

Preprint. Final version available as:

HILTZ, K., BACK J. & BLANDFORD, A. (2010). The roles of conceptual device models and user goals in avoiding device initialization errors. *Interacting with Computers*. 22.5. 363-374. DOI <http://dx.doi.org/10.1016/j.intcom.2010.01.001>

## The roles of conceptual device models and user goals in avoiding device initialization errors

Kimberley Hiltz<sup>1</sup>, Jonathan Back & Ann Blandford  
UCLIC, University College London  
Gower Street, London WC1E 6BT

Corresponding author: Ann Blandford  
Tel: 020 7679 0688  
A.Blandford@ucl.ac.uk

---

<sup>1</sup> Now at Systems Concepts, 2 Savoy Court, Strand, London, WC2R 0EZ

# The roles of conceptual device models and user goals in avoiding device initialization errors

## Abstract

While mistakes, and approaches to design and training that reduce them, have been studied extensively, relatively little work in HCI studies ‘slip’ errors, which occur when one intends to do a certain action during a skilled task but unintentionally does another. In this article we examine approaches to training that might reduce the occurrence of a slip error referred to as a ‘device initialization error’. This error occurs when skilled users of a device forget to perform some initialization action, such as positioning the cursor in a text entry box or setting the device into the correct mode, before entering data or performing some other significant activity. We report on an experiment studying the effects of two training interventions on this error, which aim to manipulate the salience of the error-prone action without making any physical changes to the device. In the first intervention participants were given a particular conceptual model of the device’s operation, to evaluate whether having an improved understanding of the effect of each action would lead to fewer errors. In the second, participants were given a new device operation goal requiring them to ‘test’ the device, to evaluate whether attending to the outcome of initialization actions would lead to fewer errors. Only participants who were asked to ‘test’ the device and also given enhanced instructions to enter dummy data after completing initialization actions showed a statistically significant improvement in performance. Post-test interviews and evidence from existing literature suggest that when participants forgot the initialization step it was because they were attending to the subsequent data entry steps. This study highlights the central roles that user goals and attention play in the occurrence (or avoidance) of slip errors.

Key Words: Human error, cognition, cognitive slips, device design, conceptual models, task instructions, task structure

## 1 Introduction

Slip errors can occur systematically even when individuals have the required ‘expert’ procedural knowledge to perform a task correctly. After learning how to perform a task, some task actions are easily performed, while others have properties that make them difficult to remember. Some actions seem lower in relevance to primary goals, especially when they are not cued by the external environment or internal user goals. Byrne and Davis (2006) found that simply telling users that an action is important to fulfilling the goal (e.g., by associating its completion with reward or its omission with penalty) does not make it less prone to error. The work reported in this article aims to determine whether it is possible to directly manipulate the relevance (or salience) of error-prone actions during training, thereby reducing the likelihood of omission. While such errors have most significant consequences in safety-critical situations, they occur widely; the focus of this paper is on the underlying causes of omission errors.

This research builds on work by Li et al (2008), who developed an experimental paradigm for studying the effects of interruptions on post-completion errors during procedural tasks, based on a simulated doughnut-making machine. A post-completion error (PCE) is the omission of a ‘clean-up’ step after the completion of a main task, such as forgetting to retrieve the original from the photocopier after the copy has been made (Reason, 2002), or forgetting to log out of a shared system after the work has been completed (Blandford and Rugg, 2002). Li et al found that an interruption immediately before the post-completion step of the task caused the PCE rate to increase significantly, but that an interruption elsewhere in the task did not. An unexpected observation in Li et al’s studies was that there was also a very high error rate on another task step. This step, which we will call the device initialization step, was to initialize a device component called the “dough port” before entering any data about the quantity of dough to be cooked; it was omitted on approximately 35% of trials. Although there were similar steps at four other points in the procedure, the error rates at these intermediate steps were much lower, typically occurring on only 3% to 5% of trials. This difference in error rate suggested something particular about the very first step in the procedure, which made it more prone to omission. The question we set out to address was whether the device initialization error (DIE) rate could be reduced by increasing the salience of the device initialization step, without making physical changes to the device interface.

Drawing analogy with the notion of visual salience (as a property of objects that makes them stand out from the background, attracting visual attention), we investigated the role of *cognitive* salience of an action (being that the action “springs to mind”). While it would be necessary to redesign the interface to influence visual salience, we investigated redesigns of training material to influence cognitive salience.

## **2 Background**

Device initialization errors are not unique to the experimental paradigm developed by Li et al, although they have been the focus of relatively little study in the area of Human-Computer Interaction. One widespread example is omitting to select the correct window in a graphical computer interface, before typing or otherwise interacting with it (Lee, 1992). Another example was related to us by a programmer who uses a specialist editor: “When I want to commit a change, I type ‘svn commit’. This throws up an editor with a strip at the top where I can type in info and at the bottom info about what I have just committed. The problem is the editor is not in insert mode. I must type ‘a’ first to put it into insert mode. Then I can type the message. I repeatedly do not type ‘a’ first.” In both

of these cases, the user knows about the step, so the error is not caused by lack of knowledge, but by a failure to attend properly to the task step.

## 2.1 Approaches to studying error

Many studies of human error have relied on retrospective accounts of the situation and events that led up to the incident that has been categorized as an ‘error’, such as the anecdotes above.

Another approach to the study of errors has been to collect, analyze, and categorize errors that occur naturally. This has been useful for revealing the variety of different error types that occur, and for identifying common patterns of errors (Reason, 1990, p. 13-14). For example, verbal protocol studies of troubleshooters were used to develop the Skills-Rules-Knowledge (S-R-K) framework (Rasmussen, 1987), which has been highly influential in thinking about error. According to Reason (1990, p.43), skill-based errors “are related to the intrinsic variability of force, space or time coordination”, rule-based errors “are typically associated with the misclassification of situations leading to the application of the wrong rule”, and knowledge-based errors “arise from resource limitations (“bounded rationality”) and incomplete or incorrect knowledge”.

While retrospective and situated studies have contributed substantially to our understanding of error, they do not provide visibility to the underlying mechanisms involved (Gray, 2004). Gray argues that rigorous study of the nature, detection, and correction of errors should be pursued in laboratory settings. Laboratory studies enable the systematic manipulation of factors that are hypothesized to mitigate or provoke errors, allowing researchers to explore specific causal explanations in a controlled setting (Reason, 1990, p.14). Laboratory studies cannot replace situated studies, but provide a useful complement, challenging, validating or providing a richer understanding of the findings from situated studies.

There are two primary approaches to the study of errors in lab settings. The first, exemplified by Gray (2000), involves the collection of large amounts of data, both with and without errors, for the same task, which is then subjected to fine-grained analyses. The goal of this approach is to develop an understanding of error-free performance in addition to an understanding of the nature of the errors that occur.

The second approach, exemplified by Byrne & Bovair (1997) and Li et al (2008), is to develop an experimental task paradigm that induces an error rate high enough to be studied and analyzed in detail. Different factors that might contribute to or mitigate the error can then be explored in a controlled setting. This is the approach that has been taken in the work reported here.

## 2.2 Theoretical accounts and studies of omission errors

The device initialization error (DIE) is an example of what Reason (2002) terms an error of *omission*, which involves leaving out a necessary step in a task sequence. It is also an example of what Norman (1981, 1983a, 1988) terms a *slip*, which occurs when the correct action sequence is executed incorrectly. A critical feature of DIEs, and slips in general, is that knowledge of the omitted step and when to execute it is intact; that is, they are not errors that arise due to a lack of knowledge, and they generally occur during well-learned procedures.

Norman discusses the emergence of slips in terms of an *activation-trigger-schema* (ATS), which proposes that action sequences are controlled by the activation, selection, and triggering of hierarchically organized memory units called schemas. Schemas can be activated by cues in the external world or internal to the individual. Each schema has a set of triggering conditions, and will be selected based on a combination of its activation level and the goodness-of-match of its triggering conditions (Norman, 1983a). An example may be found in the work of Sellen et al (1992), who studied another type of slip called a 'mode error'. In their study, Sellen et al found that when participants had to maintain the mode (by keeping a foot pedal depressed) they made fewer errors than when the system maintained the mode: the act of maintaining the mode could be regarded as a triggering condition.

Related to Norman's ATS model, Altmann and Trafton (2002) propose a theory of cognitive goal representation and management called the activation-based goal memory model (AGM). According to this model, in order for a sub-goal to direct behaviour it must be the most active goal in memory. This model makes the important assumption that a goal's activation decays gradually but continuously, and that goal forgetting occurs in two ways: through interference from other elements in memory, and through the decay process. It also highlights the importance of internal and external priming cues to increase activation of a decaying target goal at the appropriate time.

Both the ATS and AGM models suggest that the DIE may occur because of insufficient activation of the device initialization step in the task procedure. According to the ATS account, the DIE might result from something inherent in the task or internal to the user causing the initial step to lose activation compared to the schema for the subsequent step. According to the AGM model, the device initialization goal is either consistently below the interference threshold for some reason, or the activation level for the subsequent step is consistently higher for some reason. An important avenue of investigation, therefore, is to identify different types of interventions that will increase the

internal activation (or salience) of the device initialization step at the appropriate time in order to reduce the rate of omission.

### 2.3 Alternative interventions

One possible class of interventions involves making physical changes to the device. For example, Chung and Byrne (2004) examined the effect of two different types of visual interventions on post-completion error rates: the first intervention was a just-in-time visual cue in the form of blinking red and yellow arrows, which appeared immediately before the PC step should be executed; the second intervention was a visual mode indicator that displayed a change in the system state using highlighting and contextual information. Chung & Byrne found that just-in-time display of the visual cue resulted in error free performance on the PC step across all trials, whereas presentation of the mode indicator did not have a significant effect on the PC error rate. They concluded that the visual cue acted as a primer to the PC step, contributing to its level of activation and allowing its goal to be satisfied, while the mode indicator did not sufficiently prime the PC step. An analogous just-in-time visual cue might similarly prime the device initialization step, and result in fewer omissions.

However, in situations where the device can not be significantly changed an alternative class of intervention is required. For example, Back et al (submitted) found that imposing a short period of reflection (displaying the screen but not accepting input for 4 seconds) reduced the device initialization error rate, suggesting that reflection supports mental rehearsal, and hence supports the user in identifying and cueing correct actions. This approach did not require significant changes to be made to the device.

Similarly, the study reported in this paper investigates two training-based interventions that do not require any change to the device. These alternatives were motivated by Li et al's (2008) suggestion that the omission rate on an initial procedural step might be related to that step's lack of relevance to the main task goal, or its lack of *salience*. That is, if the initial step does not directly move one closer to their main goal, but is a necessary step imposed by the device design, then it may be considered of low relevance to the main goal. Relevance to the main task-based goal has also been implicated in the omission of PCEs (e.g., Byrne & Bovair, 1997). However, there are no existing studies that directly investigate the role that salience or relevance plays in omission errors. This study investigates two approaches to influencing the salience of device actions from a user's perspective.

## 2.4 Intervention 1: Learning the conceptual model for the device

The first instructional manipulation reported in this paper drew on ideas from Norman (1983a; 1983b; 1988) and Kieras and Bovair (1984) on the role of the user's mental model in guiding their interactions with a device. It has been suggested that people develop mental models of the systems and devices they interact with, and use those models to guide their interactions and predict how a system will behave (e.g., Norman, 1983a; 1983b). Norman (1988) emphasized the need for a correspondence between a user's mental model and the conceptual model that a system is designed around, arguing that greater correspondence facilitates better learning of the system and leads to improved performance and problem solving.

Kieras and Bovair (1984) conducted an empirical study that examined what role, if any, mental models play while learning to interact with simple devices. In their studies, participants learned different procedures for operating a simple device; some were given a conceptual device model that described how the device worked before they learned the operating procedures, while others simply learned the procedures "by rote". Those given the conceptual device model learned the procedures faster, retained the procedures better, and applied more efficient procedures more often during the testing phases of the experiment. They were also able to infer new procedures for operating the device using fewer actions. Kieras and Bovair suggested that having a device model improves the learning and retention of operating procedures for a device, by supporting users in inferring what the correct operating procedures must be. For their device, the critical information was the system topology combined with information about how power flows through the system.

This work leads to the first hypothesis examined in this study, which is that performance on the device initialization step can be influenced by ensuring a good match between the device designer's conceptual model and the mental model formed by participants. The study reported here investigates the effect of training users on a device model that encapsulates the integral role of the device initialization step in the correct operation of the device. We hypothesized that users who learn the device model, which emphasized the conceptual importance of the initialization step, will make fewer device initialization errors than users who learn the procedure by rote.

## 2.5 Intervention 2: Increasing relevance to the task-based goal

The second intervention drew on the notion that users focus on the actions associated with domain goals at the expense of those associated with device goals (Ament et al., 2009; Cox and Young, 2000). One explanation for the high error rate associated with the

device initialization step is that since it is not directly related to accomplishing the task-based goal, it is attributed less importance and receives less activation. Encouraging users to focus on a different goal might be an effective way to influence the level of importance assigned to different parts of a procedure.

Intervention Two investigated the effect of modifying the task-based goal (to focus on testing the machine) so that accomplishing it is directly linked to correctly executing the device initialization step. We hypothesized that while all participants will execute the same procedural steps in the same order, the salience of the device initialization step will be higher for users instructed and trained to work towards the new task-based goal, leading to fewer device initialization errors during execution of the procedure than for those who were trained on the original goal.

### 3 Method

#### 3.1 Participants

Forty-eight individuals (24 men and 24 women) participated in the study, ranging in age from 20 to 67 ( $M = 30.15$ ,  $SD = 10.74$ ). . All participants had either previously completed an undergraduate university degree or were enrolled in one at the time of participation, and all were fluent readers and speakers of the English language. Volunteers were either personally known to the experimenter (and received no compensation for their participation) or were recruited via the University College London Psychology subject pool (and received £4.00 for their participation).

#### 3.2 Task environment

Two different interfaces simulated the doughnut-making operation. The first interface, called the Wicket Doughnut Call Center, simulated a call center in which participants had to respond to incoming calls from different customer locations in London to retrieve doughnut orders. The interface is shown in Figure 1. The *customer location selector* (item A in Figure 1) specified the location of an incoming call, the *customer location tube map* (item B in Figure 1) identified all customer locations in London with a doughnut symbol on the map, and the *customer order processor* (item C in Figure 1) was used to send the corresponding doughnut order details to the second interface, the Wicket Doughnut Making Machine.

*Figure 1 about here*

The Wicket Doughnut Making Machine, shown in Figure 2, simulated production of the doughnuts ordered through the call center. The Order Sheet (item A in Figure 2)

displayed the order details after the “Show Order” button had been pressed. Participants were required to enter data from the Order Sheet into the five machine components around the outside of the interface, to produce doughnuts that matched the order. The components had to be operated in the following sequence: Dough Depositor, Puncher, Froster, Sprinkler, Fryer. Prior to entering data in any component, the corresponding *selector* button (item B in Figure 2) had to be pressed to “activate” that component. After entering data into all five components, participants had to press the “Process / Clean” button (item C in Figure 2) once to process the order, and a second time to clean the machine. The *device initialization step* is the step that involved pressing the Dough Depositor selector button to activate the Dough Depositor component before entering any data about the quantity of dough to be cooked.

*Figure 2 about here*

An additional aspect of the task was that the data presented in the Order Sheet had to be transformed before entering it into the corresponding components. For example, the Order Sheet displayed the number of doughnuts required, but the quantity entered into the Dough Depositor component (item D in Figure 2) was for the amount of dough. As such, participants had to apply simple mathematical rules specified in the Dough Depositor, such as to add 5 to the number of doughnuts required, in order to get the amount of dough needed.

The experimental instrument used was an adaptation of that described by Li et al. (2008):

1. A power button was added in the top right-hand corner to support the device model that was created for the doughnut-making machine. The device model explained operation of the machine in terms of how power flowed through the system, so providing participants with a way to turn the power on and off was important
2. The name of the first machine component was changed from “Dough Port” to “Dough Depositor” to ensure that each component name enabled visualization of the operation it represented. The name “Dough Port” was difficult to associate with a corresponding action, while the remaining four component names were easily associated with actions that could be visualized (e.g., as the “Puncher” component might invoke images of the dough being punched into different shapes, the “Dough Depositor” might invoke images of dough being deposited onto a surface).
3. A progress bar that was present in the Dough Depositor component was removed to make this component consistent with the other four components, none of which contained a progress bar.

4. In Li et al's (2008) experiment, status text was presented briefly after each selector step was executed indicating that the component had been activated. Similar status text, saying "machine cleaned", was added after the "clean" step (i.e., the PC step) had been executed. This was to make the outcome of the device-specific PC step consistent with the five selector steps.

### 3.3 Materials

The materials used in this experiment included a paper-based diagram and description of the device model, paper-based descriptions of the task-based goals, three short paper-based quizzes used to evaluate comprehension of the paper materials, training material developed in Microsoft PowerPoint used to describe the task procedure in detail, two Microsoft Windows PCs with one running the call-centre code and the other running the doughnut-machine code (both written in Microsoft Visual Basic), and a post-study interview printed on paper but delivered verbally by the experimenter. In the experimental setting, the two computers were placed at 90° to each other, so that participants had to physically turn from the one to the other for the different task phases.

#### 3.3.1 Device model description

In the conceptual model that the doughnut machine design was based on, the device initialization step (clicking the Dough Depositor selector button) played a central role in the device's operation: prior to operating the Dough Depositor, the corresponding selector button had to be used to activate that component. Therefore the device model used to train participants, shown in Figure 3, was designed to communicate the role of the device initialization step, in an effort to support the development of a mental model that more closely corresponded with the conceptual model.

This device model is analogous to that provided by Kieras and Bovair (1984). Participants were informed that the *Power Switch* acts as the on/off controller for the main power source, and when turned on it causes the *Main Controller* to warm up and the *Power Indicator* light to turn green. Power always flows from the main controller to the *Current Order Sheet* display and to the *Machine Processor*, and the remaining power coming into the main controller is allocated to the different machine components (dough depositor, puncher, froster, sprinkler, and fryer) via the *Selector Switch*. The *Machine Processor* automatically controls power flow to the *Machine Cleaner*. The device model description did not contain any information about the doughnut-making procedure that participants were subsequently trained in.

*Figure 3 about here*

### 3.3.2 Task-based goal descriptions

Two alternative goal descriptions were used as the basis for training on the procedure: a low device relevance (LDR) goal description and a high device relevance (HDR) goal description. The LDR description emphasized the outcome of the procedure (whether the doughnuts produced matched the customer's order specification). It positioned participants as bakers for the Wicket Doughnut Company, and instructed them that their task-based goal was to produce batches of doughnuts that exactly matched orders from their customers (similar to the goal used by Li et al, 2008).

In contrast, the HDR description emphasized attending to specific device interactions (including execution of the device initialization step). Participants were positioned as Machine Testers for the company, and instructed that their task-based goal was to acknowledge and evaluate the machine's visual response each time a different selector button was pressed to activate a component in the doughnut-making machine. As shown in Figure 4, the *Selector* box in the doughnut-making machine provided textual feedback each time a new component was activated, and the role of Machine Testers was to report whether the device displayed the correct information. This information was present for all participants, but only the Machine Testers were explicitly instructed to attend to it.

*Figure 4 about here*

## 3.4 Design

The experiment used a single-factor between participants design. The independent variable, type of training, had four levels:

1. **Control.** The control group was given the LDR goal (which was to focus on producing batches of doughnuts that matched customer orders), and received basic rote training that walked through the call centre and doughnut-making tasks step by step (the training was presented via PowerPoint).
2. **Device model.** Like the control group, the device model group was given the LDR goal, but also studied the device model materials prior to being trained on the procedure. Their training was the same as for the control participants, but also visualized the current state of power flow before and after each device interaction, in a manner that was consistent with the device model they had studied.
3. **Machine tester.** The machine tester group was given the HDR goal, which was to evaluate the machine's visual feedback after each selector button was pressed. They were also instructed that while executing the doughnut-making task they still had to enter data in each machine component because the machine would not allow them to proceed without it, but that it was *not* necessary to enter data that matched the doughnut order. They were also given a description of two specific problems with

the machine’s visual feedback that they should look out for: that the machine sometimes provides *no* visual feedback after pressing a selector button, and that it sometimes provides the *wrong* visual feedback. Although neither problem actually occurred during the experimental trials, these examples were included to emphasize participants’ role as machine testers. Their training on the procedure was otherwise the same as the training for the control participants, except that the PowerPoint presentation included visual examples of the two problems they had been advised to look out for.

4. **Machine tester-enhanced:** After running the first four participants in each of the three groups (*see* above), it became clear that despite having only been exposed to the HDR goal, the machine tester participants were nevertheless focusing on the LDR goal of baking doughnuts to match customer orders. Therefore, a fourth level was introduced: these participants received the HDR goal and the same training as the machine tester participants, but were also instructed to enter the value of “1” (or some other arbitrary number) for the quantities in the data entry steps of the procedure. This was intended to reinforce the notion that matching the doughnut orders was not related to successful completion of their testing goal.

<b>Condition</b>	<b>Task-based goal</b>	<b>Additional instructions or information</b>	<b>Enhancements to the PowerPoint material</b>
<b>Control</b>	Low device relevance	None	None
<b>Device model</b>	Low device relevance	Studied the device model topology diagram, which described how power flowed through the system	Enhanced to visually show the flow of power through the system as steps were executed
<b>Tester</b>	High device relevance	Do not need to enter accurate data Instructed to look out for two specific feedback problems: no feedback, and the wrong feedback	Enhanced with visual examples of the no feedback, and the wrong feedback problems
<b>Tester-enhanced</b>	High device relevance	Instructed to enter arbitrary quantities in the data-entry steps Instructed to look out for two specific feedback problems: no feedback, and the wrong feedback	Enhanced with visual examples of the no feedback, and the wrong feedback problems

Table 1. Summary of the differences in the training material presented to the four groups.

The resulting experimental design was a single-factor between participants design with four levels: control, device model, tester and tester-enhanced, as summarized in Table 1. The first 12 participants were randomly assigned to the control, device model, and machine tester groups; after the introduction of the machine tester-enhanced group, the remaining 36 participants were quasi-randomly assigned to all four groups, resulting in a total of 12 participants per group.

The main dependent measure was the number of omission errors made on the device initialization step during execution of the doughnut-making procedure. Data about omissions on all other steps was also recorded, as well as the timestamp for each device interaction and the overall trial-completion time. Additional dependent measures that were of interest were the number of correctly specified doughnut orders and responses to a qualitative interview that was conducted after all trials were complete (described below).

#### 3.4.1 *Qualitative data: Understanding the effects of training*

A structured qualitative interview was administered to participants in all groups after completion of the experimental trials. The purpose of this interview was to allow a more in-depth examination of how the different types of training influenced task performance, and how this may have influenced the device initialization error rates.

We first asked participants to describe in detail how they fulfilled their task-based goal; it was expected that participants who acted on the LDR goal (control and device model participants) would be less likely to mention the device initialization step in their verbal descriptions as it was less relevant to their goal, while participants who acted on the HDR goal (tester and tester-enhanced participants) would be more likely to mention it. In order to identify whether the training process had any unexpected influence across conditions, we asked if any of the instructions or training material stood out or regularly came to mind as participants were executing the trials. We also asked participants what factors they had considered most important when they were executing the experimental trials, to help us assess how successful the training was in communicating the task-based goal, and whether in practice participants emphasized the corresponding goal. Finally, we asked participants whether there was anything particular to the interface or task design that caused significant frustration or annoyance. This was to help us assess whether there were extraneous factors that may have inadvertently drawn attention away from the primary task.

### 3.5 Procedure

Participants in all groups read the introductory, paper-based material that described their role (baker or machine tester) and corresponding task-based goal during the experiment (LDR or HDR).

After reading through the introductory material, a brief pre-training quiz was administered to participants in all groups to ensure that the information presented had been adequately understood and internalized. If a participant failed to answer a question

correctly, they were asked to re-examine the introductory material and the quiz was administered again.

Upon successfully completing the quiz, participants were provided with training on the procedural task, then moved on to the practice phase. Each participant executed the call-centre and doughnut-making tasks, until they had completed two trials in a row without difficulty. When errors were made, the computer issued a simple dialog with the message “An error has been made. Please correct it and carry on.” Participants had to press the OK button to dismiss the dialog, then detect and correct the error before proceeding. Very few participants made any errors at all during this practice phase.

Participants then completed ten experimental trials without the experimenter present, with the option of a break after the fifth trial. At the end of each trial, a task report was presented indicating either that the correct number of doughnuts had been made, or that the order was off by a given number of doughnuts. This task report was not based on the actual data entered by participants during the trial; a positive report was shown on half of the trials, and a negative report shown on the other half. During the experimental trials a warning message was not presented when a procedural error was made (e.g., omitting a step), but the interface was designed such that participants were unable to proceed to the next step until they detected and corrected the error.

After all ten trials were complete, the experimenter verbally administered the qualitative interview and recorded participants’ responses on paper. The whole procedure lasted approximately one hour.

## **4 Results**

All data was included in the analysis. Errors were counted for each of the ordered task steps in the main procedure. As in Li et al’s (2008) work, attempting to execute a step incorrectly or out of order was counted as an error for the associated step, but repeated clicks were not counted as multiple errors.

### **4.1 Overall errors**

The total number of procedural errors across all 48 participants was 305; 92 were committed by control participants, 81 by tester participants, 48 by tester-enhanced participants, and 84 by device model participants. Of the 48 participants, 31 made at least one DIE, and the DIEs accounted for 39.3% of the total errors.

The systematicity of errors, or *error rate*, at each task step was assessed by examining the number of error occurrences at a given step in relation to the number of opportunities for that error overall (see Byrne & Bovair, 1997, and Li et al, 2008). Since each of the 48 participants completed 10 trials, the total number of opportunities for each

error was 480. The error rate at each step is illustrated in Figure 5; Byrne and Bovair (1997) propose that error rates that are above the .05 (or 5%) level can be considered to occur systematically. Consistent with Li et al’s (2008) experiments, the device initialization step, which is the first step in the doughnut-making task procedure, exhibited the highest error rate across all task steps (overall error rate = .246), followed by the PC step, which is to press the “Process/Clean” button at the end of the task (overall error rate = .146), and both can be considered to have occurred systematically. The error rate for the *Fryer’s* selector step and the Dough Depositor data step were also found to be above the .05 level.

In the following sections we consider specific categories of errors in more detail.

*Figure 5 about here*

#### 4.2 Device initialization errors

Mean error rates for the device initialization step (i.e., pressing the Dough Depositor selector button at the beginning of the doughnut-making task) for each of the 4 experimental conditions are displayed in Table 2, and also depicted in Figure 6. The error rates were computed by dividing the number of DIES in each condition by the number of opportunities for that error.

	Mean Error Rate
<b>Control</b> ( <i>N</i> =12)	.308
<b>Device model</b> ( <i>N</i> =12)	.308
<b>Tester</b> ( <i>N</i> =12)	.267
<b>Tester-enhanced</b> ( <i>N</i> =12)	.100

Table 2. Mean error rates for the device initialization error, by condition.

A Kruskal Wallis test, appropriate for a between subjects design, allowing for comparisons between all conditions, was conducted. The results of the analysis indicates that there is a significant difference in the medians,  $\chi^2(3, N = 48) = 8.34, p = .039$ . The overall test was significant, so pairwise comparisons among the groups were performed.

Three follow-up tests were controlled for Type I error across tests using the Bonferroni correction (adjusted alpha levels of .0167 per test (.05/3)). No reliable difference was found between the control and device model groups (*Mann-Whitney U* = 69.5, *Wilcoxon W* = 147.5, *Z* = -.146, *p* = .884). This indicates that knowledge of the device’s conceptual model and the role of the initial selection step in the overall device operation does not provide a sufficient internal goal-based cue to prime the device initialization step at the appropriate time. Implications of this finding are discussed

below. A reliable difference was found between the control and tester-enhanced groups (*Mann-Whitney*  $U = 30.5$ , *Wilcoxon*  $W = 108.5$ ,  $Z = -2.527$ ,  $p = .011$ ), indicating that participants in the tester-enhanced group were able to develop superior internal cues to prime the device initialization step at the appropriate time, while the control participants were not. This result supports the hypothesis that relevance of the initial step to the task-based goal plays an important role in the likelihood of its omission during execution of a procedure, as execution of the initial step was critical to successful completion of the tester-enhanced participants' HDR goal but not to the control participants' LDR goal. However, no reliable differences were found between the control and tester groups (*Mann-Whitney*  $U = 60.5$ , *Wilcoxon*  $W = 138.5$ ,  $Z = -.674$ ,  $p = .500$ ), despite the tester participants being instructed to focus on the same task-based goal as the tester-enhanced participants. The implications of these findings are discussed in section 6.

*Figure 6 about here*

### 4.3 Categories of errors

Analyses were conducted in order to further explore the impact of the training manipulations on the cognitive salience of different types of steps during execution of the main doughnut making procedure. Errors made during the doughnut-making task were grouped into three categories: selector errors (errors made on any of the 5 selector steps, including the device initialization step), post-completion errors (errors made on the *clean* step, the last step of the procedure), and data-step errors (errors made on the remaining procedural steps, which were related to entering data for the doughnut orders). Table 3 shows the mean error rates across all experimental conditions.

	<b>Selector errors</b>	<b>Post-completion errors</b>	<b>Data-step errors</b>
	<b>Mean Error Rate</b>	<b>Mean Error Rate</b>	<b>Mean Error Rate</b>
<b>Control</b> ( $N=12$ )	.090	.233	.019
<b>Device model</b> ( $N=12$ )	.098	.192	.010
<b>Tester</b> ( $N=12$ )	.097	.100	.022
<b>Tester-enhanced</b> ( $N=12$ )	.027	.058	.043

Table 3. Mean error rates for selector errors, post-completion errors, and data-step errors, by condition (data not normally distributed).

A Kruskal Wallis test, indicates that there is a significant difference in the medians, for selector steps ( $\chi^2(3, N = 48) = 10.08$ ,  $p = .018$ ) and data steps ( $\chi^2(3, N = 48) = 10.51$ ,  $p = .015$ ), but not for post-completion steps ( $\chi^2(3, N = 48) = 2.18$ ,  $p = .537$ ).

For each group two follow-up tests were controlled for Type I error across tests using the Bonferroni correction (adjusted alpha levels of .025 per test (.05/2)). Using a

Wilcoxon Mann-Whitney rank-sum test, significant differences were found between the control and tester-enhanced groups: tester-enhanced participants exhibited a reliably lower error rate on selector steps overall (*Mann-Whitney*  $U = 28.500$ , *Wilcoxon*  $W = 106.500$ ,  $Z = -2.577$ ,  $p = .010$ ), and exhibited a higher error rate on data steps, but this was not a significant difference (*Mann-Whitney*  $U = 37.500$ , *Wilcoxon*  $W = 115.500$ ,  $Z = -2.037$ ,  $p = .042$ ). These differences in categorical error rates indicate that the tester-enhanced manipulation improved internal cues for priming the selector steps.

Again, no differences were found between the tester and control groups. The device model group also did not show any significant differences from the control group, despite the device model group having access to more sophisticated information about how the device-specific steps related to the internal structure of the system.

The quantitative results were supplemented by qualitative findings that validated the procedure followed (checking participants' understanding of the tasks they were performing).

## 5 Qualitative results

Responses during the qualitative interview were reviewed for all participants.

In response to question 1, which asked participants to describe in detail how they fulfilled their task-based goal, there were no distinguishable patterns to the responses when examined by condition. Over 80% of participants mentioned the execution of the device initialization step during their verbal descriptions of how they accomplished their goal: 50% explicitly mentioned it (e.g. "click the dough selector"), and a further 31% referred to it implicitly (e.g. "select the indicator for each to see if it's working"). This confirms that DIEs were not knowledge based, and indicates that participants in all four conditions perceived the relevance of the device initialization step to be similar when considering the task in retrospect.

Similarly, there were no notable differences in responses when participants were asked whether any instructions or training material regularly came to mind while executing the trials. This suggests that the training process did not inadvertently emphasize parts of the procedure in ways that might have unexpectedly influenced performance across the conditions.

The third question asked what participants considered most important while executing the trials; the top three responses per condition are provided in Table 4 (note that participants were not restricted to a single answer). The responses indicate that the training was effective in communicating the task-based goal, and that in practice participants correctly emphasized the corresponding goal: control and device model

participants primarily emphasized the need to accurately enter the data presented in the doughnut orders (the LDR goal), while tester and tester-enhanced participants primarily emphasized the need to test the relevant feedback (the HDR goal).

Rank	Control	Device Model	Tester	Tester-enhanced
1	Accuracy of data entry (7 responses)	Accuracy of data entry (8 responses)	Testing the machine's feedback (9 responses)	Testing the machine's feedback (7 responses)
2	Getting the math / quantities / calculations right (5 responses)	Speed (5 responses)	Accuracy of data entry (7 responses)	Speed (4 responses)
3	Speed (5 responses)	Use of energy / power settings (4 responses)	Speed (3 responses)	Following the procedure in order (3 responses)

Table 4. The most frequent responses to question 3 in the post-study interview, which examined what participants emphasized during the experimental trials.

The final question in the post-study interview looked at whether anything particular to the interface or task design caused frustration or annoyance in a way that may have drawn attention away from the primary task. Most responses to this question were related to general concerns about how data was handled (e.g., that there was no consistency between how quantities were entered, and that the labels in the order sheet did not match the corresponding component labels where the data was entered). We do not consider this data further in this paper.

## 6 Discussion

The overarching goal of the reported work was to identify whether device initialization errors can be mitigated without making physical changes to the device, through the emphasis of certain information prior to learning the device's operating procedures. After learning a conceptual model there was evidence of a change in participants' resulting behaviour (as described below); however, this did not translate to a reduction in the DIE rate for participants working towards a task-based goal. Significantly fewer errors only occurred when participants were instructed to perform a task that required the same sequence of task actions, but had a fundamentally different goal structure.

### 6.1 Interpretation of tester-enhanced and tester results

While all participants executed the same procedural steps in the same order, it was expected that the tester-enhanced and tester participants, who received the HDR goal description during training, would make fewer DIEs during the experimental trials;

execution of the device initialization step (i.e., pressing the Dough Depositor selector button and evaluating the machine's feedback) was necessary in order to correctly report on the accuracy of the device's visual feedback associated with component activation.

A significant difference was found in the error rates between the control and tester-enhanced groups. This indicates that it is possible to achieve a reduction in the DIE rate without physically changing the device interface. A more fine-grained analysis of the data allows a better understanding of these results and their implications.

Tester-enhanced participants were instructed to enter an arbitrary number for the quantities in the data entry steps in order to reinforce the notion that matching the doughnut orders was not related to successful completion of their testing task. Inspection of the data log files indicates that during the experimental trials, participants in this condition only entered data that matched the order on 9 trials out of 120. This confirms that the tester-enhanced participants did not focus on the task-based goal of accurately making doughnuts. Further, responses to question 3 (the *emphasis* question) indicate that they correctly focused on the intended task-based goal of testing the selector feedback.

Focusing exclusively on the high-level task-based goal of testing the selector feedback, rather than focusing on this goal in addition to the goal of accurately filling doughnut orders, appears to have led to an increase in the relevance of different procedural steps, most importantly the goal associated with the device initialization step. It could be argued that the nature of the task was changed so that individual procedural steps were no longer subgoals. Their execution did not cumulatively contribute towards the achievement of a high-level goal. Another possible explanation is that not performing mathematical calculations reduced the working memory load, which consequently reduced the error rate.

The observed differences in error rates for the broader categories of selector and data-entry errors between control and tester-enhanced participants also provides more general evidence for the changes in cognitive salience of different subgoals in the procedure, suggesting that not only can a goal's salience be increased by making it more relevant to the task-based goal, but also that a goal's salience can be decreased. The fact that a similar effect was not observed for the tester participants highlights that a critical difference appears to have been that tester-enhanced participants focused on a single task-based goal. It also shows that increases in the cognitive salience of the device initialization step are extremely sensitive, as both groups were exposed to very similar training prior to the experimental trials.

A non-significant difference was found between the DIE rates in the control and the regular tester group, suggesting that directly linking the device initialization step to

accomplishment of the task-based goal is not an effective strategy for elevating the salience or activation of that step. However, closer examination of the available data provides further insight into the null result.

Data from the third question in the qualitative interview (the *emphasis* question; see Table 4) of the post-study interview reveals that the same number of participants in both the control and tester condition reported that they personally emphasized accuracy of data entry during the experimental trials, even though the tester participants were not required to do so. This suggests that while participants in the tester group did focus on the primary goal of testing the machine's feedback (as indicated by the fact that this was the most frequently reported answer to question 3), they also emphasized the same goal as those in the control condition (which was to produce the correct number of doughnuts). This implies that the tester participants actually acted on the same goal as the control participants, and simply *added* the goal of testing the selector feedback. Precisely why participants were so strongly drawn to the data entry tasks should be explored in future work, but the task report presented at the end of each trial, which regularly reported that the incorrect number of doughnuts had been made, might have encouraged participants to focus on correctly entering data.

This suggests that taking on the *additional* goal of focusing on the outcome of the device initialization step (in this case, the resulting visual feedback) does not on its own provide a sufficient internal cue to prime the step at the appropriate time during task execution. Further, the data demonstrates that device users might readily adopt additional goals that were not emphasized in their training on the device, especially if the device design itself naturally encourages it.

## 6.2 Interpretation of conceptual device model results

No difference was found between the control and device model groups on the DIE rate. Learning a conceptual model that emphasizes the role of the device initialization step does not result in a sufficient internal cue to prime the step during execution of the procedure. Furthermore, no differences were found between the two groups on the broader error categories of selector, post-completion, and data-step errors, indicating that knowledge of the device model provided no performance advantage during execution of the procedure.

Despite this, participants in the device model condition exhibited signs that the information provided in the device model did affect the task representations formed. In their responses to question 1 in the qualitative interview, participants in the device model

condition spoke primarily in terms of power flow when describing how they accomplished their goal (e.g., they spoke of “activating power to” the component, or “changing the power”). This is in contrast to participants in the other three conditions, who spoke primarily in terms of “clicking”, “pressing”, or “selecting” the selector button, which are more interface-level descriptions of the actions. This suggests that the concept of power flow, and the specific role that the selectors played in controlling power flow was understood, although this did not result in an improvement in their performance. While having detailed knowledge of a system’s internal mechanisms has been shown to improve performance in novel problem solving tasks (Halasz & Moran, 1983), and in learning and retaining the operating procedures for a device (Kieras & Bovair, 1984), it does not appear to have an effect on error tolerance in procedural tasks, especially on the likelihood of committing device-based omission errors.

## **7 Conclusions**

While the results from this study suggest that the goal a participant focuses on can influence their performance, they also demonstrate that when multiple task-based goals related to a given procedural task are present, it is difficult to predict which one(s) participants will choose to focus on, and it is also difficult to encourage them to focus on one over another. We found that training was inadequate; it was only possible to reduce the error rate using a task goal modification that changed task performance.

A change in goal focus fundamentally changed the task in a way that eliminated or reduced the influence of a different contributing factor. For example, by not focusing on the goal of accurately filling doughnut orders, the tester-enhanced participants also did not have to expend the effort to transform information in the doughnut-making machine’s Order Sheet to enter it into the corresponding data fields. While for the most part the data transformation was rather straightforward (as simple as entering data from the “shape” column into the “puncher” component), the task was repetitive and required participants to continuously refer back to the Order Sheet to retrieve different parts of the order. Although participants in the tester-enhanced condition executed the identical procedure to control participants in terms of interacting with the device, the cognitive load imposed by the task was less demanding as there was no need to engage working memory in order to find the relevant data in the Order Sheet and transfer it into the correct component fields. Working memory demands imposed by the task environment have been shown to have a significant impact on the occurrence of PCEs (Byrne & Bovair, 1997), which appear to share some common characteristics with the device-initialization error. As such, the possibility that a reduction in the working memory demands for participants in the tester-

enhanced condition may have contributed to the corresponding reduction in the device-initialization error rate must also be considered. This issue should be pursued in more detail in subsequent research.

### 7.1 Extending the understanding of errors

Byrne and Bovair (1997) deliberate at length on how to classify the post-completion errors that were the focus of their study. Ultimately, they conclude that PCEs are probably slip errors, but that extant accounts of error “simply do not provide enough explanatory power to make specific predictions”. Similarly, it is difficult to develop a satisfactory account of the findings of this study based on an established framework such as S-R-K (Rasmussen, 1987) because such a classification does not provide appropriate explanatory power. The work reported here builds on the tradition of Byrne and Bovair (1997), Chung and Byrne (2008), and Li et al (2008), but shifts attention from PCEs to another class of interaction slips, namely device initialization errors (DIEs), and investigates different causal factors. A common feature of this study and those cited is that the tasks are familiar, so the error is not due to lack of knowledge but to a failure of attention, and the task steps on which the slips occur are device rather than task relevant (Cox and Young, 2000). Earlier studies have highlighted the influence of working memory load (Byrne and Bovair, 1997), timeliness of feedback (Byrne and Davis, 2006), visual cueing (Chung and Byrne, 2008), interruptions (Li et al, 2008) and opportunity to reflect (Back *et al*, submitted) on such procedural slips. Earlier studies have also shown that motivation has minimal effect on procedural slips (Back *et al*, 2006; Byrne & Davis, 2006). The study reported here has highlighted that many obvious forms of training also have minimal effect on these errors. Further studies are needed to investigate the interrelationships (if any) between the factors that influence the rate of procedural slips in practiced tasks. The findings of laboratory studies such as these also need to be compared with those of situated studies that investigate the broader contextual factors influencing slips, including work organization and evolution of performance over time.

### 7.2 Implications for training and device design

The results reported here have important implications for the role of training material in guiding the way people interact with devices, by demonstrating its potentially small role in participants’ actual use of a device. Specifically, the analysis of user behaviour in the tester condition shows that regardless of the task-based goal and procedural steps that are emphasized in training, individuals may inadvertently be drawn to adopt additional goals. This demonstrates the power that a device’s design can have over defining user behaviour, and also that it may be very difficult to encourage users to interact with a

device as *intended* rather than as *designed*. If a system doesn't fully support a user's needs, or is designed to support multiple different tasks, this research suggests that training users to operate the features specific to different tasks effectively will be very challenging. This is an important consideration, particularly for systems in safety-critical situations.

The substantially different behaviour observed in the tester-enhanced condition versus the tester condition does show that a difference in focus (and corresponding behaviour) can be "unlocked" with only minor adjustments to the training; however, identifying the critical information to provide is the challenge, and this may not be consistent with the intended use of the device.

The need for a device's design to match its intended uses is reminiscent of Norman's (1983b, 1988) argument for a correspondence between a system's conceptual model and the user's mental model. The results from this study show that learning a conceptual device model, which emphasizes the importance of performing error-prone actions, does not result in improved error performance. This highlights the need for the device design to be based on specific user needs and intentions from the very beginning.

In summary, the results from this research extend our current understanding of omission errors during procedural tasks, particularly device initialization errors, as well as factors that can influence their occurrence. This research provides insight into practical ways of managing such errors, showing that careful design is essential for avoiding such errors in general. One area that was not investigated in the current study was the long-term effects of the training interventions. Another is the effects of other task and interface design decisions on performance. Further studies, both in the laboratory and in more realistic settings, are needed to better understand the nature of the device initialization error, and the effects of different types of training interventions on slip error performance.

## Acknowledgments

We are grateful to Simon Li, who developed the Doughnut Machine experimental paradigm and gave permission to use it; to Simon Li and Mike Byrne for constructive feedback on an earlier draft of this paper; and to all participants in this study. Jonathan Back and Ann Blandford were supported by EPSRC grant GR/S37494.

## References

- ALTMANN, E. M. & TRAFTON, J.G. 2002. Memory for goals: an activation-based model. *Cognitive Science*, 26, 39-83.
- AMENT, M., BLANDFORD, A. & COX, A. (2009) Different Cognitive Mechanisms Account for Different Types of Procedural Steps. *Proc. Cognitive Science 2009*.
- BACK, J. BLANDFORD, A., FURNISS, D. & CURZON, P. Submitted. Avoiding slips. Submitted for journal publication.

- BLANDFORD, A., BACK, J., CURZON, P., LI, S.Y.W., & RUKSENAS, R. 2006. Reasoning about human error by modeling cognition and interaction. *Proc. Resilience Engineering Symposium, Juan les Pins, France*.
- BLANDFORD, A. & RUGG, G. 2002. A case study on integrating contextual information with usability evaluation. *International Journal of Human-Computer Studies*, 57.1, 75-99.
- BYRNE, M. D. & BOVAIR, S. 1997. A working memory model of a common procedural error. *Cognitive Science*, 21, 31-61
- BYRNE, M. D. & DAVIS, E. M. 2006. Task structure and postcompletion error in the execution of a routine procedure. *Human Factors*, 48.4, 627-638.
- CHUNG, P. H. & BYRNE, M. D. 2004. Visual cues to reduce errors in a routine procedural task. *In Proceedings of the 26<sup>th</sup> Annual Conference of the Cognitive Science Society*.
- COX, A.L. & YOUNG, R.M. (2000). Device-oriented and task-oriented exploratory learning of interactive devices. *In Proc. Third International Conference On Cognitive Modeling*, The Netherlands: Universal Press, 70-77.
- GRAY, W.D. 2000. The nature and processing of errors in interactive behavior. *Cognitive Science*, 242, 205-248
- GRAY, W. D. 2004. Errors in interactive behavior. In W. S. Bainbridge Ed., *Encyclopedia of Human-Computer Interaction*, 230-235: Berkshire Publishing Group.
- HALASZ, F.G., & MORAN, T.P. 1983. Mental models and problem solving in using a calculator. *In Proceedings of CHI '83 Human Factors in Computing Systems*. New York: ACM.
- HUTCHINS, E. 1995. *Cognition in the wild*. Cambridge: MIT Press; 1995.
- KIERAS, D.E., & BOVAIR, S. 1984. The role of a mental model in learning to operate a device. *Cognitive Science*, 8, 255-273.
- LEE, W. O. 1992. The effects of skill development and feedback on action slips. In A. Monk, D. Diaper and M. Harrison (Eds.) *People and Computers VII*. 73-86: Cambridge University Press.
- LI, S., BLANDFORD, A., CAIRNS, P. & YOUNG, R. M. (2008). The Effect of Interruptions on Post-Completion and Other Procedural Errors: An Account Based on the Activation-Based Goal Memory Model. *Journal of Experimental Psychology: Applied*, 14.4, 314-328.
- NORMAN, D.A. 1981. Categorization of action slips. *Psychological Review*, 881, 1-15.
- NORMAN, D.A. 1983a. Design rules based on analyses of human error. *Communications of the ACM*, 264, 254-258.
- NORMAN, D.A. 1983b. Some observations on Mental Models. In A.L Stevens & D. Gentner Eds. *Mental Models*. New Jersey: Lawrence Erlbaum Associates, Inc.
- NORMAN, D.A. 1988. *The Design of Everyday Things*. New York: Basic Books.
- RASMUSSEN, J., 1987. The definition of human error and a taxonomy for technical system design. In: Rasmussen, J., Duncan, K., Leplat, J. (Eds.), *New Technology and Human Error*. Wiley, Chichester, 53-62.
- REASON, J. 1990. *Human Error*. Cambridge, UK: Cambridge University Press.
- REASON, J. 2002. Combating omission errors through task analysis and good reminders. *Qual Saf Health Care*, 11, 40-44.
- SELLEN, A., KURTENBACH, G. & BUXTON, W. 1992. The prevention of mode errors through sensory feedback. *Human-Computer Interaction*, 7, 141-164.

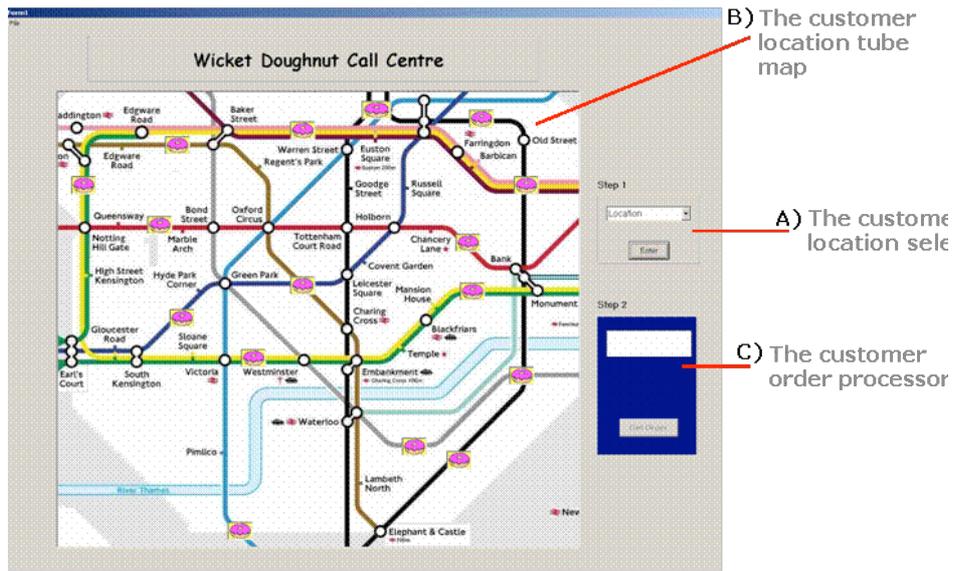


Figure 1. The Wicket Doughnut Call Centre interface used by Li (2006) to simulate a call centre for a doughnut-making operation. Participants used the interface to retrieve incoming orders from different customer locations.

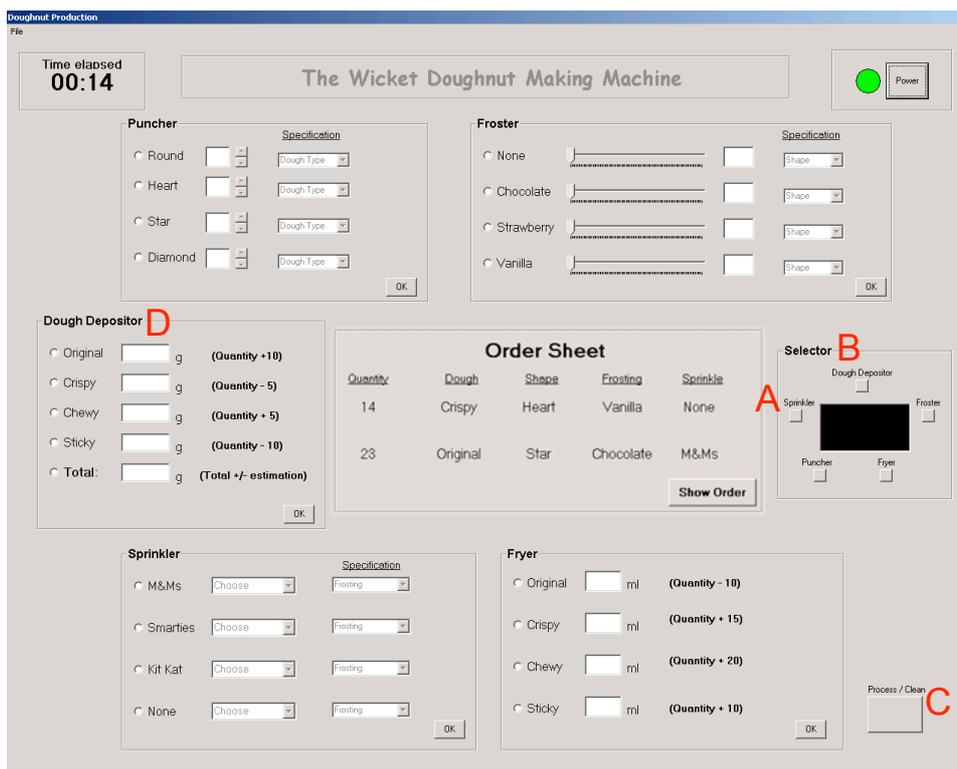
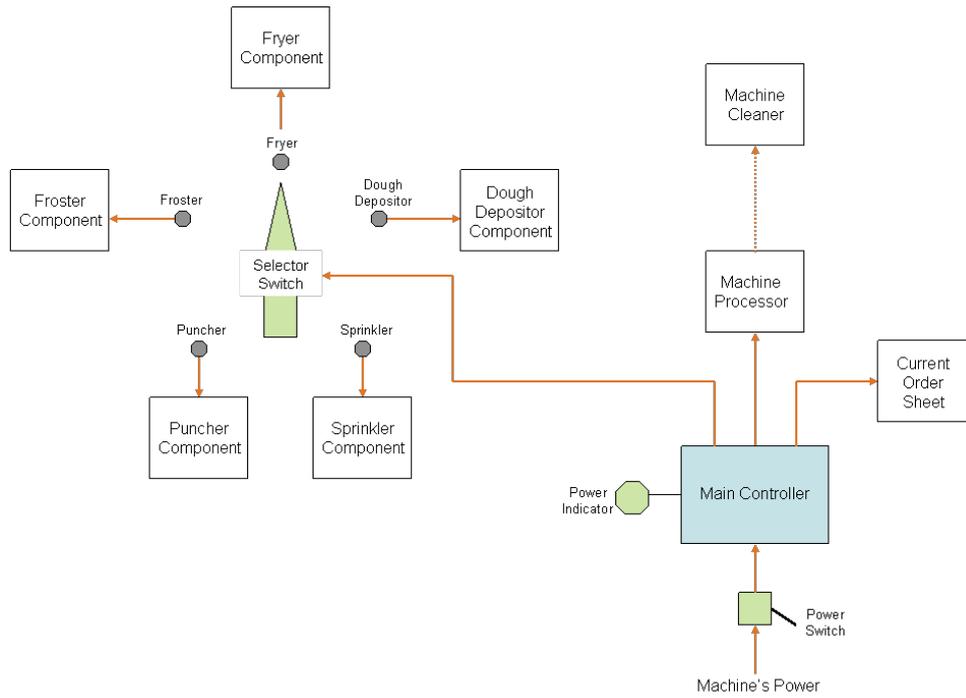


Figure 2. The Wicket Doughnut Making Machine interface.



**Figure 3.** System topology representation of the device model for the doughnut-making machine



**Figure 4.** A sample of the visual feedback displayed each time a new component was activated by pressing its selector button.

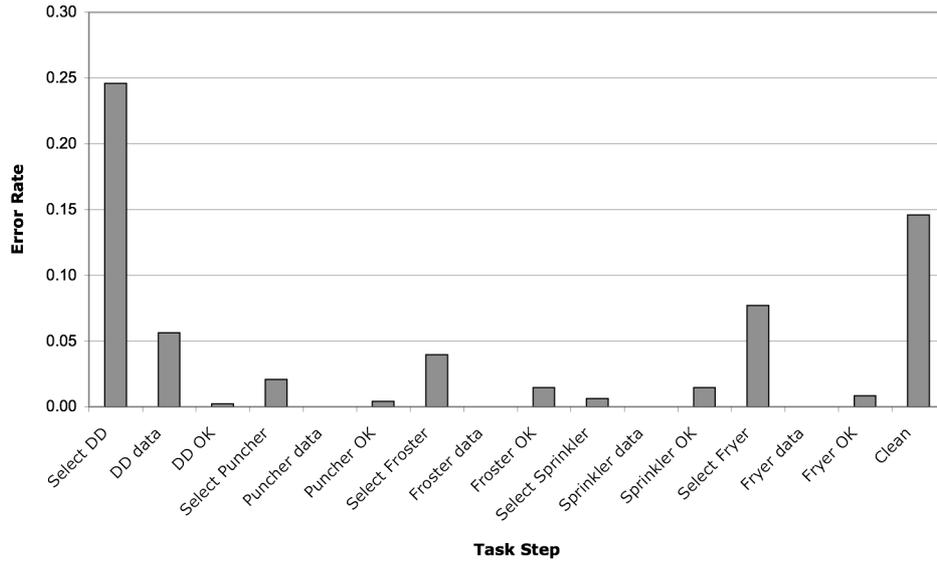


Figure 5. Error rates for task steps in the main procedure, collapsed across conditions. Error rates above .05 suggest a systematic error.

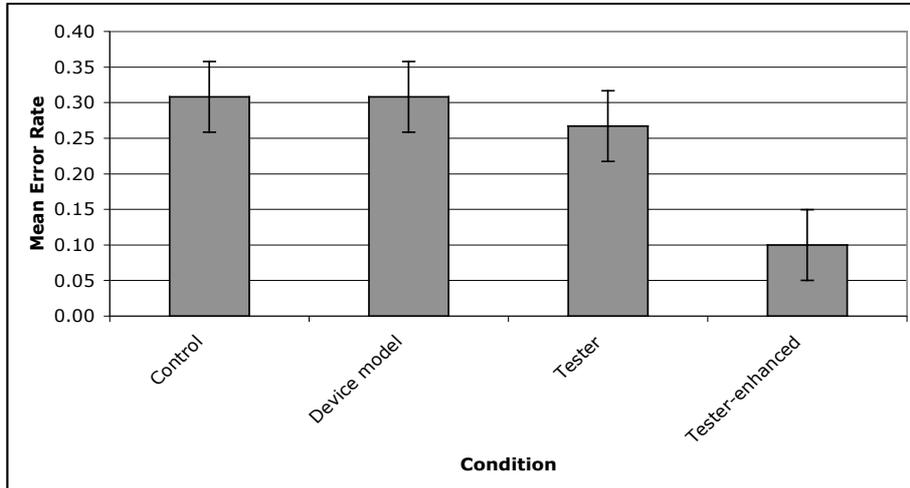


Figure 6. Mean error rates for the device initialization step in each condition, with error bars identifying the corresponding standard error.