CHOLLA FUNCTIONAL SPECIFICATION

by Richard M. Barth

Version 1.0

May 1981

This is the functional specification for Cholla, the information system for the wafer fabrication line in the Integrated Circuit Laboratory.

XEROX
Palo Alto Research Center
Integrated Circuit Laboratory
3333 Coyote Hill Road   Palo Alto   CA   94304

Table of Contents

CHOLLA FUNCTIONAL SPECIFICATION

1.0 Introduction

Cholla is the system that provides computer assistance to the silicon wafer processing facility in the Integrated Circuit Laboratory of the Xerox Palo Alto Research Center. An overview of the system is available [1].

This is the functional specification for Cholla. It is the formal vehicle that explicitly specifies what services Cholla will provide. The world that Cholla must deal with, the user interfaces Cholla provides, and the sequence Cholla will be implemented in will be described. The requirements presented are derived mainly from interviewing the operations people in ICL. Other sources are listed in the bibliography at the end of this specification.

Cholla is an information management and control system that enables the processing staff to easily take requests for processing, masking information and raw material to produce finished wafers with a database that contains information related to particular processing requests as well as more general information describing the fabrication environment at the time that processing occured.

This specification is intended to be understood in detail by both people accustomed to the processing world as well as those familiar with the computer world. As such it contains a large amount of detail that could just be assumed if it was intended for only one group or the other. This specification is not intended to describe the details of how Cholla is implemented but simply the functionality that it will provide.

2.0 World Model

This section describes the environment that Cholla deals with by defining the objects and people that appear in the processing line world and describing how those objects are dealt with by said people.

The people that Cholla must deal with are process line technicians, equipment maintenance technicians and engineers, process line manager, and process engineers. Each of these types of people have different tasks that they perform and will have different needs that Cholla must serve if it is to be of use to them. The rest of this section will describe the objects that these people deal with and then describe how each type of person must manipulate these objects in a different manner in order to carry out their particular responsibilities in the overall task of manufacturing integrated circuits.

2.1 Processing Line Environment

The processing of integrated circuits is a step by step procedure that converts a blank piece of silicon, called a *wafer*, into a wafer which has a number of die containing many patterned layers of various materials such as diffused silicon, silicon dioxide, polysilicon, and aluminum. The pattern that each of these layers is formed into is defined by a *mask*. A mask is a piece of glass that has essentially a photographic image of the pattern for a specific layer on its surface. A *step* uses some equipment according to a specification and records the results in a step descriptor.

A *specification* is a piece of prose that indicates what is to be accomplished by the step, what material and equipment are required and how to do it. Portions of a specification will be in a form that is easily interpreted by machine, e.g. target temperature is 1000 degrees, while other portions will only be read by humans, e.g. wear a face mask while performing this step.

A *step descriptor* is mainly used to record the results of a step. It will generally have a number of fields to be filled in that require specific numbers to be entered, e.g. actual temperature readings, a comments section where anything unusual that happened may be entered, e.g. wafer 5 had purple spots after the plasma descum, and it may have a parameters section that provide values required by the specification.

There are two major forcing functions that cause steps to be performed. Satisfying a processing request causes a series of process steps to be performed. Generally a processing request will ask for a group of wafers to be processed exactly, or nearly, the same. This group of wafers that have a particular set of masks and a particular processing sequence associated with them and that move through the fabrication line together is called a *lot*. A *step list* is the detailed description of the order of processing steps required to process a lot. The other function that causes steps to be performed is line maintenance. *Maintenance steps* may need to be executed because a certain amount of time has elapsed, a certain amount of usage of a piece of equipment has occured, or a piece of equipment has failed.

Process steps take a group of wafers, called a *batch*, and perform some operation on them. If a lot is subdivided into sections for a single step those sections are called *splits*. A batch is normally a single lot but may be a collection of lots or splits. There may be multiple batches using a piece of equipment at the same time. The distinguishing feature of a batch is that only a single step descriptor is filled out to record the results of that process step. A batch of wafers may contain additional wafers, called *test wafers*, that do not enter or exit the processing sequence in the same way that the device wafers do.

These test wafers are normally used to measure the result of a step. Data recorded in the step descriptor may be stored in multiple locations. Typically the lot histories of all lots in the batch processed by the step will be updated and the equipment log for the equipment used will be updated.

Lot histories record data that is specific to a particular lot. The data recorded here will be used in conjunction with data that is produced by parametric tests in order to correlate final device characteristics with process parameters. The software needed to correlate parametric test data with fabrication data is outside the scope of the Cholla project.

All wafers have a unique identifier scribed on the back that identifies the lot the wafer is contained in as well as the particular wafer within that lot.

Equipment logs help to characterize the performance of a particular piece of equipment. They generally come in two forms. One is simply a piece of text that records everything that happens to the machine. Another keeps track of specific variables as a function of time or usage and displays these as a 2 dimensional plot. These plots are generally referred to as *control charts*.

All splits of a lot must go through the same set of steps together with only changes in parameters in the step descriptor allowed. Having no processing done on a split during a step is allowed. If gross processing variations are required that cause steps of the process to be performed at different times for different wafers within the lot then the lot must be divided into separate lots which are tracked independently. Splits are generally used for performing controlled experiments to improve the process definition.

Some of the machines will have automated interfaces. These machines will be small in number compared to the total number of machines on the line. Each of these machines will be assigned to a particular workstation.

Process line technicians have responsibility for actually moving lots through the steps specified in the lot step list. *Equipment maintenance technicians and engineers* are responsible for specific pieces of equipment. They must install, characterize, and maintain the equipment. For particularly complex pieces of equipment they are also responsible for operating them rather than the process line technicians. The *process line manager* is responsible for overall management of the line. He must schedule and set priorities for the lots, trouble shoot problems in the line, and maintain the database of specifications, step descriptors, and step lists. *Process engineers* are responsible for trouble shooting problems in the line and for developing new processes.

These people interact in very complex ways. To ease the definition of the problem Cholla is required to solve the above definitions are overly simplistic. In actuality the operation of a processing line is a group effort with a lack of clear cut division of responsibility. The intent of the definitions is to describe what each person does the majority of the time so that the system will be designed such that any user can accomplish his/her tasks with a minimal amount of effort.

The handling of a process request procedes as follows. Generally the process line manager will receive a request for processing. This may be for a standard process so that he simply instantiates a copy of a generic step list for that process. If it is from a process engineer it will usually be for some special processing in which case the request may include the step list or the line manager may have to edit an existing generic step list to insert the peculiar requirements needed. In any case he will generate a step list for the request and post a message that the lot is ready for processing. The technicians will follow the step list to process the wafers, calling on any of the rest of the people for assistance as needed. When the wafer processing is finished the wafers are sent to the test area for parametric and functional testing after which they exit the Cholla environment.

There are two methods for assigning the work required to perform the processing to the technicians. One method is the model maker approach in which a lot is assigned to a specific technician who performs all of the steps in the recipe for the lot. The other method is the specialist approach in which each step of the process is assigned to a specific technician who becomes expert in performing that step. Cholla will be able to accomodate both of these approaches gracefully.

The following sections describe in greater detail what each of the people do in order to fulfill their roles in the fabrication process.

2.2 Process Line Technician's World

A process line technician performs the job of taking blank wafers and turning them into finished wafers by performing a series of steps using various pieces of equipment. A technician performs a step by walking up to a workstation, deciding what step needs to be done, reading the specification and step descriptor that is presented, performing the actions described, and recording the results of those actions. Typically there will be a fair amount of time that the workstation is idle while the actions are performed before the results are recorded. Variations from this scenario may occur, e.g. the technician may already know what step needs to be worked on or may record results while performing the actions, but most of the time the given sequence will be followed. Many of the steps will have accept/reject criteria that monitor the quality of the processing as it takes place. At any step all or part of a lot may be rejected causing a rework cycle or the lot to be discarded. A *rework cycle* may cause some extra steps to be inserted into the process flow, e.g. stripping photoresist off the wafers, then the lot goes back to some previously completed step and continues sequential processing of the steps specified by the step list from there. On occasion deviations from normal processing will require that steps further on in the step list have their parameters changed, e.g. extended time growing an oxide may require that the contact etch time be increased.

The step that a technician decides to work on depends on several factors. The lots may have different priorities so that if there are a set of lots ready for a technician to work on them the technician may need to pick a particular one based on its priority. Many of the steps in a process sequence require the technician to do something to get a machine started followed by a large amount of free time while the machine continues to process the lot so that the technician is free to perform other duties. However the timing of these steps often must be precisely controlled so that if one of these steps is about to complete the technician must work on it regardless of the priorities the lots have. Since the equipment used for integrated circuit manufacturing is usually pushing the state of the art it constantly requires maintenance, often unscheduled maintenance. This affects the availability of the equipment for processing and thus the choice of which step to work on.

2.3 Equipment Maintenance Technician's World

There is a large number of machines that must be installed, characterized, and maintained by a relatively few number of people. To ensure that this is done in an orderly manner so that good control is maintained over what is happening in the line, log books, maintenance schedules, and operating and maintenance specifications must be used. Status of which machines are down and why they are down must be readily available. Reminders should be presented when equipment requires maintenance.

Characterizing a piece of equipment consists of performing tests on it to establish operational limits in terms of precision, accuracy, and throughput, as well as determining maintenance and operating procedures. The characterization of a piece of equipment may be specific to the particular process step that is being run on it, requiring the equipment to be recharacterized if the use of the machine changes significantly from its normal operating point. The results of characterization are made known to the Cholla system by changing a specification or creating a new specification.

2.4 Process Engineer's World

A process engineer must be able to relate device characteristics to process line variables in order to determine where processing needs improvement, to assist in tracking down malfunctions in the processing line such as dirty water supply leading to high pinhole defect densities, and to create new processes. Cholla will not be designed to support this function directly. However the process engineer must be able to browse through the collection of specifications, step lists, and step descriptors in order to compose new processes. He must be able to submit new copies of these objects to the line manager for inclusion in the standard list of them.

2.5 Process Line Manager's World

The process line manager is responsible for overall control of the line. His responsibilities include activities that he performs on a more or less regular basis as well as troubleshooting when the unexpected occurs. He must keep track of how smoothly lots are moving through the line, ensure that the equipment is up and running, plan for modifications to the equipment on the line, maintain the specifications, step descriptors and step list database, and perform any other duties necessary to keep the line rolling. Much of the actual work is done by people who work for him, his role being coordination and direction.

3.0 General Requirements

This section describes some general attributes that Cholla should have.

The system should never prevent someone from doing something out of the ordinary. This means that it should be easy to operate equipment in nonstandard ways without losing the record keeping assistance that Cholla provides.

The user should never be left waiting for a workstation to do something for more than a few seconds without some indication that progress is being made. All operations which take more than a few seconds should have an abort mechanism that allows the user to indicate that he would like the system to stop doing whatever he requested so that he can do something else.

It should be a well documented, straight forward procedure to hook a new piece of equipment into Cholla. This means that the interface that Cholla provides to equipment should be sufficiently general that no major reworking is required to handle a new piece of equipment. This says nothing about the complexity of connecting to a machine because of its own peculiarities.

A single hardware failure anywhere in the system should lose an insignificant amount of state information, should not seriously degrade the performance of the system as a whole and should allow quick recovery by replacing the failed host by another host, either by making another host combine its own functionality with the failed host's (degrading performance) or by replacing the failed host with another drawn from outside of Cholla.

It is important to move all the information recording and display functions into the automated system for a number of reasons. Keeping all the information online eliminates losing data or making it hard to retrieve and use for analysis of what actually was happening when a lot was processed. It eliminates the need for dust producing paper. It provides a single data collection point, thus eliminating the need for redundant recording of data by the operators, leaving them with more time to actually do the wafer processing. It coordinates updating the database so that inconsistency is not introduced by people changing multiple copies. Finally it provides tighter control over the whole manufacturing process leading to better characterization of the process which eases the technology transfer problem, allows more efficient utilization of the equipment and people, and provides faster turnaround.

The system should endeavor to provide an audit trail for all wafers so that the question "who did what to this wafer and when" can be answered easily for any wafer.

There will be a limited number of workstations available compared to the number of people who may be working in an area. Cholla will be designed to allow a single workstation to be multiplexed amoung these people in an efficient manner.

All documentation for the system will be stored online in a form that is easily accessible. Both tutorial and expert information should be available.

Much of the information handled by the Cholla system is proprietary. All access should be restricted by a password authentication mechanism. Hardcopy should only be available to certain individuals and the hardcopy should have Xerox private data stamps and serialization information placed upon it. It will be assumed that the processing line area itself is a secure environment so that the process line technicians do not have to continuously reauthenticate. They may be required to log in and out of the processing area.

4.0 System Decomposition and Use

This section presents an overview of the manner in which the Cholla system is broken up and then presents a very vague example of how it will to used in order to establish the context that the rest of the description is assuming.

4.1 The Interfaces

The detailed functional description of Cholla will be broken into several interfaces. The people in the processing world will use some combination of these interfaces in order to carry out their responsibilities. An *interface* is a group of facilities that are provided to accomplish some particular task. The realization of an interface can be thought of as being very similar to a window in Laurel in that there is a group of command buttons that cause actions to happen, there are parameter collection areas that gather the parameters needed by the commands, there is an exception area that is used to provide feedback to the user when the system does not understand what it is the user is trying to tell it to do, and there are areas for displaying and editing the object type the interface deals with. The interfaces should be functionally orthoganol in order to minimize the amount of code that must be implemented, documented, learned by users, and maintained. The combination of them that each type of person in the processing world uses should work well together to minimize the amount of work that they do. There will undoubtedly be other tools and interfaces that are used by both the users and implementors of Cholla to accomplish their tasks but those will not be described here. In particular everyone will be using the electronic mail facilities and there will be a very low level maintenance tool for use by Cholla system wizards.

The *status interface* provides a summary of conditions relating to the processing line, particularly the status of lots and equipment. It provides for display and editing of the *bulletin board*, which is simply a collection of textual messages. The *step list interface* is used to edit and inspect the contents of step lists. The *step execution interface* is responsible for displaying a specification and step descriptor and recording results in the step descriptor for subsequent storage in the database. Database access to maintain the specifications and step descriptors is provided in this interface.

Process line technicians will need to use the status and step list interfaces to determine what to do and the step execution interface for how to do it as well as recording what was done. Equipment maintenance technicians and engineers will use the status and step list interfaces to determine what machines need servicing and the step interface to get the detailed specification of how to do it as well as recording what was done. Both of these groups of people will assist in determining the specific information the system should present. They should view Cholla as a bookkeeping and reminder service.

The line manager will need the step list and step interfaces to keep the documentation database up to date. He will also need the status interface to keep him abreast of what is happening in the line and to post messages about general line status.

Process engineers will need to browse through the database in order to compose new processes. This will require using the step list and step interfaces, but with less privileges to change things in that database than the line manager has.

Each of the interface description sections show the functions that a particular interface provides in terms of the capabilities the user has, how the database is modified, and how the interface is anticipated to be used.

4.2 An Example

An example of the use of Cholla will make the comprehension of the following sections much easier. A simple example will be presented here to give a flavor for what Cholla will do rather than exploring every nook and cranny of the system. Suppose that the process line manager receives a message via the electronic mail system requesting 20 more mpcxyz wafers. Presumably the line manager knows that the process is NSIL-II and knows what the mask identifiers are that are required to satisfy this request. He invokes the step list interface, copies the definition step list for NSIL-II, and edits the copy to insert the mask identifiers and the number of wafers required. He invokes the status interface and posts a message that indicates that the new step list exists and is ready for processing to begin. The processing technicians see this message posted and invoke the step interface for each step in the step list, recording results as processing procedes. When processing is completed the wafers are delivered to the customer, the information recorded for the lot is placed into long term storage, and Cholla removes all information dealing with the lot from its database and the bulletin board.

5.0 Database Definition

The purpose of this section is to define the objects in the Cholla database. To ease the explanation a specific implementation will be sketched. This sketch is meant for illustrative purposes only and is not necessarily the way in which the database will actually be implemented.

All the objects in the Cholla database are variable length sequences of characters, much like a Laurel message. Every object has a name it is referenced by, just like the title of a book. Comments may be sprinkled freely throughout any of the objects. There are three basic types of objects that Cholla must deal with. They are step descriptors, specifications, and step lists.

The format of step descriptors and specifications consist solely of attribute-value pairs just as the header of a Laurel message contains the "To:<value>" or "Subject:<value>" attibute-value pairs. Some attributes are required by Cholla such as who last edited the object and when the edit occurred. The distinction between specifications and step descriptors is made because the use of them differs even though their representations are the same.

The format of the step lists is more complicated. A step list contains two sections. The first is a list of attribute-value pairs that contain state information such as the next step of the process flow. The other section is a sequence of entries that specify the process flow. Each entry consists of the name of a specification, the name of a step descriptor and, optionally, a list of attribute-value pairs that provide parameters for the specification or step descriptor. The names of the specification and step descriptor can be either generic, i.e. name the most recent version of an object, or they may specifically name a particular version of an object. The parameters will be used to provide specific information needed by specifications or step descriptors such as mask identifiers.

Specifications contain the majority of the text required to tell the technicians what needs to be done. They should specify what is to be done by a step as rigidly as possible, down to the level of describing what valves to turn, or buttons to push, where those valves and buttons are and how fast to turn or push them.

Step descriptors contain attribute-value pairs that must be filled in to provide parameters to the specification and to record the results of a step. The parameters section of the step descriptor is provided so that operations whose specifications are exactly the same except for minor parameter variations can use the same specification. This reduces the size of the specifications database and, more importantly, requires that only a single specification be changed when the method of performing the operation described by the specification changes. The results may contain data pertaining to a specific run as well as data needed to characterize and maintain the equipment.

Some step lists and step descriptors may be definitions. This means that they are not associated with any particular lot or maintenance activity. Step list definitions store the process flow for the standard processes.

Multiple step lists may point at any particular step descriptor. This allows a completed step descriptor to be recorded in multiple lot and equipment logs without redundantly recording the data. This will reduce the amount of disk space required to store the information and, more importantly, it makes changing the completed step descriptor a trivial task when a mistake is made filling it out. Step descriptors also contain pointers back to all the step lists which point to them. This makes questions like "what other lots were in the batch processed by this step" easy to answer.

6.0 Status Interface

The purpose of this interface is to allow people to quickly determine the state of the fabrication line, decide what they will do next, and, if what they decide to do requires the step interface or step list interface, to invoke that interface.

Invocation of the step interface from the status interface will bring up the next step of the step list associated with the lot or piece of equipment currently selected. Invocation of the step list interface will bring up the step list of the lot or piece of equipment currently selected and will select the next step of that step list.

Subsets of the status display are selectable by the operator by requiring that certain fields match criteria set by the operator. This allows an operator to focus attention on a specific set of lots and ignore the rest. Two subsets which are provided are the subset of lots assigned to a particular process technician and the subset of lots currently being processed or ready for processing at a specific processing area.

An audible alarm should be sounded when certain critical conditions are about to occur, e.g. a step requiring attention or failure of some part of the system.

The lot status could have the following parameters listed: lot name, next step, steps completed, steps to go, date/time step started, date/time step will need action, data/time step will finish, incremental time until step will finish, incremental time until lot will finish, date/time lot will finish, current technician, current process area, date/time step will need action, incremental time until step will need action.

It should be obvious from the status on the bulletin board what order runs were started in so that in the abscence of any specific priority lots will be processed on a first in first out basis.

It should be possible for a lot to have hold status. A *process hold* is one that someone in the fabrication area has put on a lot, e.g. a mask required to process the lot is missing. An *engineering hold* is one that someone outside of the fabrication area has put on a lot, e.g. a device designer has discovered an error in a design that requires a change to a masking layer that has not been used yet. While a lot has hold status no further processing should occur on it. Hold status will serve to flag the lot so that it is obvious that some action out of the ordinary must occur before the lot can continue through the fabrication process. It should be possible to easily insert a short description of why the lot is on hold. This description should be displayed with the lot status.

The *next step* of a lot refers to the last step that has not been started. It will be possible for an operator to select a particular lot and bring up the step interface for any step in the lot step list of that lot. This will be optimized so that the next step of a lot can be brought up quickly.

The equipment status could have the following parameters listed: equipment name, up or down, who is currently using the machine, incremental or date/time that machine will be free for use, incremental or date/time that current step on machine will finish, date/time that the next maintenance step has to be or should have been performed on this machine.

Some of the status suggested above reports when certain events in the future will occur. Reporting when a step will finish requires that the step descriptor or specification contain an estimate of how long the step takes. Reporting when a lot will finish requires that some person provide the estimate. Cholla simply records this estimate and displays it. Cholla does not calculate the estimate.

It will be possible to change the state of the bulletin board that all the above status is displayed upon using this interface.

7.0 Step List Interface

This interface allows manipulation of the step lists contained in the Cholla database. Its main purpose is to allow the process technicians to examine the step list during the processing of lots. It also allows people specified by an access list to edit step lists.

Invocation of the step interface from the step list interface is optimized to invoke the step interface with the current step of the step list. It is possible to invoke the step interface for any step contained in the lot step list.

There are a number of editing functions which should be optimized. One is taking a sequence of steps from within this step list or from any other step list and inserting them at some specific point. This is useful when a new step list is being composed or when a step reject has occured that requires editing of the step list. Another is copying a step list. This will be used each time a new lot is started.

8.0 Step Interface

This interface is responsible for displaying the specification and step descriptor and recording results in the step descriptor. The functions available in this inteface are start step, finish step, start automatic step execution, perform reject recovery, and reinitialize the step descriptor.

When the step interface is invoked a specific entry in a step list has been selected for it to use. The interface displays the specification named in the entry in a read only fashion. If the technician has not seen the specification being displayed before either because it is new or it has been updated then the technician should be warned of that fact. The step descriptor named in the step list entry is also displayed in a read only manner. Note that the parameters contained in the step list entry have not replaced the attributes they correspond to in the step descriptor yet.

When a step is started the step descriptor pointed at by the step list is copied, the attributes which have corresponding attributes specified in the step list entry have their values replaced by the values given in the step list, all step lists associated with the batch being processed are updated, the technician who started the step is recorded, the date and time the step was started is recorded, the entry on the bulletin board is changed, the equipment log is updated, and editing of the step descriptor is enabled. The next step is updated.

When a step is finished the time at which it was finished is recorded and the bulletin board is updated to reflect the new state of the lot and/or the equipment.

Invoking the automatic step execution function causes the step descriptor and specification to be sent to the equipment specific driver. When the automatic portion of the step is completed the equipment driver will return an updated step descriptor that has some of the fields filled in.

If the reject recover function is used then the action performed depends on whether the reject can be handled in a canned manner or if recovery requires manual intervention. If a canned description of the recovery action is available then the contents of the step list is recomputed and the next step set to the first step of the recovery action in the new step list. If manual intervention is required then the lot is put on process hold status until the step list can be rearranged using the step list interface.

Since the step descriptor can be edited in an arbitrary fashion it is possible to make enough editing mistakes that the best recovery action is to simply discard the current completed step descriptor and make a new copy of the original step descriptor. The ability to reinitialize the step descriptor is included for this reason.

The step descriptor may be filled in while the step is in progress but normally will be filled in when the step is completed. It will be possible to partially record results for a step, use the workstation for other purposes, and then return to record more results for that step with minimal operator interaction. When the operator indicates that a step is finished the system will check to ensure that all required information has been recorded and if it has not gives the operator the choice of recording the rest of the required information or forcing the step to be completed, in which case an exception is logged (generally for use by the process line manager). It will be possible for an operator to bring back a step descriptor that has already been finished, change it, and finish it again with the contents of the database updated appropriately so that only the information recorded in the last finished descriptor for each step remains. A syntax check of the step descriptor will be made before the database is updated.

It is possible for the device wafers to begin new steps of the processing sequence before the results of previous steps have been recorded. This may happen because the results of a step come from measuring test wafers which do not continue processing along with the device wafers in a lot. Thus it is possible for there to be more than one step started but not finished for any step list.

When several lots are batched together for a step then all of their step descriptors and parameters in their lot step lists must be exactly the same.

When entries are made in equipment logs that are in the form of 2D graphs the operator should be shown the graph with at least some of the old data and all of the new data. This will provide a point for continuous human monitoring about whether the variable is drifting with time or if it is just randomly shifting about within designated error limits.

If a lot has hold status a message indicating so should be displayed to the operator whenever a step from that lot is being displayed so that it is obvious that no further processing should occur even if the technician has not looked at the status display closely enough to see that the lot is on hold.

9.0 Implementation Sequence

All of Cholla will not be implemented immediately and some functions are more critical than others. This section provides an ordering for the sequence that Cholla should be implemented in. The ordering may change due to the fact that as functions are implemented and the system is brought up new requirements are discovered and the functional specification is altered. The intent is to construct the minimal system needed at each step of the implementation process so that user feedback may occur as soon as possible.

The tools needed to generate and maintain the specifications, step descriptions, and lot step lists in a cohesive database and to access this database while processing lots are the most critical needs. With these tools it will be possible to properly document the baseline process so that the process line can begin producing wafers in a repeatable manner. It is sufficient to have a single workstation contain these tools initially since the paper run cards can still be used in parallel during this stage.

When a baseline process has been established the number of lots in the line simultaneously will grow. At this point the status interface will be required in order to keep track of what lots require action.

The furnaces in the new diffusion area will require an extensive amount of development work in order to establish well documented, stable procedures for running them. Work on them should procede in parallel with the following work. The initial interface to them will be a low level driver that is not integrated with the rest of the Cholla software.

Eventually confidence in the automated system will be established and the paper lot step lists will be phased out. Cholla will have to work on multiple workstations at this point and the automated interfaces to the IBM film thickness analyzer, the ion implanter, and the Bruce furnaces should be integrated.

When paper has been eliminated from the processing line environment it will no longer be possible for an technician to write down results from a test, walk over to a workstation, and then enter the results that were written down. This has two effects. All equipment that does not have an automated interface for recording results must be sufficiently close to a workstation that the operator can look at the output of the machine while typing the results at the workstation. Secondly a technician must have access to a workstation upon demand. It will no longer be possible for a technician to batch their use of a workstation by proceding with processing while recording results on paper for later entry all at once using the workstation.

Elimination of the redundant recording of information in the various hardcopy log books will occur next, freeing the processing technicians from that paperwork to perform more productive duties. This will require that the equipment logs and the data display portions of the database management interface and the step execution interface be completed.

When all of the previously mentioned steps have occured it will be time to assess Cholla's performance and decide what further development should take place.

10.0 Future Directions

This section proposes facilities that will not be implemented in the first version of Cholla and to which there is no commitment to ever produce. They are presented here to provoke discussion about their relative utility and to provide a guideline for people in the computer world to what facilities are of interest to those in the processing world and to provide a similiar function to those in the processing world as to what is possible to do with computers.

Process line modeling to predict the effects of adding new equipment or running new processes with different equipment requirements.

Scheduling of equipment by assigning priorities to lots and knowing what the equipment time requirements are for each step of a process and what the current state of the line is in order to optimize line use in the face of differing turnaround requirements. A useful function would be to search the current list of pending actions for the processing line when a step is started to determine if starting this step now will cause a conflict later due to overcommiting the number of technicians available or some other limited resource.

Maintenance of supply inventories with automatic generation of orders when new supplies are needed. Provide predictions of when supplies will run out based on current usage rates and known future needs.

A method of automatically identifying wafers when they pass through specific machines could be introduced to provide tighter control over the process.

11.0 Bibliography

[1] Douglas K. Brotz and Charles M. Geschke, "Cholla: A Computer Assisted Silicon Processing Facility, Overview", April 1980

[2] Leo A. Doyal, Douglas L. Weaver, and Charles W. Gwyn, "Computer Control and Data Management in an LSI Fabrication Facility", *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, vol. CHMT-3, September 1980