The Interlock Mailbox

The *Interlock Mailbox* is a facility designed to allow the operator interface panel to interact with the controller's ladder logic program. The Mailbox is simply a group of registers in the controller memory (from a minimum of 2 registers to a maximum of 20). The Mailbox can be placed anywhere in the controller memory at an address selected by the user. When the Mailbox is enabled, the operator panel cyclically reads the contents of the Mailbox from the controller. The controller can command the UniOP panel to perform certain tasks by placing data in the memory area used by the Mailbox. When the panel detects that the controller has written data in the Mailbox, it interprets the data as a numeric code for a required command. If the panel recognizes the command, it immediately executes it. If the command requires a response from the panel, then the panel places the response back into the Mailbox itself.

Definition of the Mailbox

To enable the Mailbox, select *Interlock Mailbox* from the *Project* menu. The dialog box shown in Figure 1 will then be displayed.

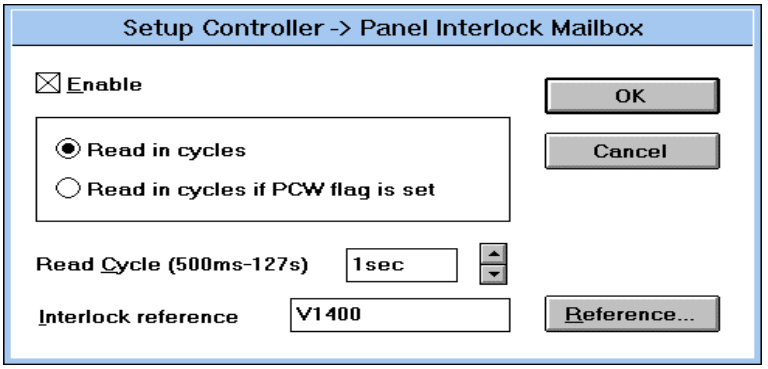

Figure 1

The Mailbox is disabled by default. To enable it, simply mark the check box entitled *Enable* within the dialog box. The two radio buttons allow the user to read the Mailbox in two different modes:

? Read in cycles
? Read in cycles if CCW (Controller Control Word) flag is set

In the latter case, the panel will cyclically read the data in the Mailbox only when bit C15 in the Controller Control Word of the Reserved Data Area is set. The purpose of this mode is to reduce the overhead of the communication time when there is no need for the controller to send commands to the panel.

The *Read Cycle* time specifies how frequently the operator panel will read the data in the Mailbox to check to see if it contains a new command. The faster the Mailbox *Read cycle*, the faster the panel will respond to commands from the controller. However, the update of the variables shown on the display will be slower.

The *Interlock reference* is the address of the first Mailbox register in the controller memory. The address can be entered directly as a string or with a fill-in-the-blank dialog box displayed by clicking on the *Reference* button. When entering the starting address, care must be taken to ensure that the controller memory used by the Mailbox does not overlap with the memory areas reserved for other panel functions such as alarms, I/O update, keyboard update, etc.

Structure of the Mailbox

The controller can issue different commands depending upon the values written to the Mailbox. Some commands require parameters (which have a binary numeric format) to be included with the command, while other commands require the panel to respond with data. Therefore, the amount of memory required for the Mailbox depends on the particular commands that are to be used. If the controller will only need to utilize simple Mailbox commands, then the amount of memory reserved for the Mailbox can be kept to a minimum. However, if more complicated commands are required, more memory will need to be reserved for the Mailbox. The structure of the Mailbox in the controller memory is shown below in Table 1. *The Status Word* and the *Command/Response Word* are always required for the Mailbox. Parameter 0 - Parameter 17 Words are only used if required by the particular Mailbox command.

Offset	
0	Status Word
1	Command/Response Word
2	Parameter 0
3	Parameter 1
...	
19	Parameter 17

Table 1

The user can determine the controller memory addresses of all the data elements in the Mailbox by using the starting address of the Mailbox area (set in the *Designer* software) and the memory map shown in Table 1.

Status Word

The *Status Word* is the main handshaking register of the Mailbox. Both the controller and the panel can write to this register. The status word contains a binary number, the meaning of which is described in Table 2.

Code	Status	Written by
0	No request, Mailbox is available	Controller
1	Pending request	Controller
2	Mailbox busy	Panel
3	Error code: Illegal command	Panel
4	Command completed	Panel
5	Panel restart	Panel

Table 2

Command/Response Word

The controller specifies which Mailbox command it wants to execute by placing the command code of the desired command in the Command/Response Word. The available Mailbox commands and their command codes are given in Table 3. The UniOP panel writes a binary value to the Command/Response Word when it has executed the command requested by the controller to indicate the result of the operation.

Parameters

The Parameter Words are used only when required by a specific command. Some commands do not require any parameters at all. The following section describes how to use and what Parameter Words are required for each Command Word.

Available Mailbox Commands

The Mailbox commands are listed in Table 3. The command codes must be written by the controller into the Command/Response Word register.

Command	Description
1	Recipe Status
2	Start Recipe Download
3	Start Recipe Upload
4	Set Parameter Set Number
5	Get Parameter Set Number
6	Get Recipe Transfer Priority
7	Set Recipe Transfer Priority
17 33	Request Data Entry Mode Password Status
49	Start Controller to Controller Transfer With No Confirmation
50	Start Controller to Controller Transfer With Confirmation
81	Set RTC
97	Clear Event List

Table 3

Recipe Status

The controller can use this command to request information on the status of the recipes in the panel.

Command code: 1

No parameters are required by this command. The panel will respond with a code in the Command/Response Word. The possible codes are listed in Table 4.

Code	Description
0	OK, Recipes contain valid data
1	Error: Recipes contain default data as defined in the project file
2	Error: There are no recipes defined
3	Error: Error in project file

Table 4

Start Recipe Download

The controller can use this command to start a recipe download from the panel memory to the controller memory.

Command code: 2

Two parameters are required by this command:

Parameter 0	Page Number of required Recipe (1 - max. page number in project file)
Parameter 1	Parameter set number (1 - max. number of pre-defined parameter sets)

The panel will respond back to the Command/Response Word with one of the codes listed in Table 5 after performing the operation. Note that a recipe transfer may take several seconds, depending mainly on the number of bytes that have to be transferred.

Code	Description
0	OK, Transfer completed successfully
1	Error: Recipes contain default data
2	Error: There are no recipes defined
3	Error: Error in project file
4	Error: Illegal Page number has been requested
5	Error: Illegal number of parameter set has been requested
6	Error: The requested parameter set is locked (that is, it is being used by the operator panel for a data entry operation)
7	Error: A communication error occurred during the transfer

Table 5

Start Recipe Upload

The controller can use this command to start a recipe upload from the controller memory to the panel memory.

Command code: 3

Two parameters are required by this command:

Parameter 0	Page Number of required Recipe (1 - max. page number in project file)
Parameter 1	Parameter set number (1 - max. number of pre-defined parameter sets)

After performing the operation, the panel will respond back to the Command/
Response Word with one of the codes listed in Table 6. Note that a recipe transfer may take several seconds, depending mainly on the number of bytes that have to be transferred.

Code	Description
0	OK, Transfer completed successfully
2	Error: There are no recipes defined
3	Error: Error in project file
4	Error: Illegal Page number has been requested
5	Error: Illegal number of parameter set has been requested
6	Error: The requested parameter set is locked (that is, it is being used by the operator panel for a data entry operation)
7	Error: A communication error occurred during the transfer

Table 6

Set Parameter Set Number

The controller can use this command to select a given recipe parameter set.

Command code: 4

Two parameters are required by this command:

Parameter 0	Page Number (1 - max. number of pages in the project file)
Parameter 1	Parameter Set Number (1 - max. number of pre-defined parameter sets)

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 7.

Code	Description
0	OK
2	Error: Recipes not defined in project file
3	Error: Error in project file
4	Error: Illegal page number requested
5	Error: Illegal record number requested
6	Error: Requested record locked

Table 7

Get Parameter Set Number

The controller can use this command to get the current recipe parameter set number.

Command code: 5

Only one parameter is required by this command:

Parameter 0	Page Number (1 - max. number of pages in the project file)

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 8.

Code	Description
0	OK
2	Error: Recipes not defined in project file
3	Error: Error in project file
4	Error: Illegal page number requested

Table 8

Get Recipe Transfer Priority

The controller can use this command to get the current recipe transfer priority level.

Command code: 6

This command requires no parameters.

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 9.

Code	Description
0	Normal
1	Highest
2	Error: Recipes not defined in project file

Table 9

Set Recipe Transfer Priority

The controller can use this command to set the current recipe transfer priority level.

Command code: 7

Only one parameter is required by this command:

Parameter 0 Priority Level (0=Normal, 1=Highest)

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 10.

Code	Description
0	OK
2	Error: Recipes not defined in project file
3	Error: Illegal priority setting requested

Table 10

Request Data Entry Mode

The controller can use this command to switch the operator panel to Data Entry Mode.

Command code: 17

Two parameters are required by this command:

Parameter 0 Index Number of required Page

	(1 - max. page number in project file)
Parameter 1	Index Number of required Data Field (1 - max. number of data fields in the page)

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 11.

Code	Description
0	OK
1	Error: Illegal page number has been requested
2	Error: Illegal field number has been requested
3	Error: Panel is not ready for data entry mode

Table 11

Password Status

The controller can use this command to request information about the current password status.

Command code: 33

The command requires no parameters.

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 12.

Code	Description
0	No passwords are defined in the project file
1	No valid password has been entered
2	A valid password has been entered
3	The system default password (0007) has been entered

Table 12

If Code 2 is returned, the Parameter Words will contain additional information as follows:

Parameter 0	Current password level
Parameter 1	First character (ASCII coded) of the current password
Parameter 2	Second character (ASCII coded) of the current password
Parameter 3	Third character (ASCII coded) of the current password
Parameter 4	Fourth character (ASCII coded) of the current password

Start Controller to Controller Transfer With No Confirmation

This feature allows the user to transfer data in two different manners:

1) Between two controllers. In this case, the controllers can be connected by a controller network or with the UniNET network.

2) Between two registers located within the same controller.

Note: Once more than one job has been created, it will be ***impossible*** *to entirely delete the first job without also deleting the second job. For example, assume there are three jobs, created in the order Job 1, Job 2, Job 3. If the user wishes to delete Job 2, they must also first delete Job 3. On the other hand,* ***modifications*** *can be made to any job at any time.*

We will call one complete data transfer a *job*. When the controller requests a job to be executed, the panel responds immediately back to the controller and executes the job as soon as possible. Therefore, the controller does not know when, or if, the requested job will be executed.

To set up a transfer, select *Data Transfer* from the *Project* menu. The dialog box shown in Figure 2 will then be displayed.

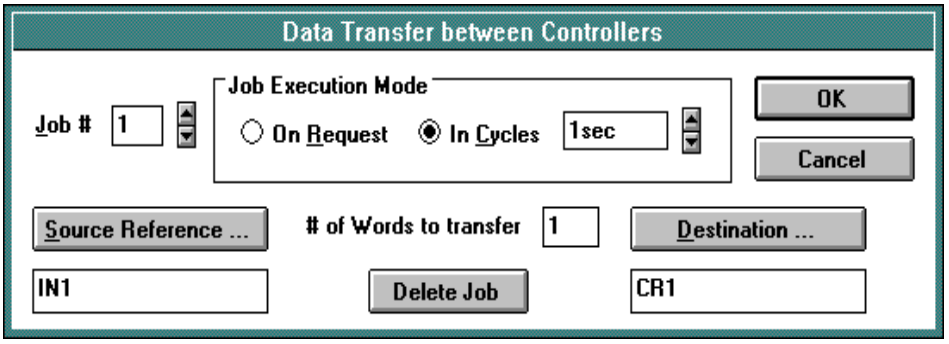

Figure 2

The *Job #* field is used to select between the different jobs. Click on the *Source Reference* button to specify the register to be read. The dialog box shown in Figure 3 will then be displayed.

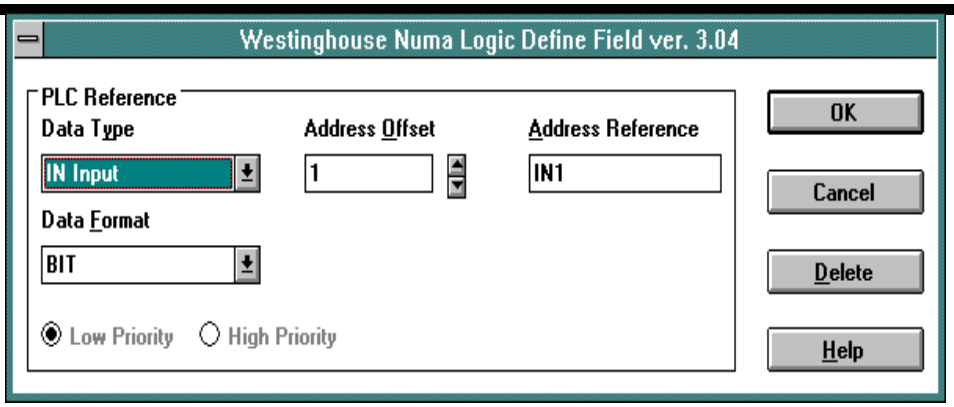

Figure 3

These fields will be slightly different depending on the controller that is being used. Use these fields to specify the type of data, format, and address of the register that is to be read. Once this information is entered, pressing OK will return the user to the dialog box shown in Figure 2. Enter the number of words to be transferred in the appropriate box. Clicking on the *Destination* button brings up a dialog box similar to the one shown above in Figure 3 and allows the user to enter the address in which the data is to be written.

If the *Job Execution Mode* is set to *In Cycles*, the panel will automatically transfer the data on each cycle. The cycle period can be set between 1 second and 127 seconds. Remember the faster the cycles, the faster the panel will transfer data, but this will slow down the rest of the system. If the *Job Execution Mode* is set to *On request*, the controller must send a request to the Interlock Mailbox before the data will be transferred. After completely setting up the dialog box, click on *OK*. Now select *Interlock Mailbox* from the *Project* menu. Set up the Mailbox as described within the section entitled *Definition of a Mailbox*. When the controller sends a request to the Mailbox, the Mailbox will read the command word and the parameter. The panel will then send a response word back to the controller and execute the command.

Command code: 49

This command requires only one parameter:

Parameter 0 Job Number (1-255)

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 13.

Code	Description
0	OK
1	Error: Controller to Controller not defined at all
2	Error: Illegal job number
3	Error: Queue is full

Table 13

Start Controller to Controller Transfer With Confirmation

This command is set up in the same manner as the command *Start Controller to Controller Transfer With No Confirmation*.

Command code: 50

This command requires only one parameter:

Parameter 0 Job Number (1-255)

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 14.

Code	Description
0	OK
1	Error: Controller to Controller not defined at all
2	Error: Illegal job number
3	Error: Not used
4	Error: Communication error while reading source data
5	Error: Communication error while writing destination data

Table 14

Set RTC (Real Time Clock)

This command enables the controller to set the Real Time Clock located within the panel.

Command code: 81

This command requires six parameters:

Parameter 0 Day (1-31)
Parameter 1 Month (1-12)
Parameter 2 Year (94...99, 00....)
Parameter 3 Hour (0-23)
Parameter 4 Minute (0-59)
Parameter 5 Second (0-59)

After performing the operation, the panel will respond back to the Command/Response Word with one of the codes listed in Table 15.

Code	Description
0	OK
1	Error: RTC error
2	Error: Requested setting out of range

Table 15

Clear Event List From Controller Mailbox

The controller can use this command to request that the UniOP panel clear its event list.

Command code: 97

The command requires no parameters.
After performing the operation, the panel will respond back to the Command/Response Word with a 0, indicating that the event list has been cleared.

8.3 - Mailbox Programming

To use the mailbox successfully, the panel and the controller must interact tightly. The interaction between the panel and the controller can be thought of as a short conversation between two people.

The first person (the controller) tries to start a conversation with the second person (the UniOP panel) on a particular topic (the Mailbox command code). After the first person has introduced himself and the topic of the conversation, the second person decides whether he feels competent enough to join in. If the second person decides not to join in the conversation, he politely informs the first person that he feels he does not know enough about that particular topic to make an informed comment and waits for a change in the topic of the conversation. If the second person does decide to enter into the conversation, he ponders the topic and then expresses his well considered opinion. The vital ingredient in a successful conversation is that nobody is interrupted while they are talking.

The flow chart in Figure 4 describes the program sequences that must be executed in parallel by both the UniOP panel and the controller ladder logic program to ensure a successful Mailbox session.

The following important notes should be kept in mind:

? At system start-up, the UniOP panel writes the special Status Code *Panel Restart* to inform the controller that the panel itself has been reset.

? Both the panel and the controller can write to the Status Word and the Command/Response Word; however, they must not write at the same time to the same memory location. Proper operation of the Mailbox is ensured if the sequence described in the figure is followed.

? The controller program must always wait for the completion of the current Mailbox session before attempting to start a new one.

? The controller program should always check the status returned by the UniOP panel on completion of a session.

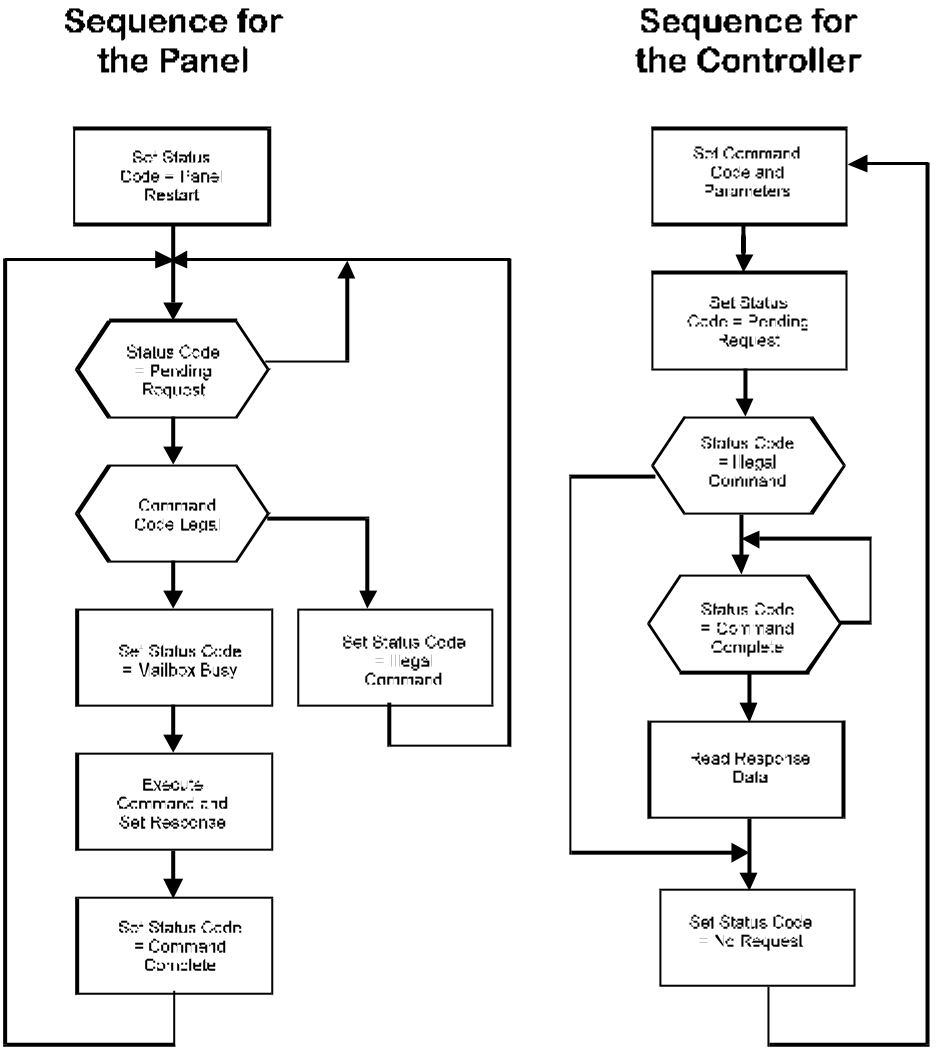

Figure 4

Technical assistance may be obtained :
In North America EXOR at 1-513-874-4665
In Europe EXOR Italy at +39 045 - 8779068