Implementation Resources

Background

The system is to be implemented using a pair of microprocessors. One (the Equipment Control Processor) will perform the closed-loop control functions such as controlling vat level and pH, and will also monitor and operate the mechanical equipment such as the labeler and the bottle release mechanism. The other processor (the User Interface Processor) will accept and process keyboard input from the line operators and the area supervisor, and will produce screen displays for Line Status and Area Status.

There will be a set of interrupt lines between the processors; in other words, either processor will be able to cause an external interrupt to the other by issuing a command. In addition, there will be a standard data communications line between the processors cabable of transmitting messages. The "handshaking" relating to the D.C. line will be handled by system software. An application task in either processor need only set up a message buffer and request transmission. The completion of transmission will generate a software interrupt in the receiving processor, with the message stored in a buffer accessible to application tasks.

Each processor will be supplied with an operating system capable of managing an optional background task and a set of interrupt-activated tasks. There are three types of interrupts; external, software, and system clock. An interrupt of one of the first two types will cause control to be transferred to the application task associated with the interrupt. A system clock interrupt causes a lookup in a user-supplied table, which specifies task activations in terms of offset and frequency. For example, the table entries 1,5,A and 2,10,B specify that the first clock interrupt after startup and every fifth interrupt thereafter will activate A, and that the second clock interrupt after startup and every tenth thereafter will activate B.

The interrupt-activated tasks themselves may be interrupted, but not by the same type of interrupt that activated the task; in other words, the handling of an interrupt must be completed before another of the same type is honored. When the last outstanding interrupt-activated task reports completion, control is transferred to the background task, if there is one, until the next interrupt; otherwise, the operating system idles.

Transformation Schemas and Structure Charts

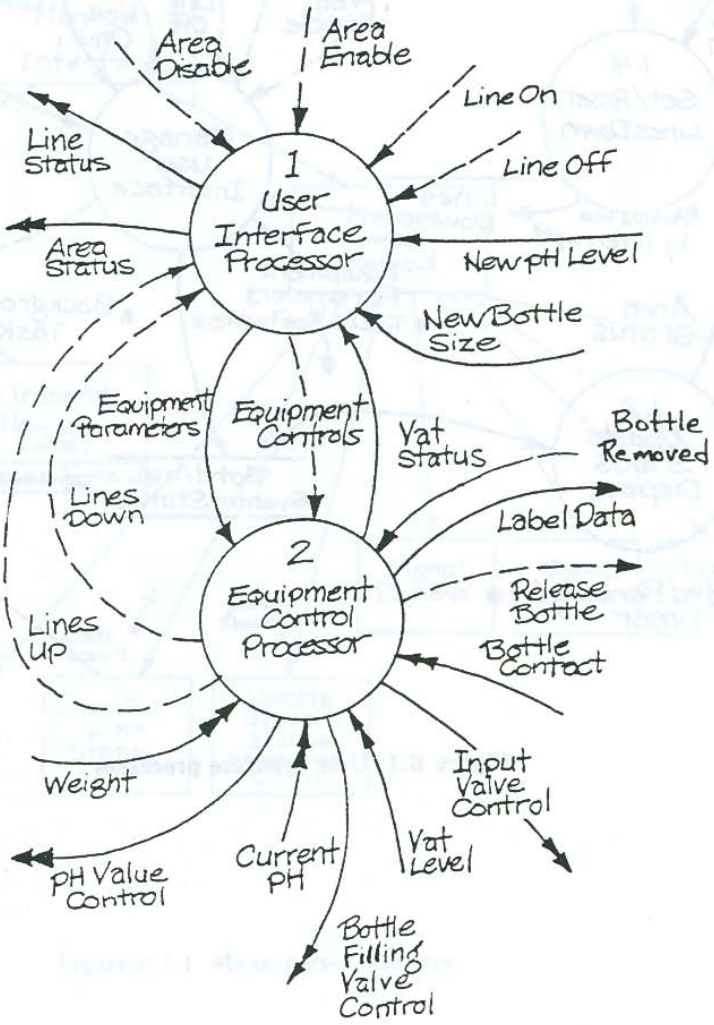

Figure B.0 Bottling system processors.

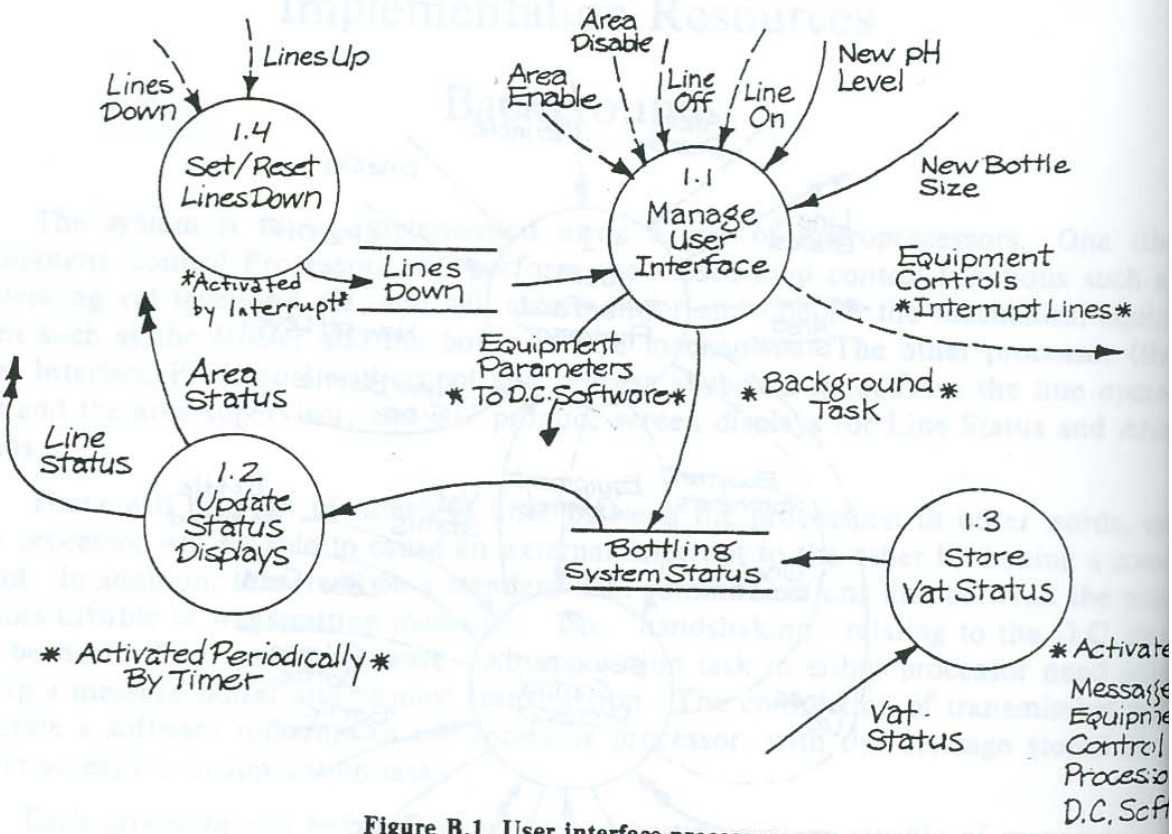

Figure B.1 User interface processor.

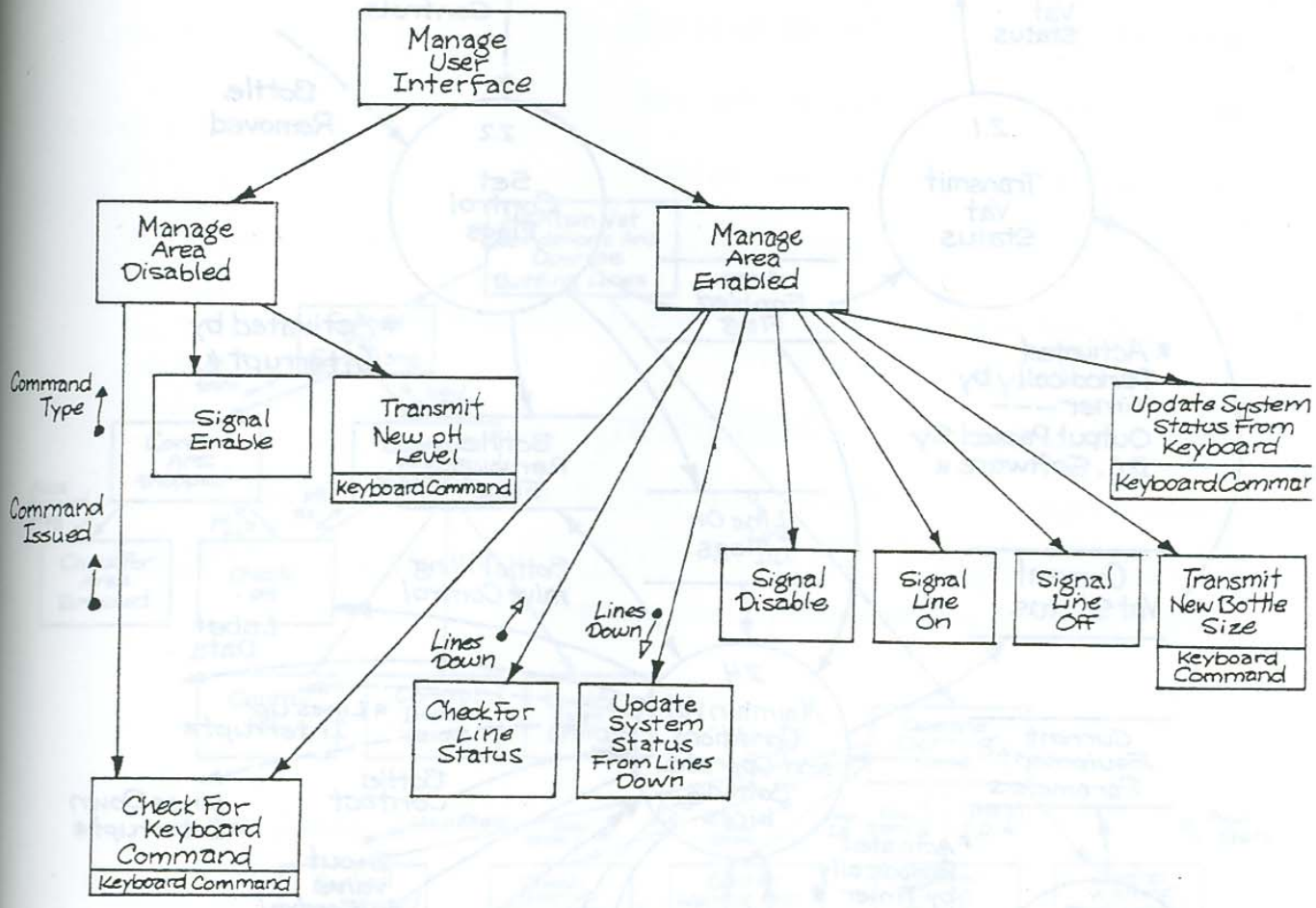

Figure B.1.1 Manage user interface.

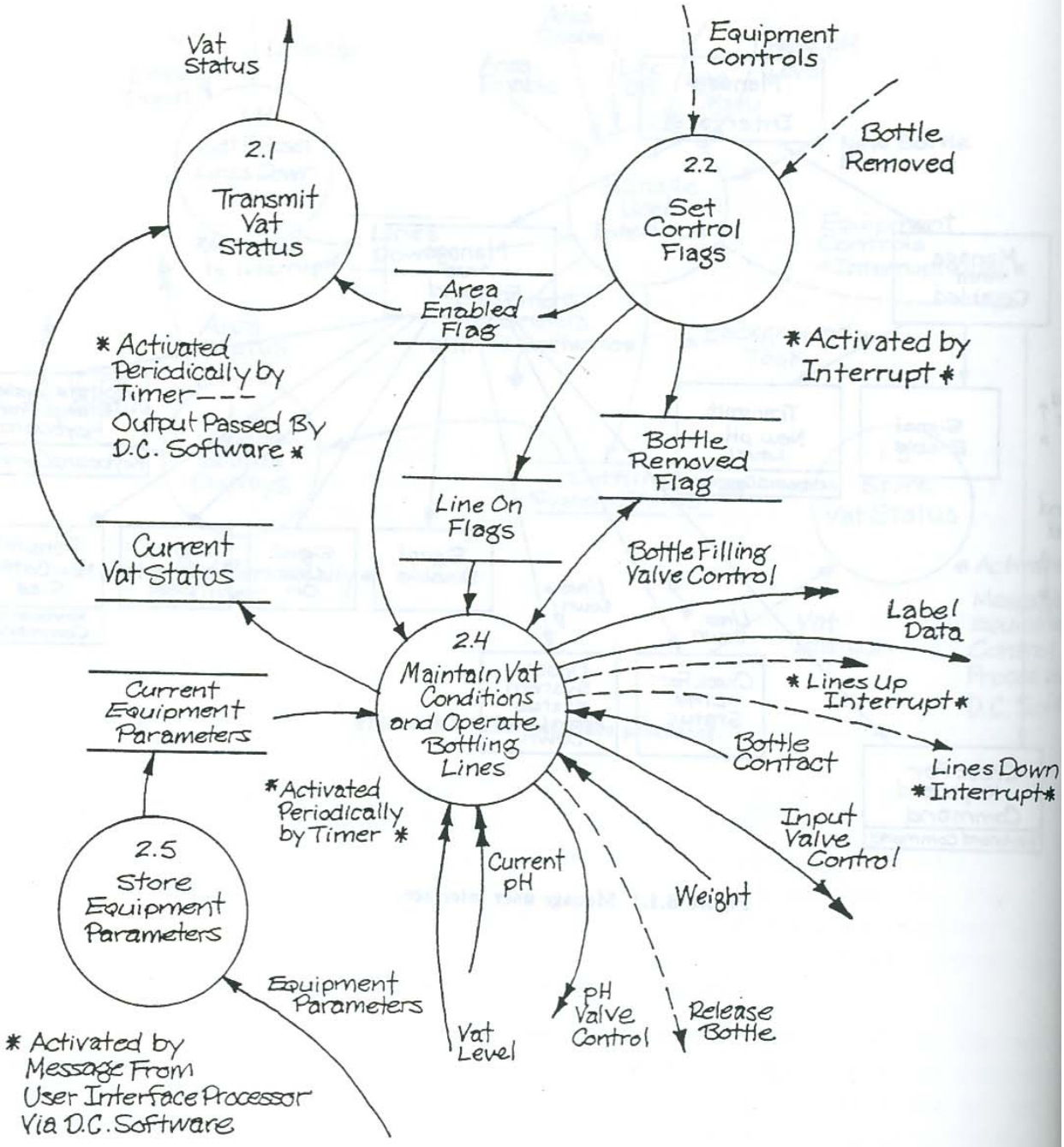

Figure B.2 Equipment control processor.

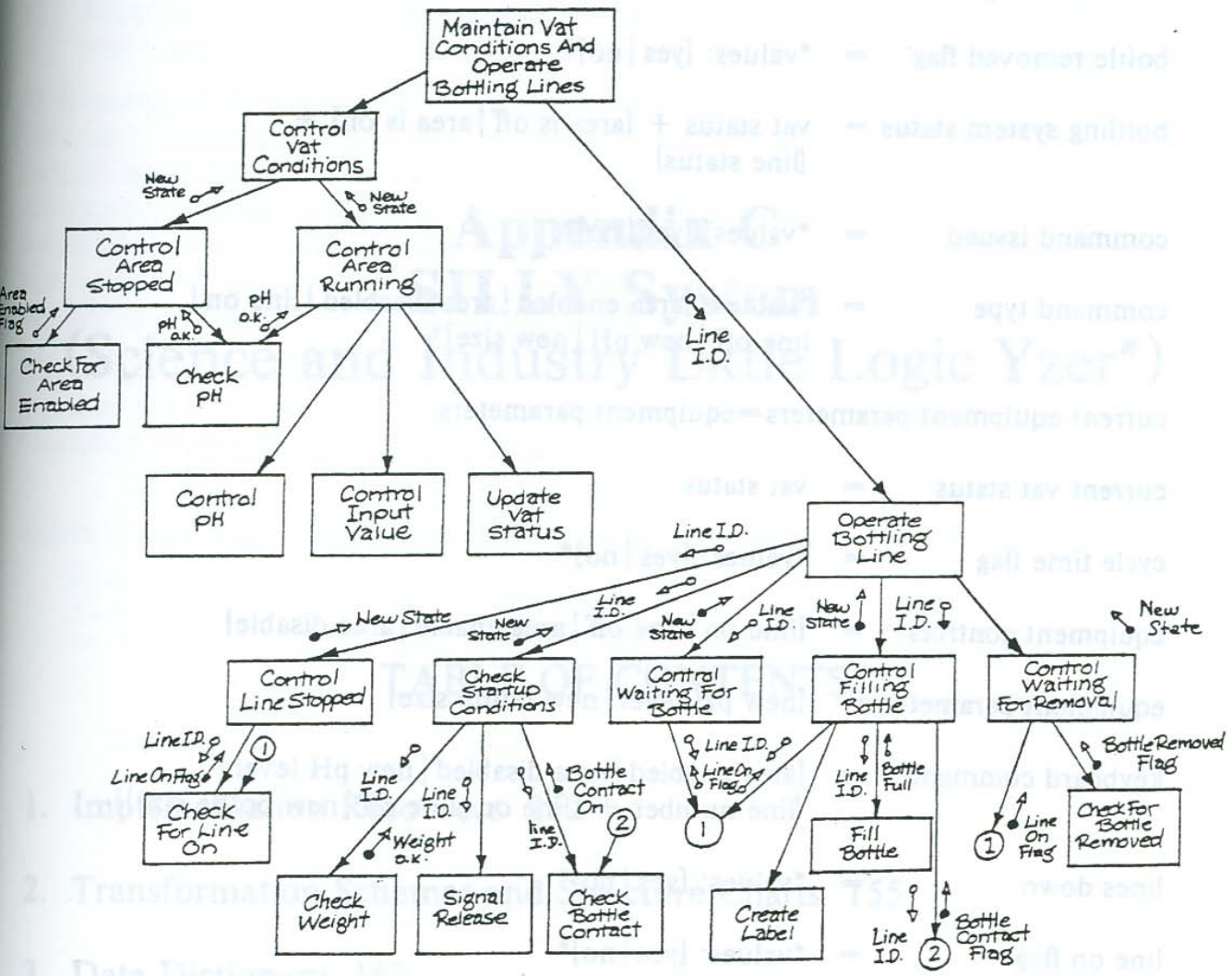

Figure B.2.4 Maintain vat conditions and operate bottling lines.