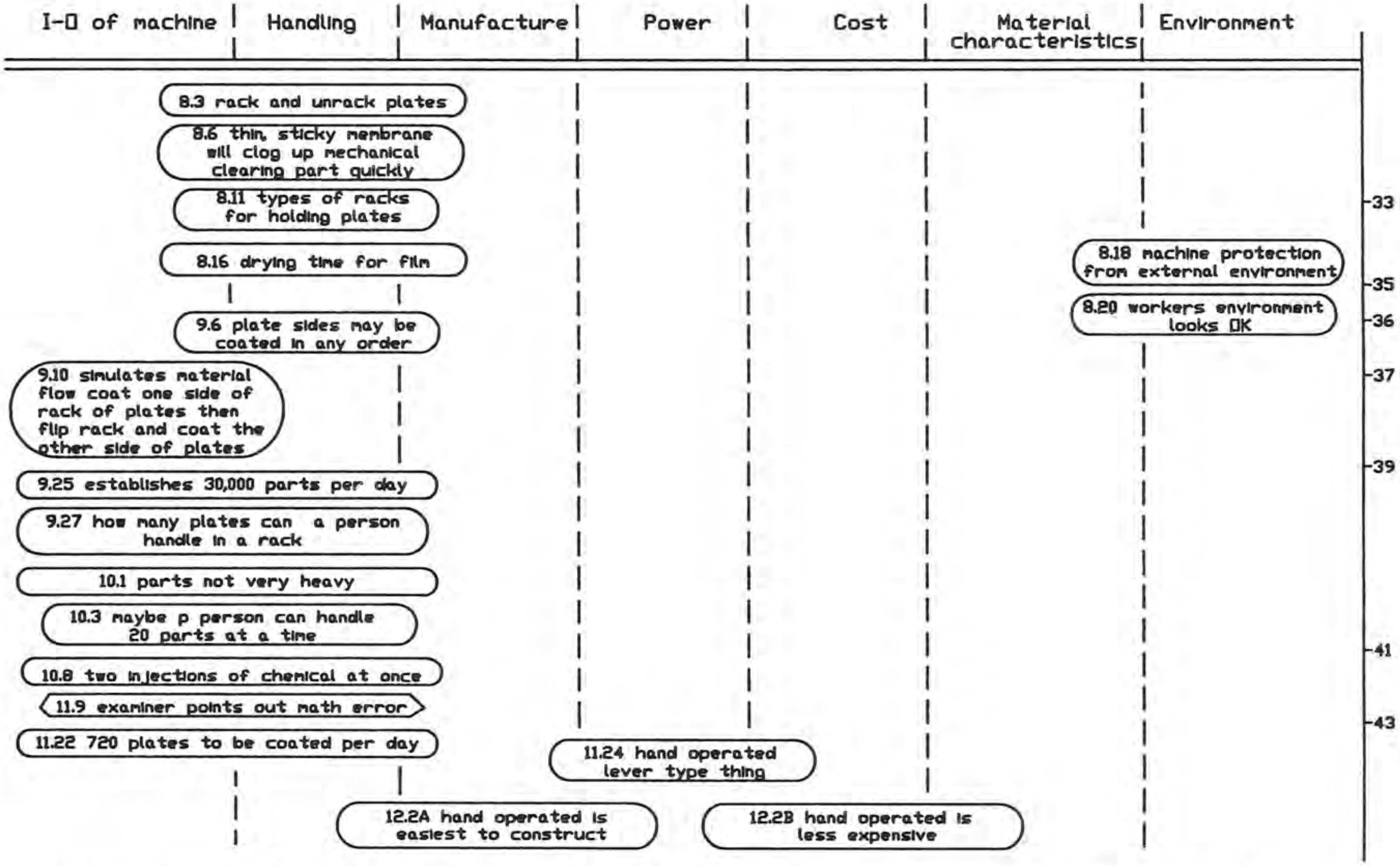


Figure 4. S5 Functional Coarse Breakdown page 2 of 3

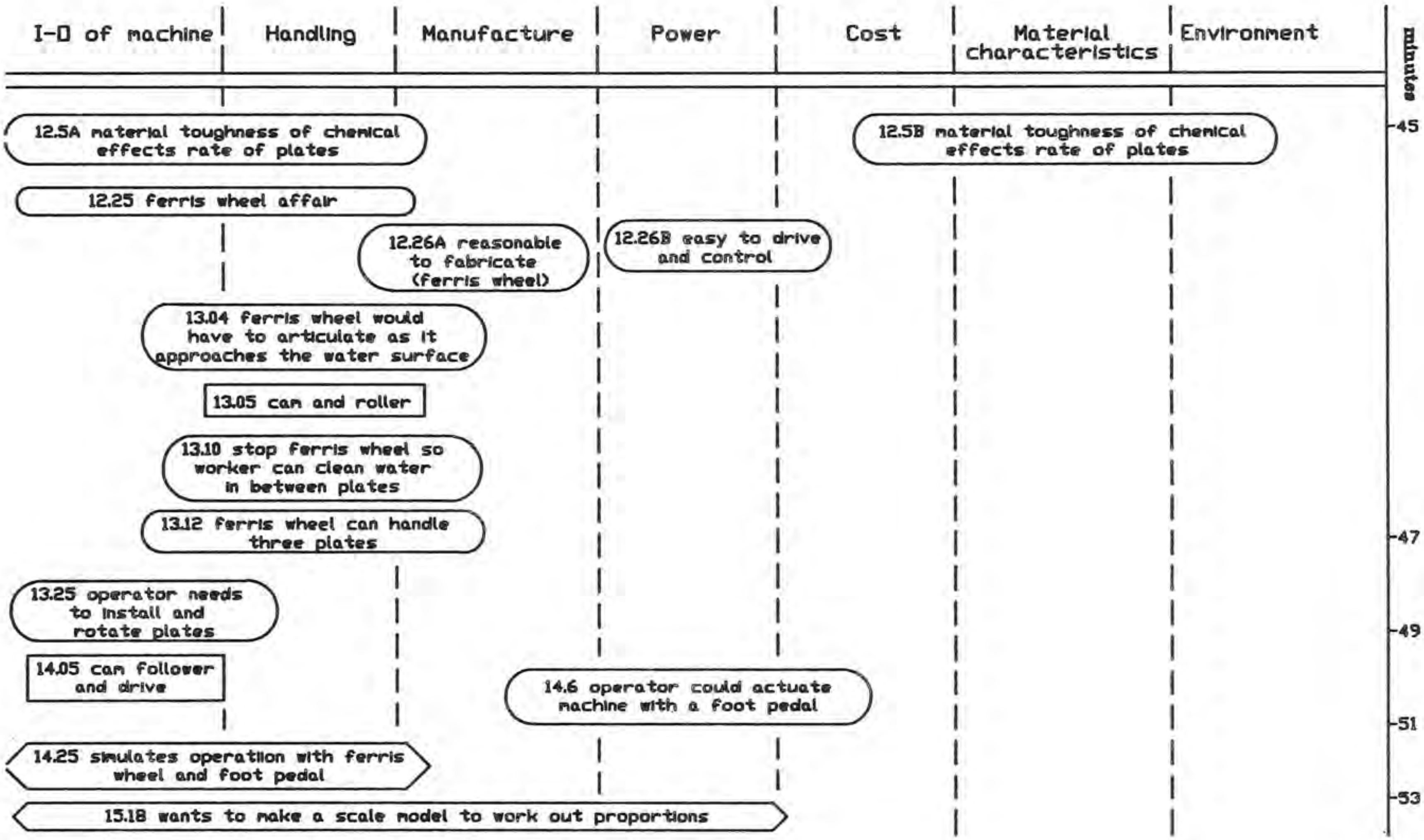


Figure 4. Functional Coarse Breakdown page 3 of 3

design in 20 minutes, S5 required 42 minutes, largely because he considered several more functions at the beginning of his effort. In addition to the "operation," "power," and "environment," S5 also considered "manufacturing," "cost," and "material characteristics" before developing a first conceptual design. In these early stages, S5 spent time considering all these functions, before concetrating his efforts on the operation of the machine.

At this stage, the subject also introduced several peripheral functions that we never intended him to address. Two of these were (a) how to clean excess chemical from the waterbath (events 3.21, 7.22, 7.24, and 7.27) and (b) how to handle the plates after coating (events 8.3 and 8.11). Both of these problems and several other minor concerns would be important in actual design, but the examiner told the subject not to worry about them in order to keep the problem-solving sessions to a manageable length.

One point of interest consumed nearly five minutes of the protocol. S5 became concerned about the number of plates that had to be handled in a day. He calculated a quantity of 30,000 plates per day in event 7.25. Concerns about this large quantity led him to consider elaborate ideas such as making the chemical coating a continuous process or coating them several at a time. Eventually, the examiner pointed out a math error to the subject, and the quantity was recalculated at 720 parts per day in event 11.22. He then realized the problem would not require such an eloborate solution and proceeded.

Even though the large quantity requirement was the result of a math error and the subject knew it, a machine was designed that could handle three plates at a time to unnecessarily speed up the process. In event 12.25, he stated "...the idea just came across my mind to use, kind of like a ferris-wheel affair..." which would make the plate dipping process semi-continuous. It is evident by this and other examples that he never fully forgot the result of his math error. After he mentioned the ferris-wheel-type process, his thoughts became very focused as he developed this idea further. For the next eight minutes, he imagined how the operation of a ferris wheel could be applied to his problem and added forms such as a cam and roller in event 13.05 to cause the plate to articulate as it approached the water surface. By event 15.18, the subject has arrived at a conceptual idea and his thinking becomes more form-oriented.

4.5 The Fine Breakdown

Interesting sections of the coarse breakdown were analyzed to produce "fine breakdowns." The fine breakdowns follow the same format as the coarse breakdown except that we considered every utterance the subject made. While most of the information was in the protocol, some things needed to be inferred to fully understand what the subject was thinking.

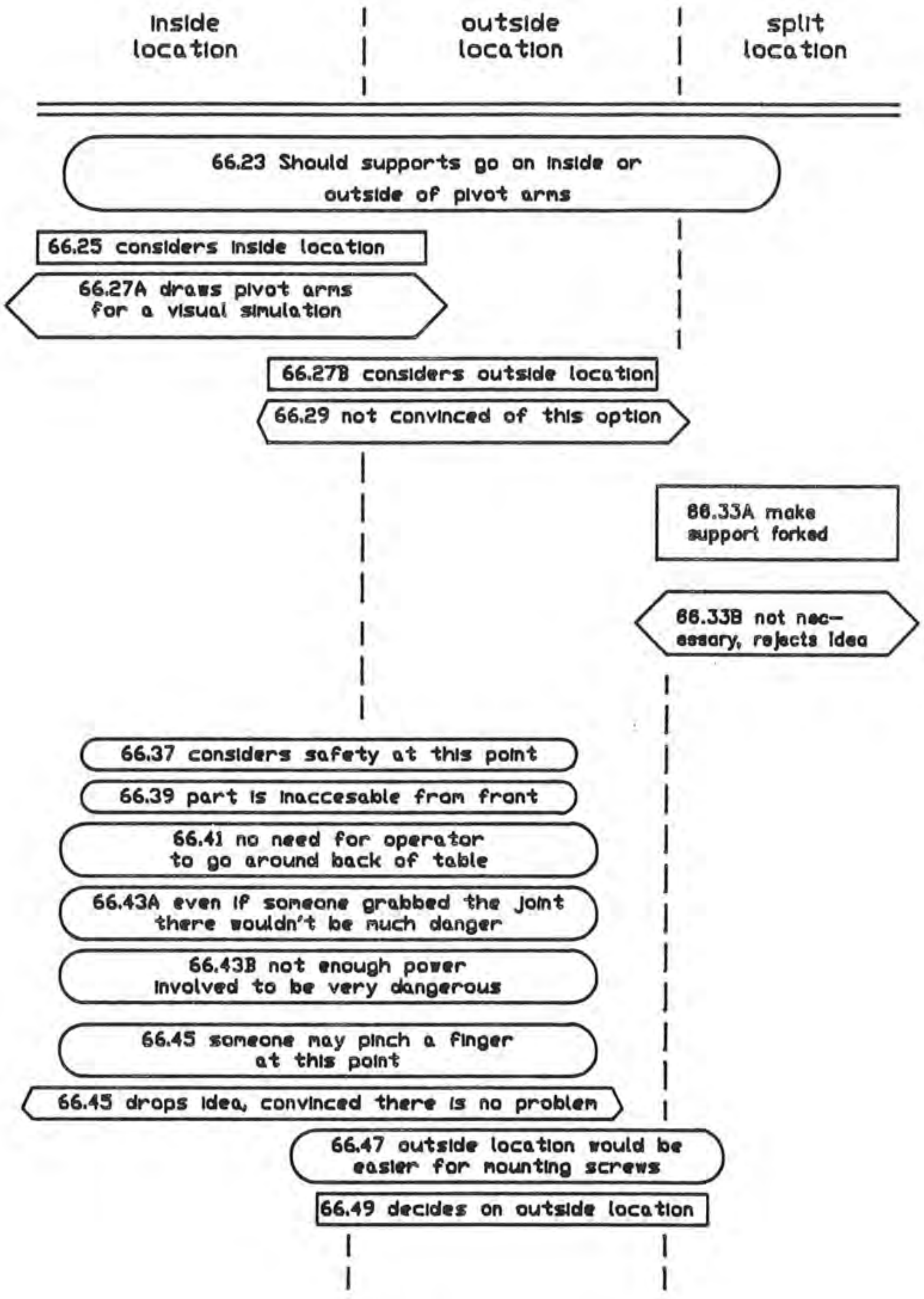
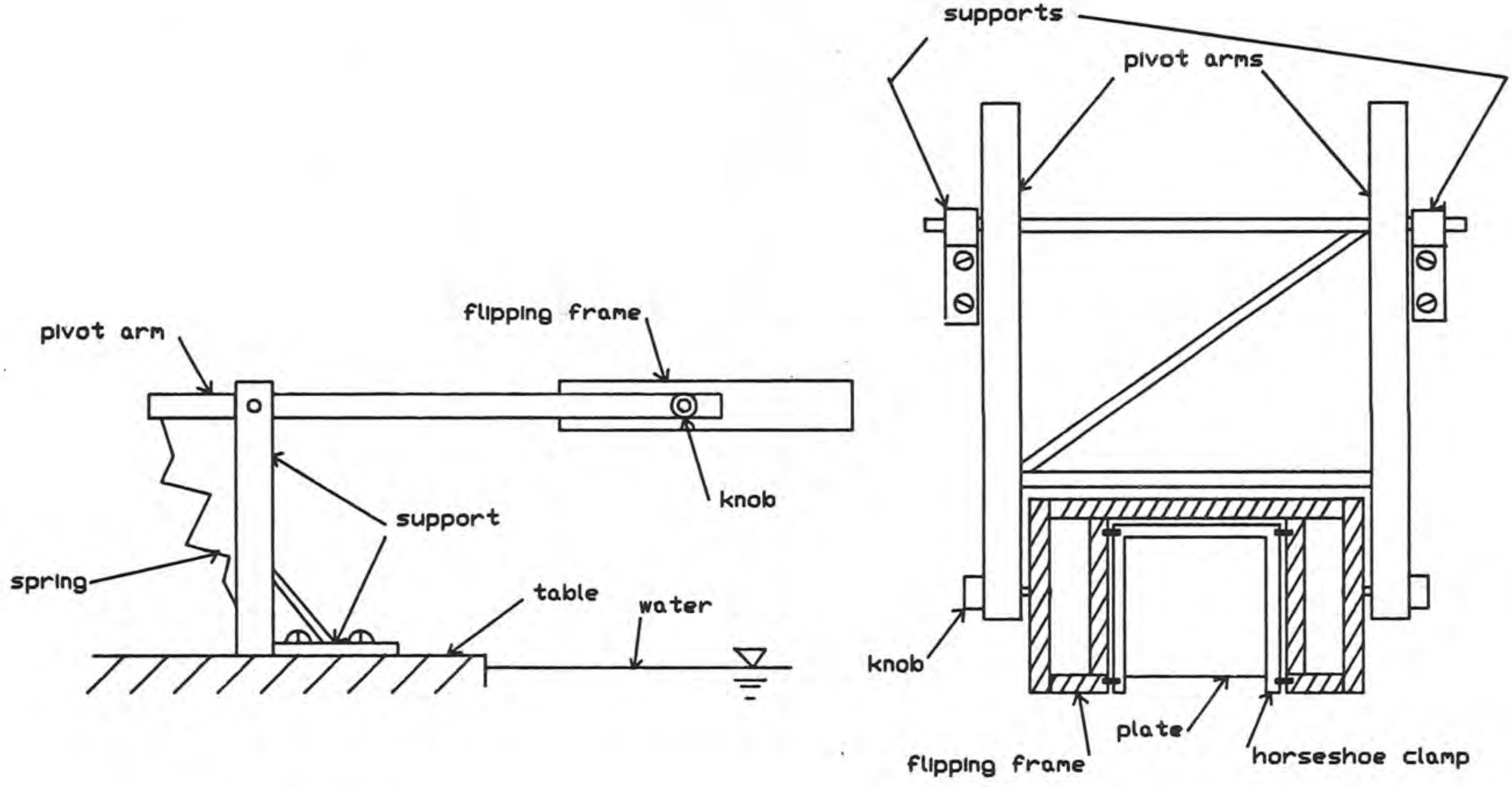


Figure 5. S6 Fine Breakdown



side view

top view - rotated 90 deg

Figure 6. Flipper–Dipper Design by S6

The fine breakdown for a 1.5 minute section of S6's protocol is shown in Figure 5. In this section, S6 was trying to locate the support for the pivot arms (See Figure 6). The subject began by stating his goal in event 66.23, "do I want to put them [the supports] on the inside or outside [of the pivot arms]?" He initially considered the inside location but was not sure, so he sketched the pivot arms to "see" his options. In event 66.27B, he considered the outside location but was not convinced of that either. In event 66.33A, he went beyond his stated options and considered a forked support—a sort of compromise that supports the pivot arm on both the inside and the outside. So while attempting to satisfy his goal of locating the support of the pivot arm on either the inside or the outside, he relaxed it to the higher-level goal of just supporting the arm.

The idea of the forked support was quickly rejected, and he was left with the original two options but with no new ideas on how to evaluate and compare them. At this point in the protocol, S6 became opportunistic. He abandoned his generate and test strategy and instead investigated the safety of the design in the area where the pivot arm and support are connected. It appears from the video tape that he had focused in on safety concerns of the pivot point due to a mental simulation of the pivot arms rotating about the support. He does not appear to be thinking about any particular support location but simply the support and pivot arm connection in general. He addressed several safety concerns in events 66.37–66.45A. He did not evaluate the inside or outside location with respect to safety, so presumably he was thinking about them collectively. By event 66.45B he had not identified any important safety concerns, and he simply dropped the topic, evidently convinced there was no safety problem. He returned to the support location problem and, after a few seconds of thought, recognized that an outside location for the support would make it easier to screw the support to the table than an inside location (event 66.47). He therefore decided on the outside location and drew the support on his sketch.

5 Observations from the Protocol Data

Preliminary reduction of the protocol data has yielded much information that allow us to begin to characterize the mechanical design process. Those results we have confidence in are reported below, complete with comparisons to the current design theories and the results from the studies on software design. The implications of these findings for the development of future CAD tools are also discussed. These results are based primarily on the partial reduction of three protocols (S2, S5, and S6)—all experienced mechanical designers. We are confident however that the findings reported below are common in experienced mechanical designers. We do not, however, know whether they are examples of good or bad design practice.

5.1 Designers pursue a single conceptual design

In all of the protocols, an initial, preliminary design was established very early in the effort. This design became the theme for the solution to the problem and was modified and patched until it worked. There was little evidence of considering alternative conceptual designs. We seldom saw parallel development of more than one idea, and if several alternatives were initially developed, all but one would be eliminated very rapidly. This single-concept strategy was observed not only for the overall design, but also for individual components within the design at all levels of detail. If problems with the original concept were later uncovered, they were solved by patching the design rather than discarding it and developing a new approach. The first idea was almost sacred, and sometimes even highly implausible patches would be applied to make it work.

This is contrary to the principles espoused for good design practice in the current design theories (Hubka, 1980; Pahl and Beitz, 1984). However, we feel that this single-concept strategy is quite common, and that it is often suppressed by working in groups or by design reviews with management. We were concerned that possibly our subjects were behaving in this manner because the design was never to be reviewed or built. However, we find this unlikely because the subjects were also aware that they were being video-taped and thus recorded for all time. This probably encourages them to take extra care in their design. Experience in protocol analysis indicates that subjects tend to work more slowly and carefully on camera. Hence, we conclude that this single-concept strategy is their normal problem solving strategy.

It is interesting to note that the protocol studies of software design (Section 3.2) have found very similar behavior in their subjects. In virtually all cases, the design developed around a kernel idea that was successively refined during design process.

The implications of this for the study of mechanical design theory are quite serious. Either our subjects are poor designers because they employ a weak strategy, or else the theories about efficient design methods are incorrect. Since the subjects are experienced designers, well-respected by their employers, there is little reason to believe that they are inefficient or that they produce inferior designs. However, it is conceivable that design is such a complex task that there are very few people who do it well. In this case, the methodologists may be correct, and our subjects may need to be retrained or provided with better design tools. The only way to settle this issue is to develop some means for judging the efficiency of the design process and the quality of the designs. At present, we do not have objective ways of making this judgment.

It is important to resolve this question before we construct AI-based CAD tools for designers. If the single-concept strategy is appropriate, it will significantly simplify the development of CAD tools, because they will not need to store and update multiple alternative designs. On the other hand, if the single-concept strategy is inferior, it will be critically important for AI-based CAD tools to support multiple alternatives and to assist the designer in managing these alternatives. This will require much more complex software. The complexity of such systems suggests that human designers follow the single-concept strategy because it is cognitively easier rather than because it produces superior designs more efficiently.

5.2 Notes and drawings play a critical role in design

During the problem solution, the engineers took extensive notes both mental and written. Most of the written notes for these mechanical designers were in terms of drawings and sketches. From the protocols, we have observed six uses of the act of drawing and sketching in the design process. These are:

(a) To archive the geometric form of the design.

(b) To communicate ideas from one designer to another and from the designers to the manufacturing personnel.

(c) To provide a visual simulation of ideas. Often the designers sketch various options in an effort to simulate configuration or information flow.

(d) To act as an analysis tool. Often, missing dimensions and tolerances are calculated directly on the drawing as it is developed.

(e) To serve as a completeness checker. As sketches or other drawings are being made, the details remaining to be designed become apparent to the designer. This, in effect, helps establish an agenda of design tasks left to accomplish.

(f) To provide a kind of "external memory." The designers often made sketches to help them remember ideas that they were afraid they might forget. This was not always successful, as we saw many cases of the designer forgetting ideas anyway. This is further discussed below.

This use of notes is consistent with the behavior found in the software studies. In our comparison, we are not yet able to determine whether more notes are taken by subjects working in unfamiliar domains as reported by Adelson and Soloway (1984).

It is interesting to observe that in the software studies, the process of mental simulation and test-case execution served many of the same purposes as drawing does in mechanical design. In particular, mental simulation of algorithms and programs has been observed to provide functions (e) and (f) in software design.

Current mechanical engineering CAD tools only serve the designer directly on items (a) and (b) and indirectly on items (c) and (d). The drawings do not offer any active mechanism to point out missing dimensions or to assist in calculations. It is in the development of tools to better serve the last two items that we may see an increase in design efficiency.

5.3 Progression from systematic to opportunistic behavior

Our design engineer subjects began their design efforts in a systematic manner and became, as the design progressed, increasingly opportunistic. Initially each designer went over the problem statement to get an overview of what he/she faced and to identify important design functions. They noted key points and established an initial agenda for solving the problem. This was followed by a systematic period of conceptual design. The subjects usually followed an organized plan of attack at this early stage.

As the design progressed the subjects became more opportunistic. For example, a designer might be focused on a specific problem to be solved and suddenly "notice" another (usually adjacent) problem that either overwhelms the current focus or displaces the other items on the subject's informal agenda. We saw an example of this in the previous section, where S6 interrupted his efforts to locate the pivot arm supports and shifted to worrying about safety issues (Figure 5). In this case, we believe that he switched his attention due to a mental simulation of the pivot arms rotating about the support. In other areas, we suspect that once the basic form has been developed and sketched, the designer starts to be driven by visual cues. Ideas and problems are triggered by noticing patterns or configurations in the sketches. Sometimes the focus of attention is immediately shifted to one of these, perhaps because the designer doesn't want to lose the "good idea" by forgetting it. We also suspect that as the design progresses, the complexity increases to the point where the designer can only keep a small portion of the whole design in his short term memory or in front of him as a sketch. Consequently, his agenda is formed by what he can remember and see.

There is no reported evidence of this behavior with the software designers. Both our work and the software studies show the flow of the design from the abstract to the concrete, but this flow from systematic to opportunistic was unexpected. Perhaps we are seeing this behavior because our design problems are significantly larger that those studied previously, and hence, our subjects reach the point where they become cognitively overloaded.

There is no discussion of this kind of behavior by the design theorists either. The design theories are, by definition, *systematic* strategies for conducting design, and thus our designers again diverged from the current thinking on what makes for a good design process.

There are two basic implications of this opportunistic behavior for the development of future CAD systems. First, CAD systems should provide support for opportunism. They should not put the designer in a methodological straight-jacket that prevents him/her from pursuing important opportunities and problems as they are discovered. Second, if our speculations are correct that one cause of opportunism is cognitive overload, AI-based CAD systems should also assist the designer in managing this overload, for example, by maintaining an agenda of suspended tasks and reminding the designer of tasks requiring further attention.

5.4 Balanced development

In most design theory and in the software studies, it is claimed that the designer makes an effort to keep the design balanced. We have seen both supporting and opposing evidence for this claim.

Subject S2, working on the battery contacts problem, seemed to work in a very unbalanced manner, in that he focused on one small part of the total problem before even addressing many other aspects of the problem. This one aspect was developed in great detail before continuing.

Subject S6, working on the flipper-dipper problem, seemed to be operate in a balanced way. He repeatedly worked on each component in the design, beginning with the clamp that holds the plate and ending with the question of how the machine attaches to the table. This corresponded to a spatial traversal of the device from one end to the other. At the end of each pass, he would analyze the overall design and then return to the clamp and address each component again. This process was his overall strategy. He did diverge from it to make minor decisions. Only when S6 got to the detail stages of the design did he concentrate on developing one component at a time.

We hesitate to draw any conclusions from these observations because of ambiguity in the term "balanced development." We have been defining it as the effort to keep all the elements of the design at the same level of abstraction while moving the design toward completion. However, an alternative definition is that the design process is balanced if the designer addresses every element of the design on each pass, regardless of the level of abstraction of the elements. According to this definition, each pass is intended to reduce the level of abstraction of the design as a whole.

In our protocol analysis, it is easy to determine whether the subject is satisfying the second definition of balanced development. However, to evaluate the first definition, we must be able to determine when elements are "at the same level of abstraction." We do not yet have an objective definition of levels of abstraction.

As with the other behaviors we have already discussed, any departure from balanced development suggests that future CAD tools should allow the designer great flexibility in exploring the design space. Tong's work on AI-based CAD tools for VLSI design (Tong, forthcoming) attempts to provide this flexibility. Tong suggests that departures from balanced development are necessary and appropriate when the design problem contains identifiable bottlenecks. In such cases, the designer should take some time to explore the bottleneck in detail and determine whether a feasible design can be constructed. This may explain the behavior of S2, since he was working on the battery contacts problem which has a very tight spatial constraint—a bottleneck in Tong's terminology.

5.5 Designers are forgetful

We have seen repeated evidence that the designers forget their earlier decisions. For example, early in his design, subject S6 considered the problem of how to mount the support for the pivot arms. He decided to mount the support on the outside and even made a sketch recording the decision. Later on, he reconsidered this problem in the episode shown in Figure 5. Again he considers alternatives, rejects some of them, and settles on the "outside" location. Throughout the second episode, he gives no indication that he remembers his previous solution.

There are, of course, cases in which our subjects readdress problems because new constraints or interactions have arisen and require attention. However, this cannot explain cases such as S6's in which the exact same decision is made twice, with no new constraints or any sign of recollection on the part of the subject.

This forgetfulness of designers is not at all surprising given the complexity of the design process. Future CAD tools should provide support for recording design decisions and helping the designer keep track of what work has been accomplished and what work remains to be done.

6 Concluding Remarks

Mechanical design is a very complex cognitive task, and we have seen that it is still very poorly understood. This paper has demonstrated that the method of protocol analysis can be a valuable technique for studying the design process. The major observations that we have made so far (based on only a preliminary analysis of our data) can be summarized as follows:

- Designers pursue a single conceptual design both at the level of the overall design problem and at the level of each individual subproblem.
- Notes and drawings play a critical role in design. They are not merely a method of recording the final design decisions.
- Designers progress from systematic to opportunistic behavior as the design evolves.
- Designers do not always seem to conduct balanced development.
- Designers are forgetful and repeat their work.

These observations have important implications for the study of design methodology and the development of AI-based CAD tools. In design methodology, it is important to develop objective criteria for judging the quality of the design process as well as the quality of the final design. Many of the behaviors that we have observed have previously been labeled as suboptimal by design theorists. The protocol data suggest that these judgments may be premature. There may be very sound reasons why human designers work in this way. Further study is required to determine whether these behaviors are in fact inferior, and if so, whether they result from poor training or from cognitive limitations.

In any case, the data also suggest that future AI-based CAD tools must be highly flexible, so that they can allow the designer the freedom to depart from balanced development or pursue opportunistic search strategies. In addition, future CAD tools have an important role to play helping the designer manage the complexity of the design, for example, by keeping an agenda of outstanding problems and tasks and by providing a smooth conceptual interface that helps the designer grasp the entire state of the design and the interactions between various components.

It is our hope that further analysis of our protocol data will help us to sharpen these observations and clarify many of the outstanding issues mentioned above. The development of an empirically-justified theory of design will provide a sound basis for the development of future CAD tools and for the training of future mechanical engineers.

7 Acknowledgements

The authors gratefully acknowledge the support of the National Science Foundation under grant number DMC-8514949.

8 References

Adelson, B., and Soloway, E., 1984. A cognitive model of software design, Rep. No. 342, Department of Computer Science, Yale University.

Bailey, W. and Kay, E. 1986. Toward the standard analysis of verbal data. Tektronix Report. Tektronix, Inc. Beaverton, OR.

Ericsson, K.A. and Simon, H. A. 1980. Verbal reports as data. *Psychological Review* Vol. 87 (3) 215–251.

Hubka, V. 1982. *Principles of Engineering Design,* London: Butterworth Publishers.

Kant, E. 1985. Understanding and automating algorithm design. In *Proceedings of IJCAI-85,* Los Altos, CA: Morgan-kaufmann. 1243–1253.

Kant, E., and Newell, A., 1982. Naive algorithm design techniques: a case study. In *Proceedings of the European Conference on Artificial Intelligence,* Orsay, France, July. 40–51.

Love, S.F. 1980. *Planning and Creating Successful Engineered Designs.* Advanced Professional Development, Inc., Los Angeles.

Marples, D. L. 1960. The decisions of engineering design. *The Engineering Designer.* December.

Mitchell, T. Steinberg, L, Shulman, J., 1984. VEXED: a knowledge-based VLSI design consultant, Rutgers AI/VLSI Project Working Paper No. 17.

Mitroff, I. I., 1967. A study of simulation-aided engineering design. Doctoral dissertation. University of California, Berkeley.

Newell, A., and Simon, H. A. 1972. *Human Problem Solving.* Englewood Cliffs, NJ: Prentice-Hall.

Ostrofsky, B. 1977. *Design, Planning, and Development Methodology.* Englewood Cliffs, NJ.: Prentice-Hall.

Pahl, G. and Beitz, W. 1984. *Engineering Design.* London: Springer-Verlag.

Rabins, M. et al. 1986. Design theory and methodology—A new discipline. *Mechanical Engineering,* Vol 108 (8) 23–27.

Steier, D. M., and Kant, E., 1985. Symbolic execution in algorithm design. In *Proceedings of IJCAI-85*, Los Altos: Morgan-Kaufmann, 225–231.

Tong, C. Forthcoming. Knowledge-based circuit design. Doctoral dissertation. Department of Computer Science. Stanford, University.

Weisberg, R. 1986. *Creativity.* New York: W. H. Freeman and Co.

Appendix I. Protocol Design Problems

I.1 Design Problem 1: The Flipper-Dipper

Our manufacturing company needs a machine to coat thin, aluminum "plates." A thin chemical layer will be cast on the surface of a water bath for coating the plates. The machine will dip the plates into this chemical bath. We need three of the machines produced and have standard machine tools in-house.

Specifically, the machine must coat the plates as follows:

A worker loads the machine with a .063 × 10 × 10 inch aluminum plate. Since the worker needs to load and unload these plates all day from a standing position, fatigue should be kept to a minimum.

The worker visually insures that the surface of the water is clean and then uses a syringe to inject a pre-measured amount of chemical in solvent solution on the surface of the water. The chemical solution spreads as an oil slick over the surface. When the solvent evaporates (just a few seconds) the 500 Angstrom thick chemical layer is ready to be applied to the surface of the fin. The chemical is nontoxic and safe to handle.

The chemical is applied to the surface of the plate by gently lowering the fin onto the water where surface tension will cause them to bond instantly. Once the plate is coated, it is moved away from the surface with one edge leading, and the process is finished for that side of the plate. The excess chemical on the surface of the water is cleared from the bath manually by the worker (the layer is very thin and sticky).

The process is repeated to coat the other side of the plate. After coating both sides, the plate is then presented to the worker for unloading. To dip the plate, add new chemical, flip it, and dip again takes about 40 seconds.

There are a few constraints on the problem, namely:

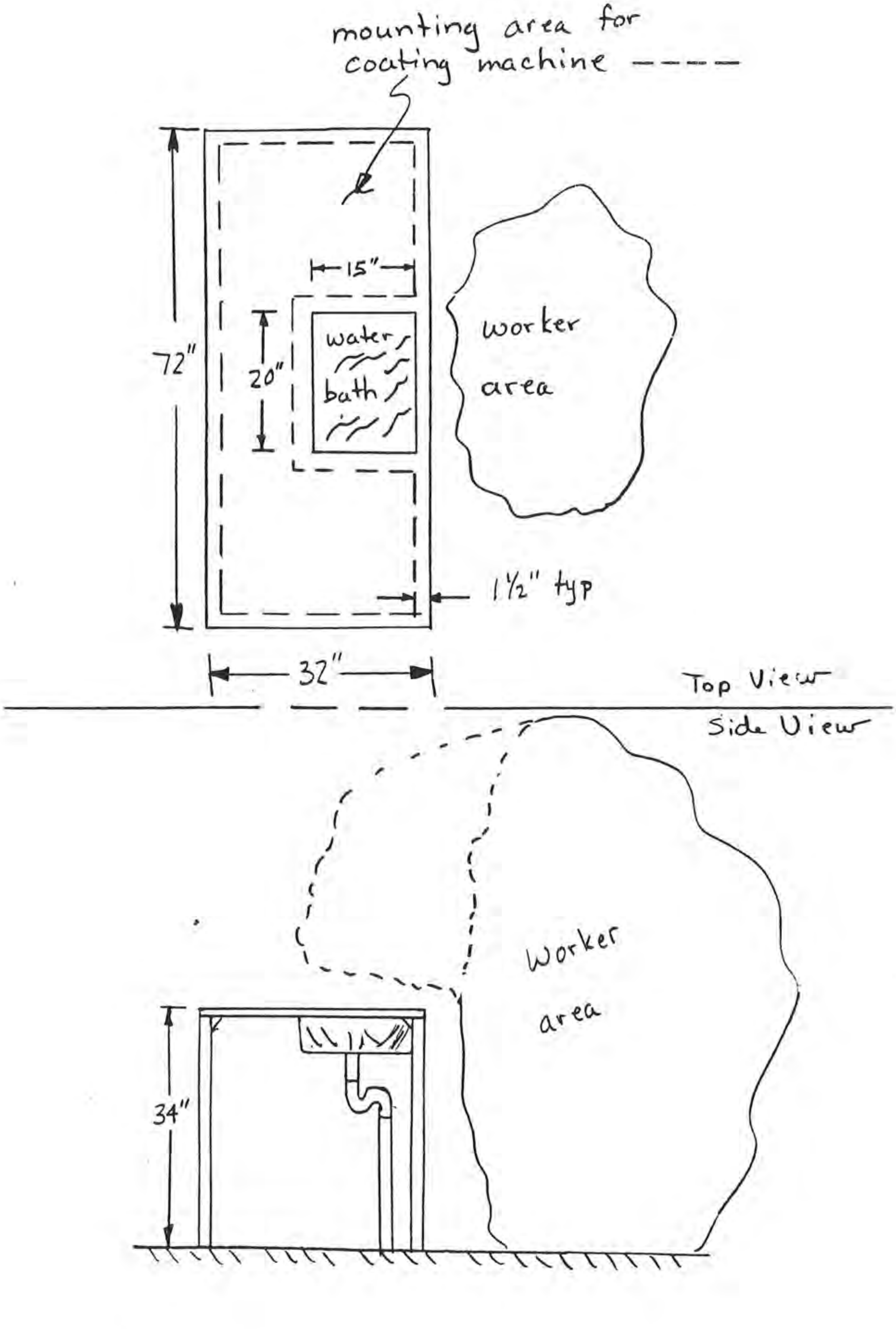
1. The plates can only be edge handled. Only the edge 1/4 inches around the periphery of the fin can be touched by either the worker or the machine at any time. See attached page.

2. The water must be kept clean, as any impurities can affect the integrity of the chemical. This is especially true of organic materials.

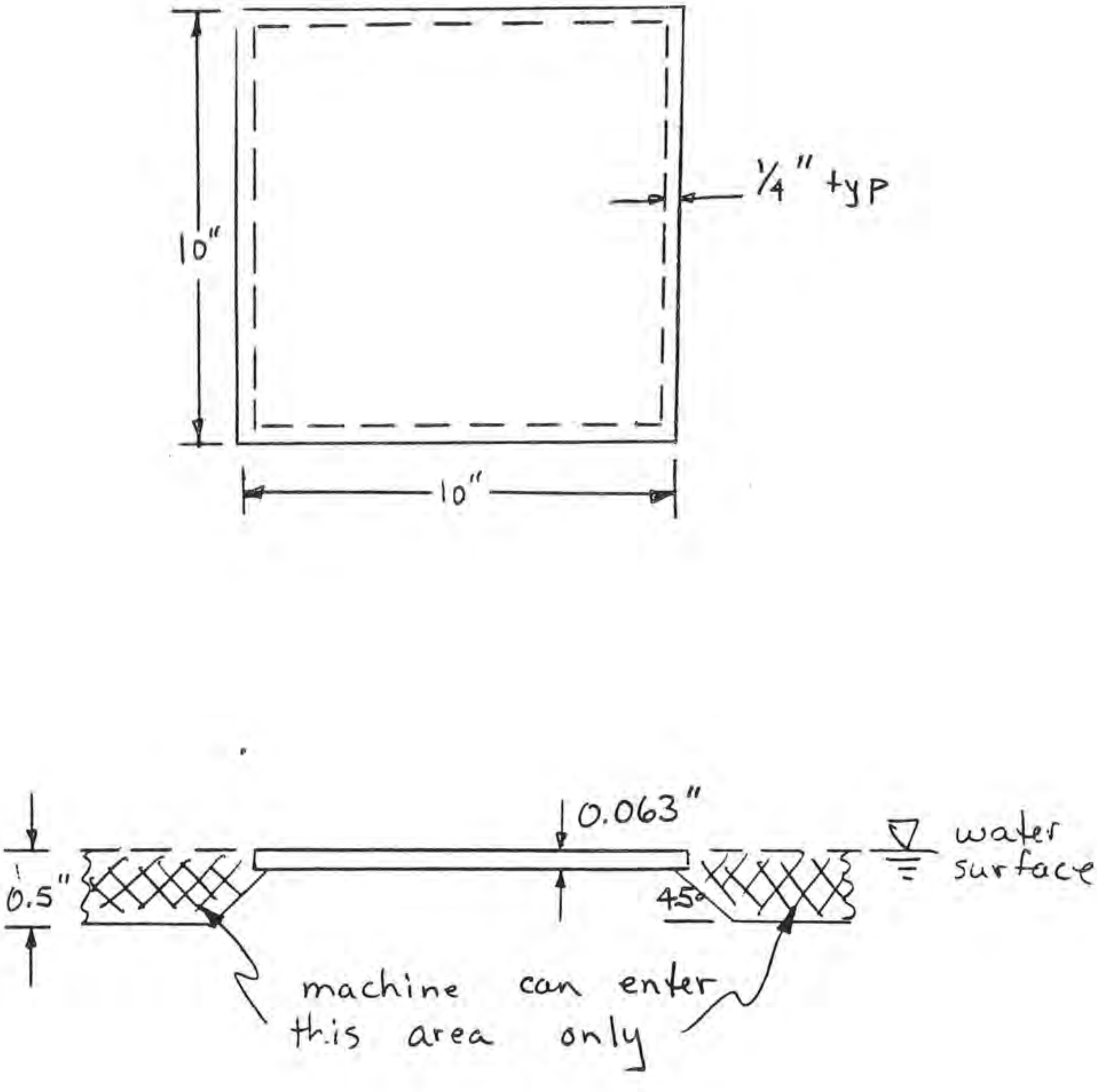
3. Parts of the machine that hold the plate can enter the water, outside the periphery of the plate, to a depth of up to one-half inch.

4. It is anticipated that the machine will mount on the table surface in the areas shown. The machines cannot extend beyond the boundary of the table.

5. The water bath level is automatically maintained 0.5 inches below the surface of the table, plus or minus 0.005 inches.



WORK STATION



ALUMINUM PLATE

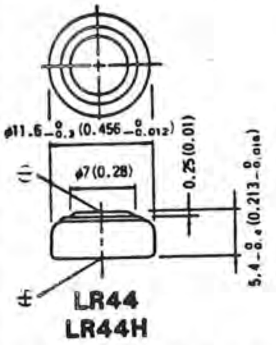
I.2 Design Problem 2: Electrical Contacts

A high-tech electronics company is manufacturing a new portable computer. As part of the overall design, three batteries are needed to power a time clock. The company needs you to design the electrical contacts for holding these batteries and connecting them to a printed circuit board.

The specific design requirements are

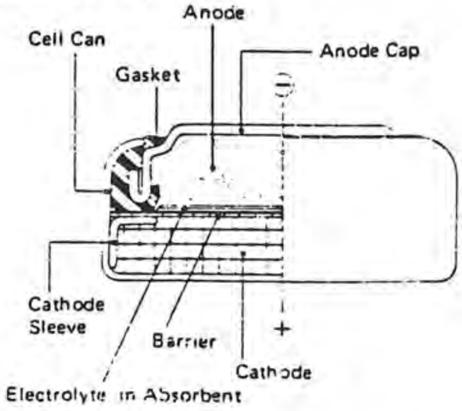
1. Batteries. Three required in series, type LR44, see attached sheet for specifications.
2. Envelope. The components must fit within the plastic case walls. See attached figure. The walls may have slots for contacts and locating features, etc. The envelope has five walls. The sixth side (where the batteries go in) must remain open. Another component, designed by someone else, will butt up against your envelope and keep the batteries from falling out. The dimensions given are for the interior of the envelope. Wall thickness and inside shape are up to you, but the thickness should not exceed 60 thousandths of an inch.
3. Contacts. The contact force shall be 0.1 lb. minimum and 1 lb. maximum at the printed circuit board and at the batteries. The contact plating shall be nickel. The printed circuit board contacting area has a diameter of .100 plus or minus .005 inches. The contact locations are shown on the attached sheet. Any contact locations for X1 and X2 will be compatible on the printed circuit board. (The design of this board has not been finalized.) The contacts cannot extend below the bottom of the envelope.
4. Assembly. The computers will be assembled by robots, so the contacts will be handled and fit into place by a robotic end-effector (yet to be selected, but probably 1/4 inch suction type). The envelope has an upper and lower half. During assembly, the lower half will already be in place. After the electrical contacts are in place (the part you are designing), the printed circuit board and the top half of the envelope will be set in place.
5. Quantity. 50,000 units will be assembled per month for three years.

Model No.	JIS	I.E.C	Nominal Voltage (V)	Nominal Capacity (mAh)	Recommended Drain			Dimensions		Weight (g)	Others
					High (mA)	Standard (mA)	Low (μA)	Diameter (mm)	Height (mm)		
LR621H		LR60	1.50	14(1.2)		0.05	1	6.8	2.15	0.4	
LR626			1.50	17(1.2)		0.05	1	6.8	2.60	0.4	
LR626H			1.50	19(1.2)		0.05	1	6.8	2.60	0.4	
LR726		LR59	1.50	21(1.2)		0.05	1	7.9	2.60	0.5	
LR41	LR41	LR41	1.50	24(1.2)	5	0.10	1	7.9	3.60	0.6	192
LR41H	LR41	LR41	1.50	35(1.2)	5	0.10	1	7.9	3.60	0.7	
LR920			1.50	24(1.2)		0.05	1	9.5	2.05	0.7	
LR927		LR57	1.50	35(1.2)		0.10	1	9.5	2.70	0.8	
LR1120	LR1120	LR55	1.50	23(1.2)	5	0.10	1	11.6	2.05	0.8	191, 91A
LR1130	LR1130	LR54	1.50	44(1.2)	10	0.10	1	11.6	3.05	1.2	189, 89A
LR43	LR43	LR43	1.50	70(1.2)	15	0.10	3	11.6	4.20	1.6	186, 86A
LR44	LR44	LR44	1.50	105(1.2)	50	0.10	5	11.6	5.40	2.0	A76, SB-F9 76A
LR44H	LR44	LR44	1.50	145(1.2)	20	0.10	5	11.6	5.40	2.2	SB-F9H
LR9		LR9	1.50	190(0.9)	50	5.00	10	15.7	6.10	3.3	AM-D
LR50		LR50	1.50	580(0.9)	100	10.00	25	16.4	16.80	10.0	AM-P
PX-825		LR53	1.50	300(0.9)	50	5.00	15	23.1			
PX-24		2LR50	3.00	580(1.8)	100	10.00	25	16.9			
PX-30		2LR53	3.00	300(1.8)	50	5.00	15	24.0			
PX-19		3LR50	4.50	580(2.7)	100	10.00	25	16.9			
PX-21		3LR50	4.50	580(2.7)	100	10.00	25	16.9			
7K31			4.50	105(2.4)	50	5.00	5	(11.5X17.2)			
4LR44		4LR44	6.00	105(3.0)	50	3.00	5	13.0			



Note : ※Nominal capacity shown above is based on standard when battery is used till the end voltage indicated in parenthesis (
Some items require special order manufacture, so be sure to confirm the delivery date, etc., in advance.

● Cutaway view



● Shelf life characteristics of Alkaline Button Battery

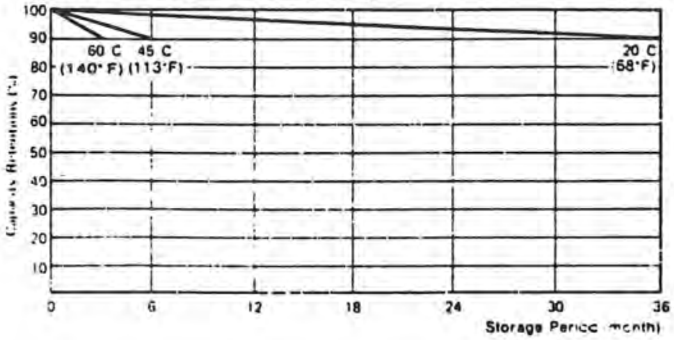


FIGURE 1 BATTERIES

Ⓐ BATTERIES SLIDE IN HERE

Ⓑ PRINTED CIRCUIT BOARD CONTACTS, .100 ±.005 DIA.

Ⓒ PRINTED CIRCUIT BOARD SHOWN IN DASHED LINES — — —

Ⓓ SEPARATION BETWEEN TOP & BOTTOM HALVES OF PLASTIC ENVELOPE

Ⓔ BATTERIES (TOP VIEW ONLY)

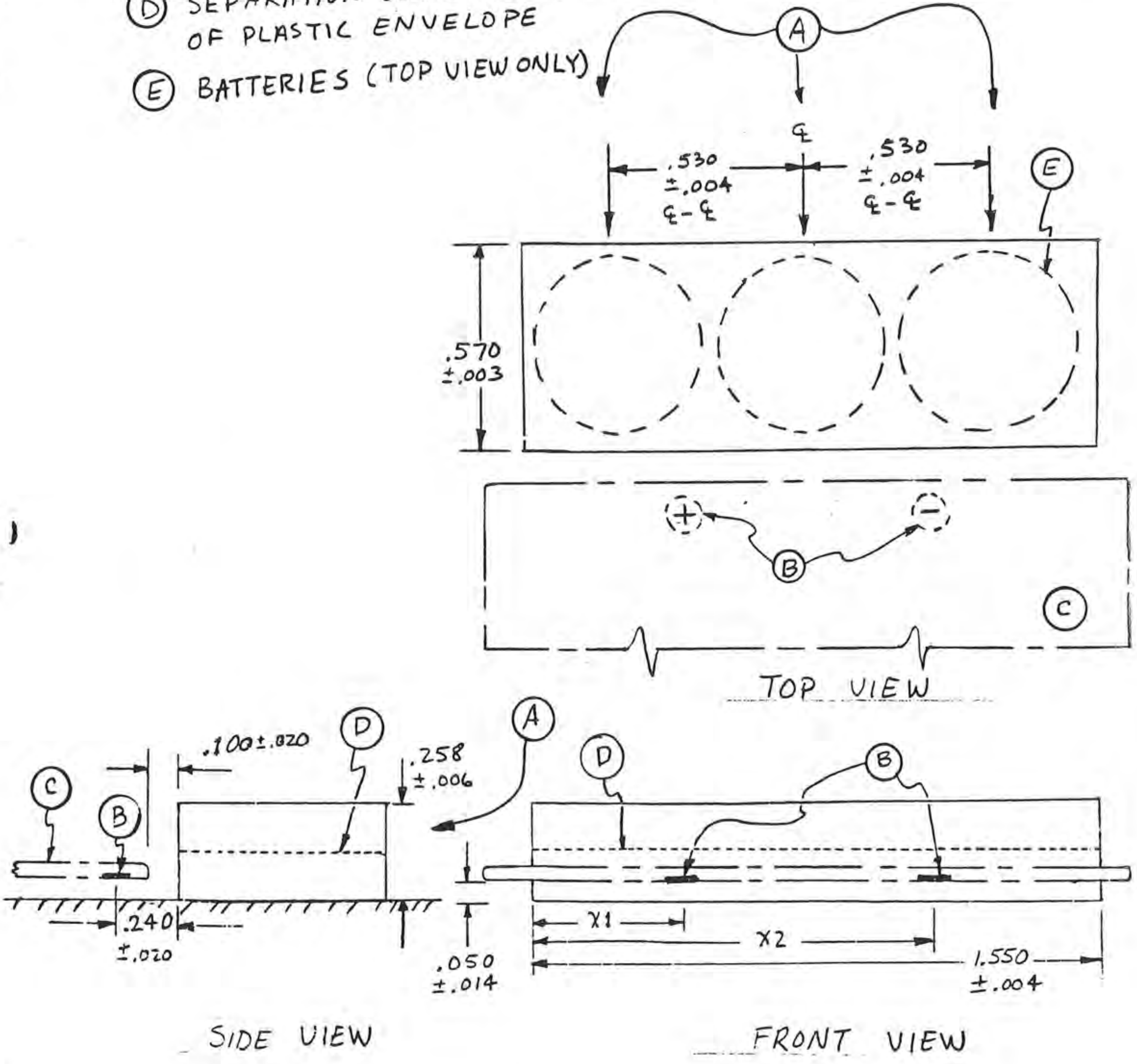


FIGURE 2 BATTERY HOUSING

Appendix II. Written Instructions for Subjects

In this study, I will present you with a design problem, and I want you to solve it the way you would ordinarily solve any other design problem. The only difference is that I want you to THINK ALOUD as you work on the problem. Say what you are thinking about at each point. I want you to talk aloud constantly from the time I present the problem until you are done. I don't want you to plan out what you are going to say or even explain to me your thoughts. Simply act as if you were alone talking to yourself.

There will of course, be times when you are thinking about something non-verbal. In those cases, just say something like "I'm visualizing" or "I'm imagining how it will work" or "I'm rotating it in my mind." I will not interrupt you at all, except to remind you to think aloud.

When you are working on the design problem, please feel free to consult any catalogs or reference books. I want to see how you actually go about solving design problems. If there are any questions about the problem, please ask me. Also, it may be that, in the process of solving a problem, you would normally make a phone call to some vendor or colleague. In such cases, you should instead ask me the questions that you would ask them. Remember, please think aloud. Say everything that comes to mind. Also, please take your time in solving the problem. Due to the verbalization, your solution will take longer than if you were silent: don't let this bother you. I am interested in how you go about solving the problem, not in the speed or feasibility of your solution. Try to work as you normally would, but remember ...THINK ALOUD!

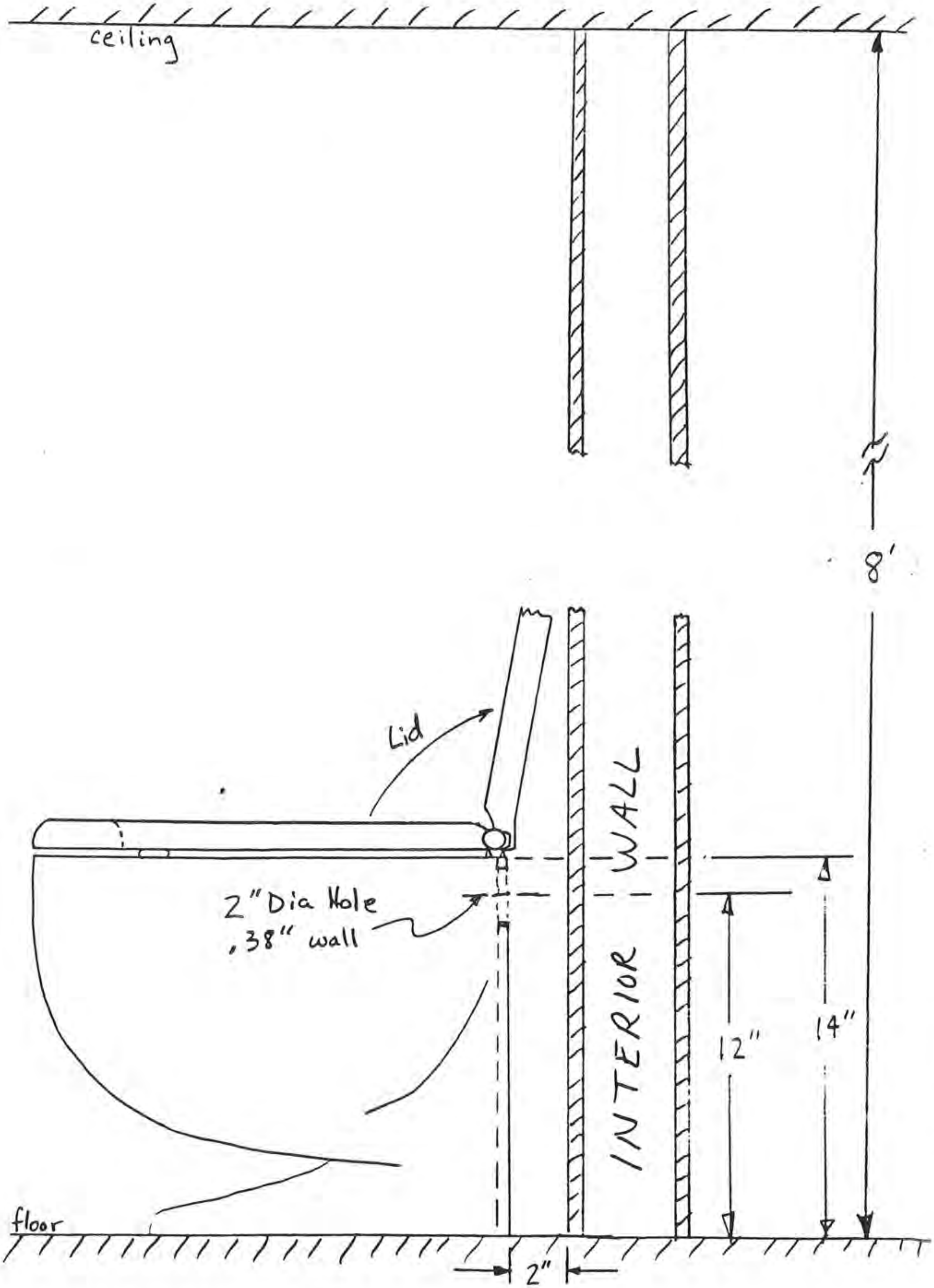
Appendix III. Practice Problem

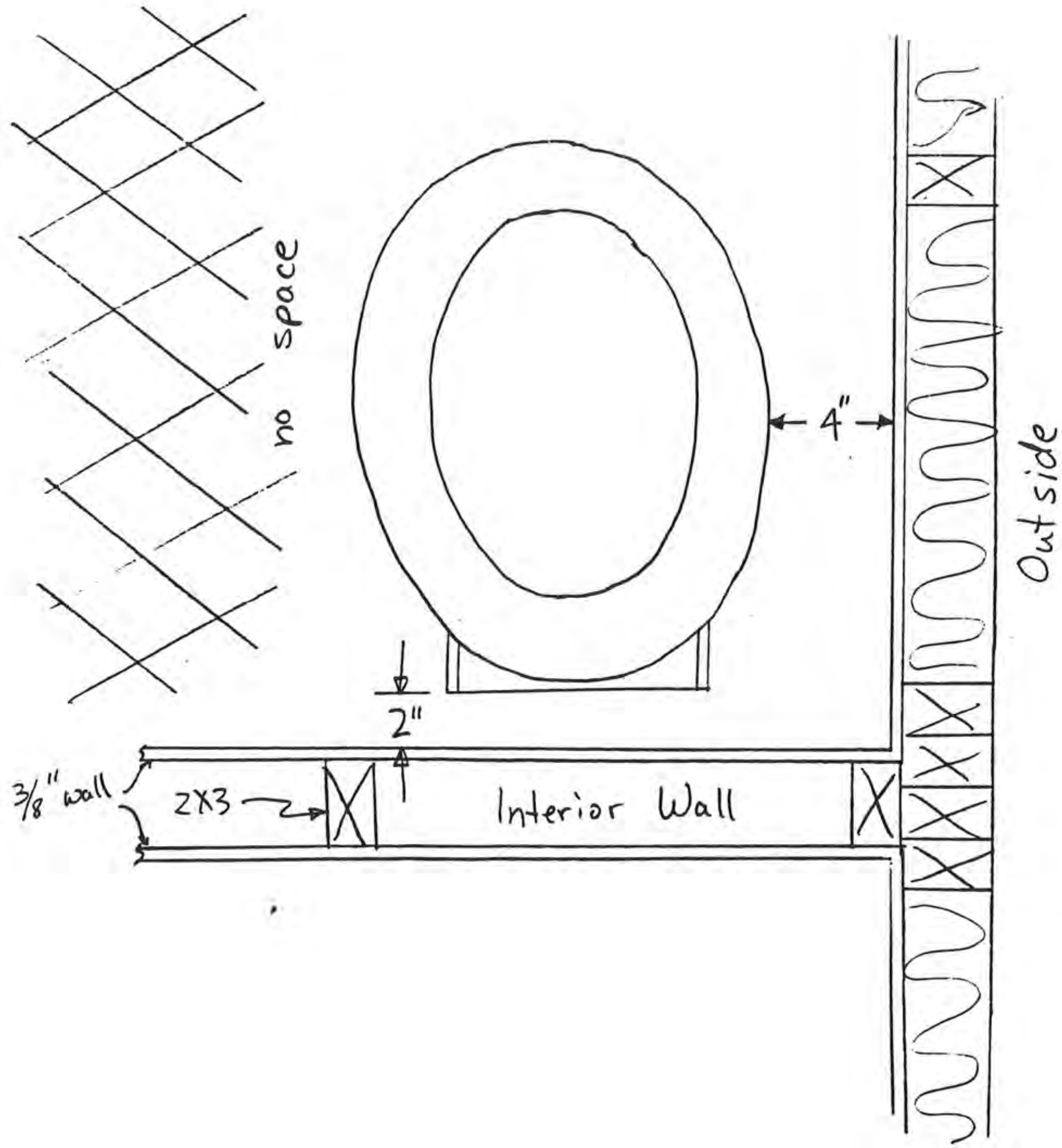
Our company, which manufactures mobile homes, has a space problem in the bathrooms of one of our new models. The bowl of the toilets must mount within two inches of the wall, therefore standard toilets can not be installed due to the size of their tanks. A company has been found who can supply a fiberglass bowl without a tank.

We need you to design a tank that fits in the available space, see figures.

The following constraints apply:

1. The bowl has a two inch inlet hole for the flush water.
2. Nine gallons of water is needed for each flush.
3. Supply water to the mobile home will be available at 20 psi.
4. The company anticipates manufacturing four-thousand units over the next several years.


ceiling
8'
Lid
INTERIOR WALL
2"Dia Hole
.38" wall
12"
14"
floor
2"


no space
4"
Outside
2"
3/8" wall
2X3
Interior Wall