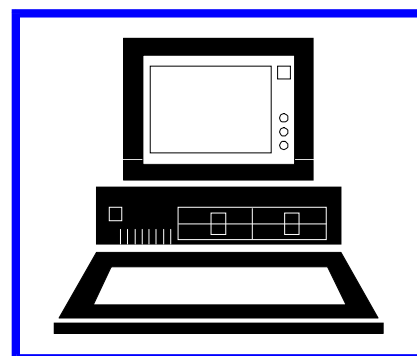
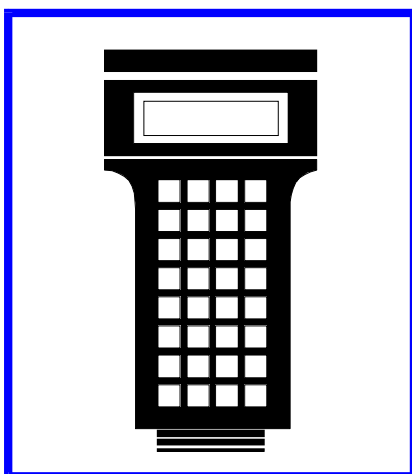
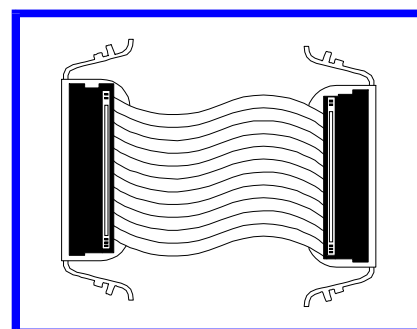
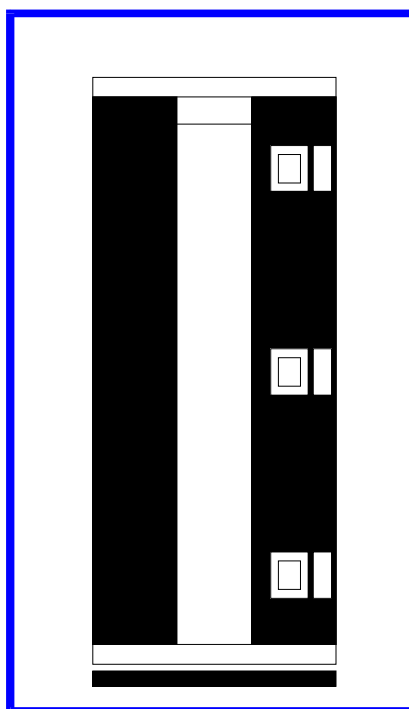
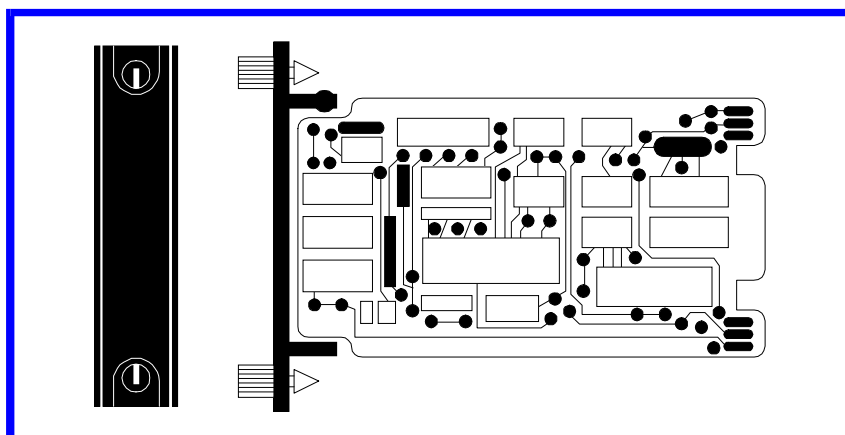
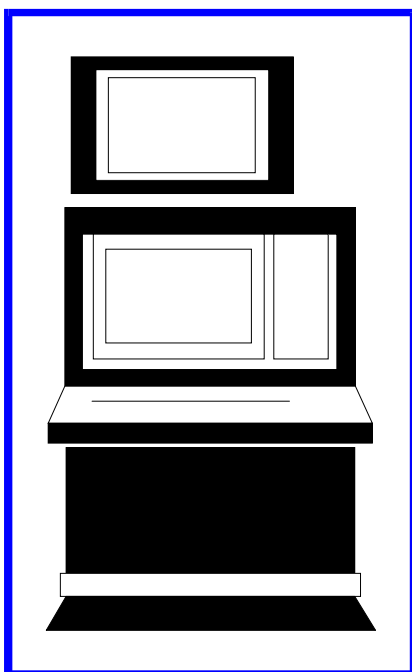


WBDM

Bailey®
infi 90

Instruction

Batch Data Manager (Software Release 4.0)



WARNING notices as used in this instruction apply to hazards or unsafe practices that could result in personal injury or death.

CAUTION notices apply to hazards or unsafe practices that could result in property damage.

NOTES highlight procedures and contain information that assists the operator in understanding the information contained in this instruction.

WARNING

INSTRUCTION MANUALS

DO NOT INSTALL, MAINTAIN, OR OPERATE THIS EQUIPMENT WITHOUT READING, UNDERSTANDING, AND FOLLOWING THE PROPER **Elsag Bailey** INSTRUCTIONS AND MANUALS; OTHERWISE, INJURY OR DAMAGE MAY RESULT.

RADIO FREQUENCY INTERFERENCE

MOST ELECTRONIC EQUIPMENT IS INFLUENCED BY RADIO FREQUENCY INTERFERENCE (RFI). CAUTION SHOULD BE EXERCISED WITH REGARD TO THE USE OF PORTABLE COMMUNICATIONS EQUIPMENT IN THE AREA AROUND SUCH EQUIPMENT. PRUDENT PRACTICE DICTATES THAT SIGNS SHOULD BE POSTED IN THE VICINITY OF THE EQUIPMENT CAUTIONING AGAINST THE USE OF PORTABLE COMMUNICATIONS EQUIPMENT.

POSSIBLE PROCESS UPSETS

MAINTENANCE MUST BE PERFORMED ONLY BY QUALIFIED PERSONNEL AND ONLY AFTER SECURING EQUIPMENT CONTROLLED BY THIS PRODUCT. ADJUSTING OR REMOVING THIS PRODUCT WHILE IT IS IN THE SYSTEM MAY UPSET THE PROCESS BEING CONTROLLED. SOME PROCESS UPSETS MAY CAUSE INJURY OR DAMAGE.

AVERTISSEMENT

MANUELS D'OPÉRATION

NE PAS METTRE EN PLACE, RÉPARER OU FAIRE FONCTIONNER L'ÉQUIPEMENT SANS AVOIR LU, COMPRIS ET SUIVI LES INSTRUCTIONS RÉGLEMENTAIRES DE **Elsag Bailey**. TOUTE NÉGLIGENCE À CET ÉGARD POURRAIT ÊTRE UNE CAUSE D'ACCIDENT OU DE DÉFAILLANCE DU MATÉRIEL.

PERTURBATIONS PAR FRÉQUENCE RADIO

LA PLUPART DES ÉQUIPEMENTS ÉLECTRONIQUES SONT SENSIBLES AUX PERTURBATIONS PAR FRÉQUENCE RADIO. DES PRÉCAUTIONS DEVRONT ÊTRE PRISES LORS DE L'UTILISATION DU MATÉRIEL DE COMMUNICATION PORTATIF. LA PRUDENCE EXIGE QUE LES PRÉCAUTIONS À PRENDRE DANS CE CAS SOIENT SIGNALÉES AUX ENDROITS VOULUS DANS VOTRE USINE.

PERTURBATIONS DU PROCÉDÉ

L'ENTRETIEN DOIT ÊTRE ASSURÉ PAR UNE PERSONNE QUALIFIÉE EN CONSIDÉRANT L'ASPECT SÉCURITAIRE DES ÉQUIPEMENTS CONTRÔLÉS PAR CE PRODUIT. L'AJUSTEMENT ET/OU L'EXTRACTION DE CE PRODUIT PEUT OCCASIONNER DES À-COUPS AU PROCÉDÉ CONTRÔLE LORSQU'IL EST INSÉRÉ DANS UNE SYSTÈME ACTIF. CES À-COUPS PEUVENT ÉGALEMENT OCCASIONNER DES BLESSURES OU DES DOMMAGES MATÉRIELS.

NOTICE

The information contained in this document is subject to change without notice.

Elsag Bailey, its affiliates, employees, and agents, and the authors and contributors to this publication specifically disclaim all liabilities and warranties, express and implied (including warranties of merchantability and fitness for a particular purpose), for the accuracy, currency, completeness, and/or reliability of the information contained herein and/or for the fitness for any particular use and/or for the performance of any material and/or equipment selected in whole or part with the user of/or in reliance upon information contained herein. Selection of materials and/or equipment is at the sole risk of the user of this publication.

This document contains proprietary information of Elsag Bailey, Elsag Bailey Process Automation, and is issued in strict confidence. Its use, or reproduction for use, for the reverse engineering, development or manufacture of hardware or software described herein is prohibited. No part of this document may be photocopied or reproduced without the prior written consent of Elsag Bailey.

Preface

The Batch Data Manager (BDM) interface provides data management and application integration functionality for all the INFI-BATCH™ tools. Batch 90™ and User Defined Function Code (UDF), Dynamic Debugger (DBG), Unit Procedure Editor (UPE), and Master Recipe Editor (MRE) are some of the tools. All batch tools are designed for the Microsoft® Windows® 95 operating system.

™ Trademark of Elsag Bailey Process Automation.
® Registered trademark of Microsoft Corporation.

List of Effective Pages

Total number of pages in this instruction is 144, consisting of the following:

Page No.	Change Date
Preface	Original
List of Effective Pages	Original
iii through viii	Original
1-1 through 1-4	Original
2-1 through 2-11	Original
3-1 through 3-5	Original
4-1 through 4-9	Original
5-1 through 5-8	Original
6-1 through 6-13	Original
7-1 through 7-20	Original
8-1 through 8-30	Original
9-1 through 9-12	Original
10-1 through 10-9	Original
11-1 through 11-5	Original
12-1 through 12-6	Original
Index-1 through Index-4	Original

When an update is received, insert the latest changed pages and dispose of the superseded pages.

NOTE: On an update page, the changed text or table is indicated by a vertical bar in the outer margin of the page adjacent to the changed area. A changed figure is indicated by a vertical bar in the outer margin next to the figure caption. The date the update was prepared will appear beside the page number.

Table of Contents

	<i>Page</i>
SECTION 1 - INTRODUCTION	1-1
INTRODUCTION	1-1
INTENDED USER	1-1
HARDWARE REQUIREMENTS	1-1
SOFTWARE REQUIREMENTS	1-2
INSTRUCTION CONTENT	1-2
HOW TO USE THIS INSTRUCTION	1-2
DOCUMENT CONVENTIONS	1-3
GLOSSARY OF TERMS AND ABBREVIATIONS	1-4
REFERENCE DOCUMENTS	1-4
SECTION 2 - DESCRIPTION AND OPERATION	2-1
BATCH PROCESS	2-1
BATCH RECIPES	2-1
BATCH DATA MANAGER	2-2
Batch Projects	2-2
Create Project	2-5
Add Classes	2-5
Create Class Programs	2-6
Add Unit Definition Files	2-6
Develop Recipes	2-6
Download Programs, Files, And Recipes	2-7
Execute Recipes	2-7
Debug Recipes	2-7
UDF Projects	2-7
Create Project	2-7
Create Program	2-8
Download Program	2-8
Execute Program	2-8
Debug Program	2-8
BATCH 90 SUPPORT	2-8
Unit Data Declarations	2-8
Unit Parameter Declaration	2-9
UDF PROGRAMS	2-10
ONLINE PROGRAM CHANGES	2-10
SECTION 3 - INSTALLATION	3-1
INTRODUCTION	3-1
INSTALLATION	3-1
START-UP	3-1
DIRECTORY STRUCTURE	3-2
SECTION 4 - PROJECTS	4-1
INTRODUCTION	4-1
ADD	4-1
CLOSE	4-2
OPEN	4-2
COPY	4-3
DELETE	4-3
RENAME	4-4
EDIT USERS LIST	4-4
Add User	4-5

Table of Contents (continued)

	<i>Page</i>
<hr/>	
SECTION 4 - PROJECTS (continued)	
Delete All Users	4-5
Refresh.....	4-5
Delete.....	4-6
Modify	4-6
Reset Password	4-6
PROPERTIES	4-7
Drive	4-7
Project.....	4-7
Users List	4-8
REFRESH.....	4-8
SECURITY	4-8
<hr/>	
SECTION 5 - CLASS LIBRARIES AND CLASSES	5-1
INTRODUCTION	5-1
CLASS LIBRARIES	5-1
Delete.....	5-1
Remove Reference.....	5-1
Add	5-2
Rename	5-2
Copy.....	5-3
CLASSES.....	5-3
Add	5-3
Copy.....	5-5
Rename.....	5-6
Delete.....	5-6
Change Class Program	5-6
PROPERTIES	5-7
Class Library	5-7
Class.....	5-7
<hr/>	
SECTION 6 - BATCH 90 AND UDF PROGRAMS	6-1
INTRODUCTION	6-1
BATCH 90 PROGRAMS	6-1
Edit.....	6-1
Compile.....	6-1
Print.....	6-3
Source File	6-3
UDF PROGRAMS	6-4
Add	6-4
Edit.....	6-4
Copy.....	6-5
Compile.....	6-6
Delete.....	6-7
Rename.....	6-7
Print.....	6-8
Source File	6-8
UNITS.....	6-8
Add	6-9
Compile.....	6-9
Print.....	6-11
Delete.....	6-11

Table of Contents (continued)

	<i>Page</i>
<hr/>	
SECTION 6 - BATCH 90 AND UDF PROGRAMS (continued)	
Properties	6-11
REBUILD	6-12
<hr/>	
SECTION 7 - UNIT PROCEDURES AND UNIT RECIPES.....	7-1
INTRODUCTION.....	7-1
UNIT PROCEDURES	7-1
Add.....	7-2
Edit	7-2
Copy	7-9
Rename	7-10
Print	7-10
Delete	7-11
UNIT RECIPE	7-12
Add.....	7-12
Edit	7-13
Copy	7-14
Rename	7-15
Delete	7-15
PROPERTIES	7-16
Unit Procedure.....	7-16
Operation	7-18
Phase.....	7-18
Unit Procedure Editor	7-19
Unit Recipes	7-19
<hr/>	
SECTION 8 - MASTER RECIPES AND CONTROL RECIPES	8-1
INTRODUCTION.....	8-1
MASTER RECIPES	8-1
Add.....	8-1
Edit	8-2
Copy	8-16
Rename	8-17
Print	8-18
Delete	8-20
CONTROL RECIPES	8-20
Add.....	8-20
Edit	8-22
Copy	8-23
Rename	8-23
Delete	8-24
PROPERTIES	8-25
Master Recipe	8-25
Segment	8-26
Unit Procedure.....	8-27
Operation	8-28
Phase.....	8-28
Master Recipe Editor	8-28
Control Recipe	8-29
Train.....	8-30

Table of Contents (continued)

	<i>Page</i>
SECTION 9 - INFORMATION DOWNLOAD	9-1
INTRODUCTION	9-1
FORMAT.....	9-1
DOWNLOAD	9-2
Batch 90 Program Files	9-2
UDF Program Files	9-2
Units.....	9-3
Unit Recipes	9-3
Control Recipes	9-4
VERIFY.....	9-4
Batch 90 Program Files	9-4
UDF Program Files	9-4
Units.....	9-5
Unit Recipes	9-5
Control Recipes	9-5
DELETE	9-5
Batch 90 Program Files	9-6
UDF Program Files	9-6
Units.....	9-6
Unit Recipes	9-6
Control Recipes	9-7
DOWNLOAD UTILITY.....	9-7
Change Module Mode	9-8
Read Module Directories.....	9-8
Search For Modules.....	9-9
Communication Settings	9-9
Size	9-10
Delete.....	9-10
Print.....	9-11
Properties.....	9-11
SECTION 10 - DYNAMIC DEBUGGER UTILITY.....	10-1
INTRODUCTION	10-1
DEBUGGER CAPABILITIES	10-3
Stop Execution	10-3
Start Execution	10-3
Step Execution	10-4
Go To Specific Line	10-4
View and Watch Variables	10-5
Change Operation	10-6
Change Phase Debugged	10-7
Breakpoints.....	10-7
Remove Variables From Watch	10-8
Find Specific Text	10-8
Escape Loop	10-8
Properties.....	10-8
SECTION 11 - ARCHIVE LOG	11-1
INTRODUCTION	11-1
VIEW	11-2
PAUSE.....	11-2
RESUME	11-2

Table of Contents (continued)

	<i>Page</i>
<hr/>	
SECTION 11 - ARCHIVE LOG (continued)	
SORT	11-2
PRINT	11-2
FILTERS	11-3
Display Filter	11-3
Write Filter	11-3
LOG SIZE.....	11-4
DELETE.....	11-4
PROPERTIES	11-5
<hr/>	
SECTION 12 - ERROR MESSAGES AND RECOVERY	12-1
COMPILER ERROR CODES.....	12-1
RUNTIME FAULT CODES.....	12-4
<hr/>	

List of Figures

<i>No.</i>	<i>Title</i>	<i>Page</i>
<hr/>		
2-1.	Composition of a Recipe	2-2
2-2.	Unit Procedure Application Creation	2-3
2-3.	Master Recipe Application Creation	2-4
2-4.	UDF Program Application Creation.....	2-5
3-1.	Batch Data Manager Window Example	3-2
3-2.	Batch Data Manager Directory Structure	3-2
4-1.	New Batch Project Windows Example	4-2
4-2.	New UDF Project Window Example.....	4-3
4-3.	Users List Window Example	4-4
5-1.	Class Library Window Example Containing Classes	5-4
6-1.	Batch 90 Editor Window Example	6-2
6-2.	UDF Editor Window Example	6-5
6-3.	Class Library Window Example Containing Units	6-9
7-1.	Unit Procedures Window Example	7-1
7-2.	Unit Procedure Editor Window (Unedited) Example	7-3
7-3.	Unit Procedure Editor Window (Edited) Example	7-4
8-1.	Master Recipes Window Example	8-2
8-2.	Master Recipe Editor Window (Unedited) Example	8-3
8-3.	Master Recipe Editor Window (Edited) Example.....	8-4
9-1.	Batch Downloader Utility Window Example	9-7
10-1.	Batch Debugger Window Example	10-1
11-1.	Batch Archive Log Window Example	11-1
<hr/>		

List of Tables

<i>No.</i>	<i>Title</i>	<i>Page</i>
1-1.	Glossary of Terms and Abbreviations	1-4
1-2.	Reference Documents	1-4
3-1.	Batch Data Manager Files.....	3-3
12-1.	Compiler Error Messages.....	12-1
12-2.	Runtime Fault Codes.....	12-4

SECTION 1 - INTRODUCTION

INTRODUCTION

The Batch Data Manager (BDM) interface provides an interactive tool for creating, managing, and utilizing Batch 90 and User Defined Function (UDF) data. The batch data manager, which functions in the Windows 95 operating system, is the major component of the WB9040 (Batch 90 programming language from Windows), WUDF40 (UDF programming language for Windows), and WMRE40 (Master Recipe Editor for Windows) software packages.

All data accessed by the batch data manager is organized into projects and then into unit procedures, master recipes (if the master recipe editor option is installed), or UDF programs. Projects contain either Batch 90 or UDF data. The batch data manager handles all aspects of project creation, downloading, and debugging.

INTENDED USER

This instruction provides the necessary information to install the WB9040, WUDF40, and WMRE40 software packages. It also contains information about how to operate the batch data manager interface. It is intended for programmers with batch process or sequential logic control experience, working knowledge of function codes and the Elsas Bailey CAD/EWS software. The programmer should also be familiar with the configuration of operator interface stations.

HARDWARE REQUIREMENTS

The WB9040, WUDF40, and WMRE40 software packages are certified to function on the following computer systems:

- DEC™ Celebris 5-90 (90 MHz Pentium®).
- Gateway 2000™ P4D-66 (66 MHz 80486).

Other computer systems not certified must meet the following minimum configuration:

- Intel® 486 - 66 megahertz central processing unit.
- 16 megabytes of RAM memory.
- 200 megabytes of available hard disk space.
- Super VGA graphics capability.
- Mouse.

[™] DEC is a trademark of Digital Equipment Corporation.

[®] Pentium and Intel are registered trademarks of Intel Corporation.

[™] Gateway 2000 is a trademark of Gateway 2000.

- 3.5 inch floppy disk drive.

SOFTWARE REQUIREMENTS

The only software requirement is that the Microsoft Windows 95 operating system be installed and operating properly.

INSTRUCTION CONTENT

This instruction contains 12 sections. It also includes a Table of Contents, List of Figures, List of Tables, and Index giving several options to locate specific information quickly. The sections that make up this instruction include:

Introduction	Provides an overview of the batch data manager and this instruction.
Description and Operation	Describes the functions, capabilities, and usage of the batch data manager.
Installation	Details how to install the WB9040, WUDF40, and WMRE40 software packages.
Projects	Describes what projects are and how they are created.
Class Libraries and Classes	Explains what class libraries and classes are and how they are created.
Batch 90 and UDF Programs	Provides an overview of Batch 90 and UDF programs and how to create or edit them using the batch data manager. Also described are units and how they are created.
Unit Procedures and Unit Recipes	Details what unit procedures and unit recipes are and how they are created.
Master Recipe and Control Recipes	Describes what master recipes and control recipes are and how they are created.
Information Download	Details how to download Batch 90 and UDF project contents. Also explains the batch download utility and its functions.
Dynamic Debugger Utility	Explains how to use this utility to debug Batch 90 and UDF programs.
Archive Log	Describes how to set up and use the archive log.
Error Messages and Recovery	Provides a listing of the possible error messages and corresponding recovery procedures.

HOW TO USE THIS INSTRUCTION

Read this instruction through in sequence before attempting to use the batch data manager. It is important to become

familiar with the entire contents of the instruction prior to creating, managing, and downloading any Batch 90 or UDF projects.

This instruction limits the information presented in each section to only specific items required to complete the desired task. The organization enables finding specific information quickly, and permits using this instruction as a reference after becoming fully familiar with the batch data manager interface.

Be sure to read notes in text. Notes provide:

- Additional information.
- Information that should be considered before using a certain command or statement.

DOCUMENT CONVENTIONS

This document uses standard text conventions throughout to represent keys, display items, and file names:

KEY Identifies a keyboard key.

Example: Press **ENTER**.

Display item Any item that displays on the screen appears as italic text in this document.

Examples: *Add B90 Project...* (menu selection)

Invalid Recipe ID (message)

Select function (prompt)

File name Any file names and file extensions appear as bold-italic text.

Example: **SETUP.EXE**

The document uses a specific set of text conventions for user inputs:

BOLD Identifies any user input or part of a command line that is **not** optional or variable, and must be entered exactly as shown.

italic Identifies a variable parameter in user input or a command line.

[] The brackets indicate a parameter is optional. Text within the brackets still follows the previously described conventions.

Example: **VAR STRING** *vstrgname* [(const)]

GLOSSARY OF TERMS AND ABBREVIATIONS

Table 1-1 is a glossary of terms and abbreviations used in this instruction. It contains those terms and abbreviations that are unique to Elsag Bailey or have a definition that is different from standard industry usage.

Table 1-1. Glossary of Terms and Abbreviations

Term	Definition
CAD	Computer aided design. A Bailey computerized drawing package that allows an engineer/technician to create function block control logic.
Control Module	Directs field processes through an I/O module; the multi-function processor is an example.
INFI-NET®	Advanced data communication highway.
MFP	Multi-function processor module. A multiple-loop controller with data acquisition and information processing capabilities.
OIS	Operator interface station. Integrated operator console with data acquisition and reporting capabilities. It provides a digital access into the process for flexible control and monitoring.
Plant Loop	Network 90® data communication highway.

REFERENCE DOCUMENTS

This instruction provides information about the batch data manager and its options. Table 1-2 lists additional documents that relate to the software and hardware.

Table 1-2. Reference Documents

Number	Document
I-E96-200	Function Code Application Manual
I-E96-201	Multi-Function Processor Module (IMMFP01)
I-E96-202	Multi-Function Processor Module (IMMFP02)
I-E96-203	Multi-Function Processor Module (IMMFP03)
I-E96-717	Personal Computer Software Computer Aided Drawing/Text (Release 5.4)
WBPEEUI370252A0	Batch 90 and User Defined Function (UDF) Programming language Reference Manual (Release 4.0)

SECTION 2 - DESCRIPTION AND OPERATION

BATCH PROCESS

Manufacturing processes are generally classified as continuous or batch. The process classification depends on whether the output from the process appears in a continuous flow or in discrete quantities.

In a continuous process, a product is made by passing material through different pieces of equipment. Each of these pieces of equipment operates in a steady state and performs one dedicated processing function. The output from the continuous process is a continuous flow.

In a batch process, a product is made by processing measured quantities of raw materials in a sequential order using one or more pieces of equipment. Each piece of equipment performs a specific processing function on a finite quantity of material. The product generated by a batch process is referred to as a batch.

BATCH RECIPES

Every batch production facility uses a recipe. The recipe is a complete set of information that specifies the control requirements for making a batch of product. The recipe is a combination of procedure, formula, header, and equipment requirements. Procedures define the order in which actions are performed. Formulas are sets of parameters that define information such as quantities of ingredients, duration, and magnitude of process variables. Headers provide information about the purpose and version of the recipe, such as product identification and description. Equipment requirements specify the type and number of pieces of equipment needed to produce a batch. Figure 2-1 represents the composition of a recipe.

The unit procedure or master recipe is a generic form of the recipe. In this form, the recipe provides the header, procedure, formula information, and defines the class of equipment used. It does not specify a particular unit within a class. The unit definition or train includes the additional information required to make the recipe equipment specific. In this form, the recipe contains the equipment specific information necessary to complete the making of a batch of product.

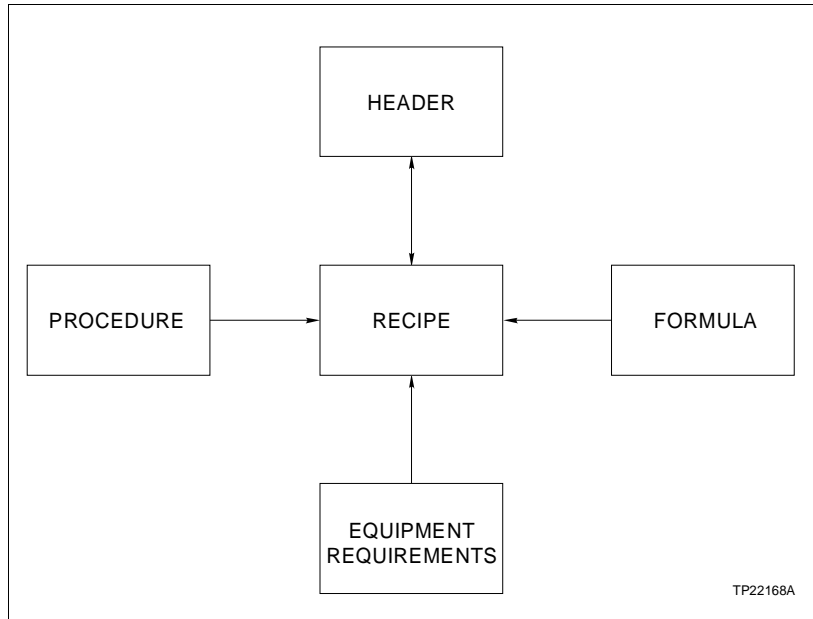


Figure 2-1. Composition of a Recipe

BATCH DATA MANAGER

The batch data manager software is an interactive tool for the creating, editing, managing, downloading (to MFP modules), and debugging of Batch 90 and User Defined Function (UDF) applications. Figures 2-2, 2-3, and 2-4 show the sequence of operations required to produce batch and UDF applications using the batch data manager. This figure assumes the appropriate function codes have been configured into the function blocks of the MFP module.

Batch Projects

Typically, a plant site is organized by processes. Each process is usually contained within a limited area for which boundaries can be defined. These process boundaries may be geographical or defined by the process. The walls of a building would be an example of a geographical boundary, while the completion of a specific operation may define a process boundary. Establishing these boundaries defines a process area or project and is important to the task of designing a control application. A large complex facility may contain many projects, while a small facility may be defined by only a single project. Equipment is grouped within the boundaries of a project. Projects are the foundation of the batch data manager system. Projects are the largest grouping of information for a batch application in the INFI-BATCH tools system.

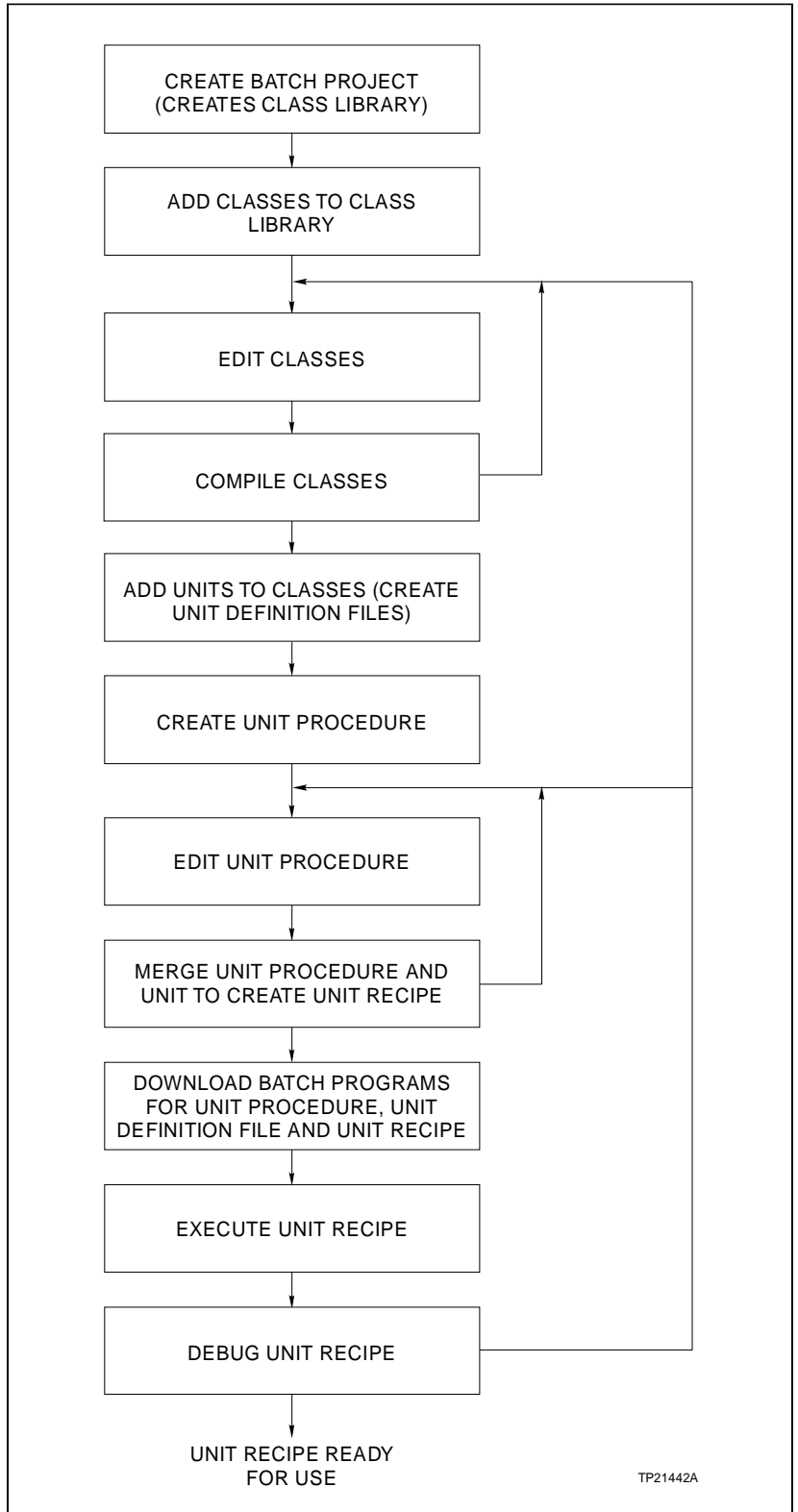


Figure 2-2. Unit Procedure Application Creation

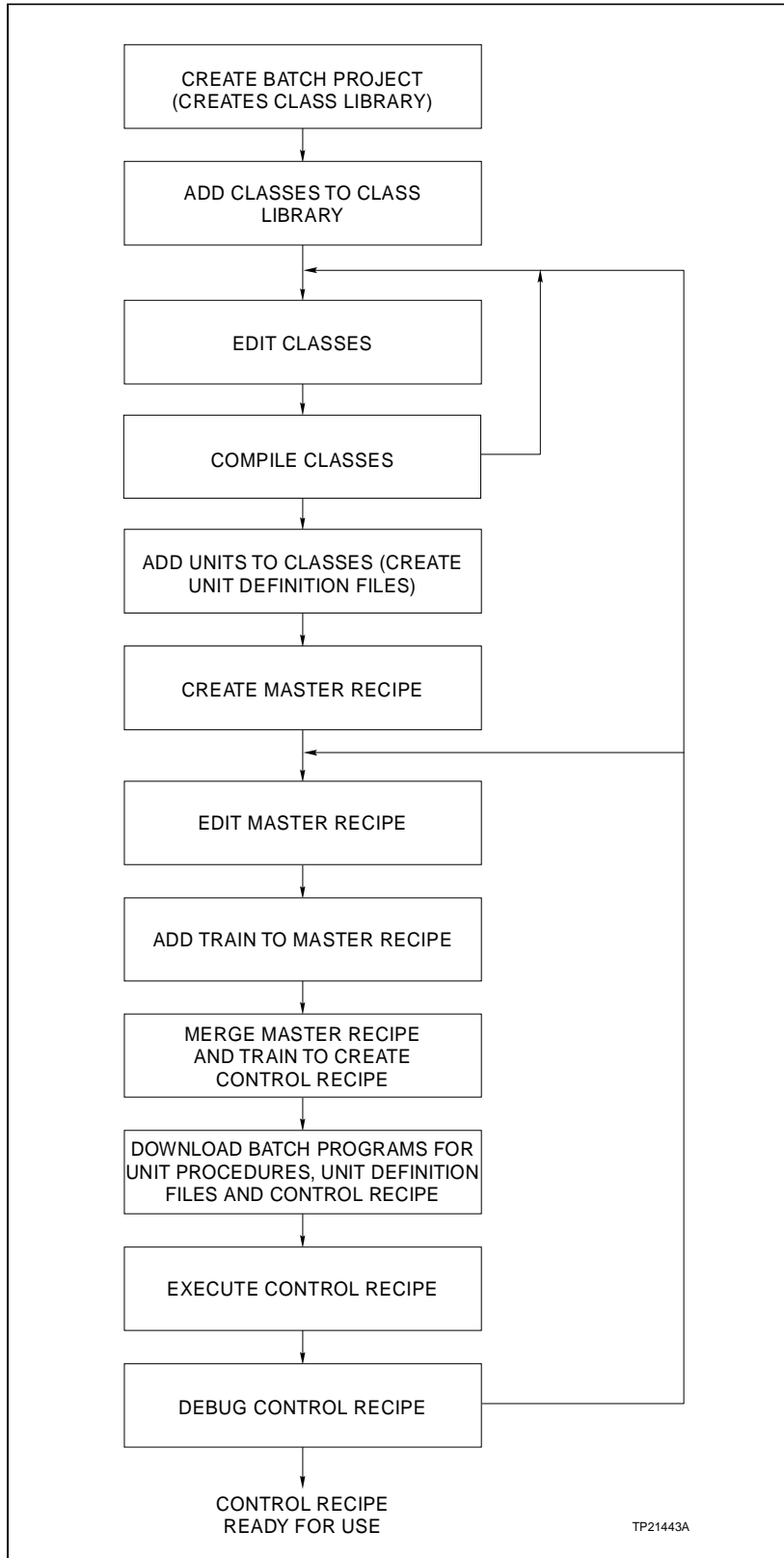


Figure 2-3. Master Recipe Application Creation

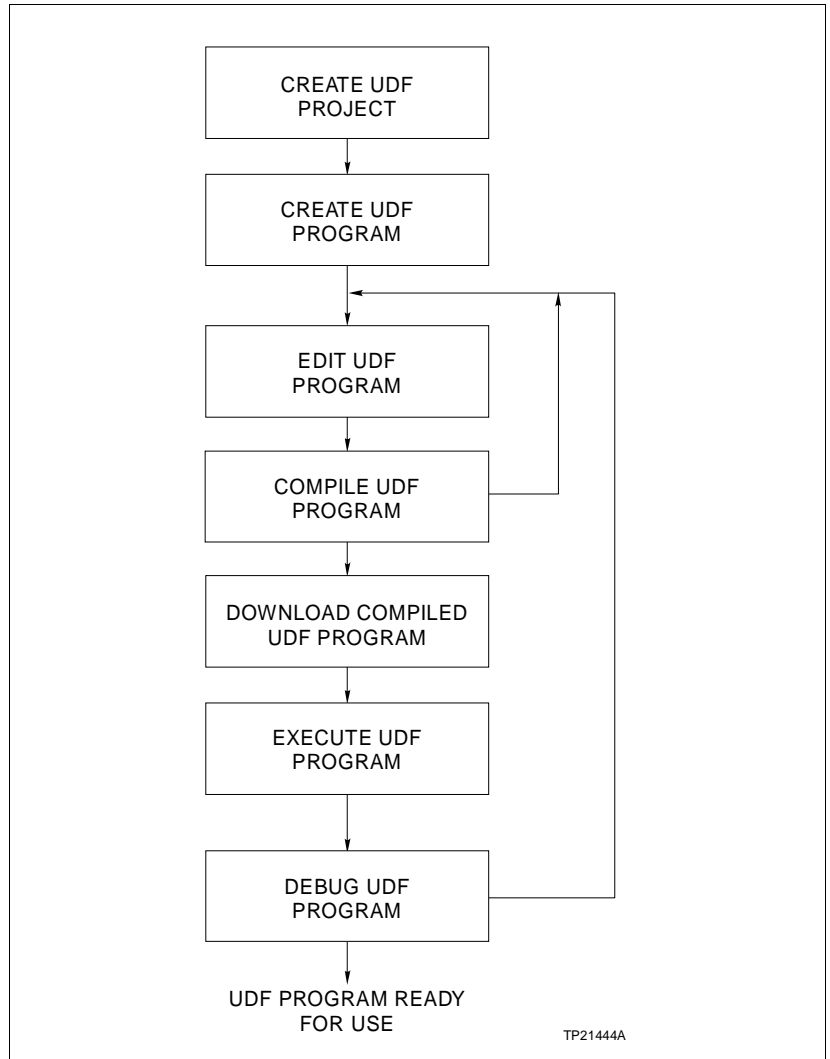


Figure 2-4. UDF Program Application Creation

CREATE PROJECT

The first step in creating a batch application is to create a project. The batch data manager uses projects to organize the many pieces of information that comprise a batch application. During the creation of a project a class library is created (or referenced). A class library is a directory that will hold classes.

ADD CLASSES

The second step in creating a batch application is to add a class to the class library for each different type of process equipment that exists within the process area associated with the project. Each class will have one Batch 90 program associated with it. However, one Batch 90 program may be associated with multiple classes.

CREATE CLASS PROGRAMS

The third step in creating a batch application is to create a batch program that contains (among other things) phase sub-routines that describe all the process actions possible by a certain type of equipment. Add or reference this program to the class to create a generic list of process actions supported by all equipment of this type.

ADD UNIT DEFINITION FILES

The fourth step in creating a batch program is to create a unit (unit definition file) for each specific process unit within the equipment class. The unit definition file contains unit specific information such as constant values (capacity, design pressure, etc.), unit specific strings (unit name, material of construction, etc.) and block address information for unit specific control devices (device drivers, common sequences, multi-state device drivers, etc.).

DEVELOP RECIPES

The fifth step in creating a batch application is to develop the recipes used to produce a batch of product. Master recipes and unit procedures are two ways of developing recipes.

Unit Procedures and Unit Recipes

Unit procedures specify the sequence of operations and phases a class of equipment must perform to produce a batch of product. Like the class program, this file is generic to all equipment of the type specified by the class. The unit procedure (generic) and a unit definition file (specific) are merged to create a unit specific recipe that will produce a batch of product on the equipment specified in the unit. This recipe is called a unit recipe.

In order to execute a unit procedure on a specific process unit, generate a unit recipe by merging the unit procedure with the unit definition file for the process unit and download the resulting unit recipe.

Master Recipes and Control Recipes

The ability to utilize master recipes is an optional feature that provides for the easy management of processes that involve multiple process units. Within the INFI-BATCH tools, master recipes are represented by a collection of unit procedures and equipment requirements in the form of equipment classes. Production trains are created by selecting specific process units from the identified equipment classes. The combination of master recipes with production trains results in the creation of control recipes.

The Batch Recipe Manager (BRM) is an optional software package that simplifies recipe management for processes that involve multiple process units in the production strategy. The

batch recipe manager provides the ability to organize unit procedures and production trains within master recipes. The Master Recipe Editor (MRE), which is contained in batch recipe manager, enables the organization of individual unit procedures required for the complete production process. The master recipe editor also allows the creation of production trains, which when merged with the master recipe results in a control recipe.

DOWNLOAD PROGRAMS, FILES, AND RECIPES

The sixth step in creating a batch application is to download the necessary programs, files, and recipes to the MFP module that will execute them. The Batch Download Utility (BDU) provides for the downloading of information and the maintenance of files within the MFP module. Use the batch download utility to download batch programs, UDF programs, unit definition files, unit recipes, and control recipes. The batch data manager makes the downloading of programs, files, and recipes quick and easy.

EXECUTE RECIPES

The seventh step in creating a batch application is to try to execute the recipe that makes up the application. Carefully monitor all important quantities and verify proper recipe execution.

DEBUG RECIPES

The eighth and final step in creating a batch application is to correct any errors or control logic problems. The Dynamic Debugger (DBG) provides the functions necessary to debug programs and recipes. For example, the debugger permits single step execution, displaying of internal variable values, halting execution, setting of execution break points, restarting execution, etc.

UDF Projects

The purpose of the UDF programming language is to create function codes that perform unique algorithms. Projects are the foundation of the batch data manager system. Projects are the largest grouping of information about a UDF process.

CREATE PROJECT

The first step in creating a UDF application is to create a UDF project. The batch data manager uses projects to organize the pieces of information that make up a UDF application.

CREATE PROGRAM

The second step in creating a UDF project is to create a UDF program that specifies the series of operations needed to perform the required algorithm. Typical applications involve complex sequential operations with many steps.

DOWNLOAD PROGRAM

The third step in creating a UDF application is to download the UDF program to the MFP module in which it will execute. The batch data manager makes the downloading of UDF programs quick and easy.

EXECUTE PROGRAM

The fourth step in creating a UDF application is to try to execute the program. Carefully monitor all important quantities and verify proper program execution.

DEBUG PROGRAM

The fifth and final step in creating a UDF application is to correct any errors or control logic problems. The batch debugger provides the tools needed to execute the program one line at a time or in groups of lines in order to find any errors or problems.

BATCH 90 SUPPORT

The Batch 90 language provides sequential control for batch applications. Write batch programs to support the operation of a single process unit. In order to be compatible with complex, multiple unit processes, use the batch data manager to assemble the various unit procedures into an integrated master recipe. To accomplish this integration, the batch programs for each class of equipment in the process must include certain data structures used by the batch data manager. Within a batch program there exists three different program sections used for data declarations. The unit parameters section contains recipe parameters that are global within a unit procedure. The batch data section contains global data structures for all units within the equipment class. Prior to the advent of unit relative programming, batch data included function block declarations for control devices. The unit data section was created to provide unit relative programming capability, thus unit specific function block declarations.

Unit Data Declarations

Include all unit specific information in the unit data section of batch programs to be used by the batch data manager. All data

that is not unit specific remains in the batch data section of the program. The following is an example of the unit data declaration structure.

```
UNIT DATA
  BLOCK INLET_VALVE, FC = DD
  BLOCK OUTLET_VALVE, FC = DD
  .
  .
  .
  STRING UNIT_NAME (16)
  STRING UNIT_CAPACITY
END DATA
```

Unit Parameter Declaration

Within the INFI-BATCH tools, recipe dependent connections between process units are created by CSEQ declarations in the unit parameter section. The unit parameter section may also be used to declare other recipe parameters.

The general form of the unit parameters section is:

```
UNIT PARAMETERS
  TYPE1    NAME1
  TYPE2    NAME2
  .
  .
  .
END DATA
```

If utilized, the unit parameter section must be inserted into the batch program before all the phase subroutines in the program. Each batch program can have only one unit parameter section.

When common sequences (CSEQ) are passed via the unit parameters, they are used by the batch data manager to define the connections between unit recipes. The syntax for these declarations in the unit parameter section is:

CSEQ *name*, **CLASS** = (*class1*, *class2*, ..., *classN*)

-or-

CSEQ *name*, **LIST** = "*filename*"

where:

name Name on the common sequence that is used by the phase subroutines.

<i>class</i>	Name of the classes used in the common sequence.
<i>filename</i>	Name of the ASCII text file that contains CSEQ declarations written in the format as required by the CSEQ declarations in the BATCH DATA section.

UDF PROGRAMS

UDF programs are used primarily to create function blocks that perform unique algorithms. These programs combine the flexibility of the UDF high level programming language with the simplicity of function block configurations.

ON-LINE PROGRAM CHANGES

Downloading a program sends a copy of the program to the NVRAM memory of the target MFP module. This is the modules master copy of the program. When a batch sequence function block starts or restarts a batch program, it makes a working copy of the program in RAM memory. This allows a modified version of the program to be downloaded with no immediate effect on the working copy. The following modifications are **not** allowed during on-line changes:

- Changes to batch data, unit data, and unit parameters.
- Changes to recipe arguments and local data declarations of an active phase subroutine.
- Deletion of phase subroutines.
- Changes to the #MAXPARALLEL value.

If the modified program contains any of these modifications, a -21 runtime fault code will be generated.

To execute the new version of the program:

1. Put the existing program into holding (that is, set the run/hold input to hold and wait for the program to complete the hold logic).
2. Restart the program (that is, set the run/hold input to run).

The BSEQ function block makes a copy of the new program and checks for changes. Depending on the changes found, one of the following actions takes place:

- The program status changes to complete (inactive). The program can be started but all previous data will be lost.

This action is a result of finding changes in the batch data, unit data, and unit parameters of the new program. Changes to the active phase subroutine and recipe values will also cause this action.

- The program status remains in holding. The program can be started but all phase data will be lost. Batch data will not be effected. This action is a result of finding changes to the local data declarations section of the active phase subroutine. Execution will start at the beginning of the normal logic for each parallel phase subroutine of the current operation.
- The program status changes to running with all previous data intact. Execution will start at the beginning of the restart logic for each parallel phase subroutine of the current operation.

SECTION 3 - INSTALLATION

INTRODUCTION

This section describes how to install and start up the WB9040 (Batch 90 programming language for Windows), WUDF40 (UDF programming language for Windows), and WMRE40 (Master Recipe Editor for Windows) software packages. Also described is the security available from the batch data manager.

INSTALLATION

To install the WB9040, WUDF40, and WMRE40 software packages:

1. Insert the software CD-ROM disk, 5.25 floppy disk number one, or 3.5 floppy disk number one into the appropriate drive.
2. Select *Run* from the Windows 95 start menu.
3. In the run dialog box, type:

drive_designator:**SETUP.EXE**

where:

drive_designator Letter designator for the drive being used to load this software.

4. When prompted, select the directory that will hold the software and data information.

When the installation process is complete, *Batch Data Manager* and *Readme Notes* menu options will be added to the standard Windows start menu.

START-UP

To start up the WinB90, WinUDF, and WinMRE software packages:

1. Select *Batch Data Manager* from the start menu.
2. Enter a user name and password in the BDM logon dialog box.
3. Click on *OK*.

The batch data manager window will appear. Refer to Figure 3-1 for an example batch data manager window.

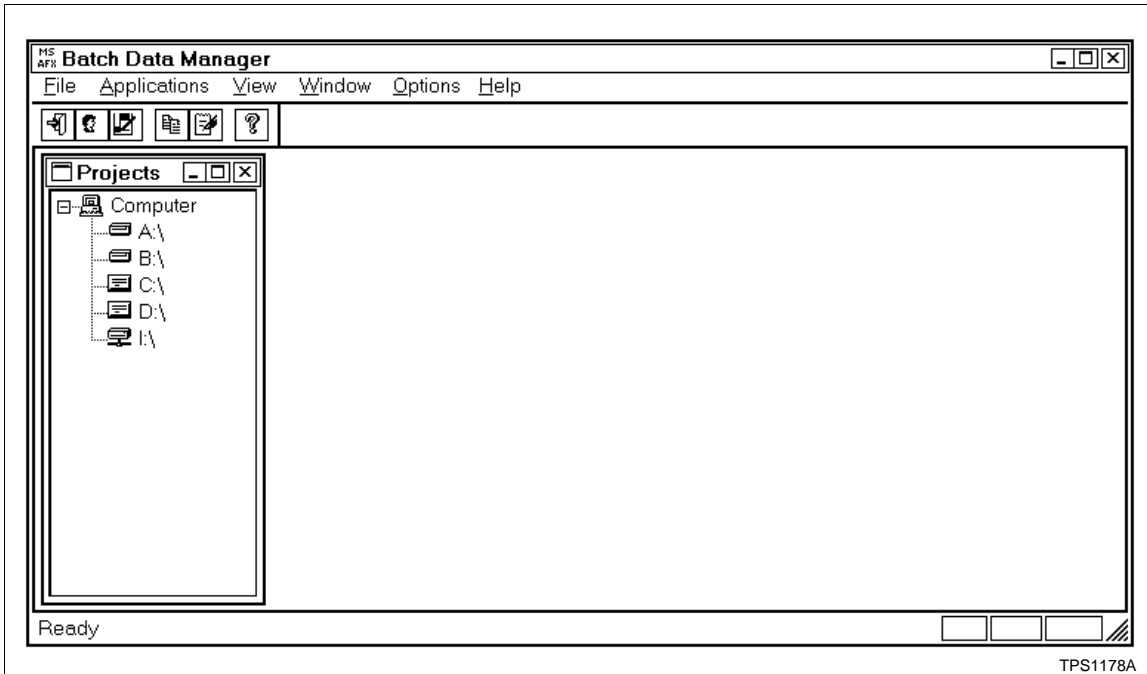


Figure 3-1. Batch Data Manager Window Example

DIRECTORY STRUCTURE

Figure 3-2 shows the directory structure of a working batch data manager system. Table 3-1 describes the files generated and in which directory they reside.

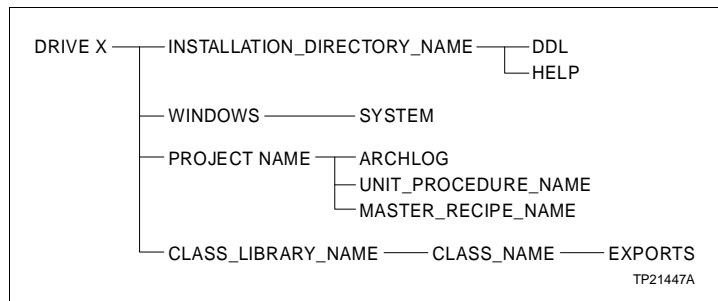


Figure 3-2. Batch Data Manager Directory Structure

Table 3-1. Batch Data Manager Files

Directory	File	Description
<i>installation_directory_name</i>	BDM.EXE	Batch data manager executable file.
	MRE.EXE	Master recipe editor executable file.
	UPE.EXE	Unit procedure editor executable file.
	BDU.EXE	Batch download utility executable file.
	UDF.EXE	UDF compiler executable file.
	B90.EXE	Batch 90 compiler executable file.
	DBG.EXE	Dynamic debugger executable file.
	ARLOG.EXE	Batch archive log executable file.
DDL	B90UR.B90	Batch 90 source templates.
	B90NUR.B90	Batch 90 source templates.
	FC.DEF	Batch 90 compiler file.
	RESWD.DEF	Batch 90 reserved words file.
	RESWD88.DEF	Batch 90 reserved words file.
	ERROR.DEF	Batch 90 error messages file.
	ERROR88.DEF	Batch 90 error messages file.
	B90CONST.INC	Batch 90 compiler file.
	UDFRESWD.DEF	UDF reserved words file.
	UDFERROR.DEF	UDF error messages file.
	UDF1.UDF	UDF source template file.
	UDF2.UDF	UDF source template file.
	PROJECT.DBD	Project database schema file.
	PROJECT.KEY	Project database key file.
	PROJECT1.DAT	Project database record file.
	ARCHLOG.DBD	Archive log database schema file.
	ARCHLOG.KEY	Archive log database key file.
	ARCHLOG.DAT	Archive log database record file.
	RECIPE.DBD	Database schema file.
	RECIPE.KEY	Database key file.
	RECIPE1.DAT	Database record file.
	RECIPE2.DAT	Database record file.
	RECIPE3.DAT	Database record file.
	SYMTBL.DBD	Symbol table database schema file.
SYMTBL.KEY	Symbol table database key file.	
SYMTBL1.DAT	Symbol table database record file.	
SYMTBL2.DAT	Symbol table database record file.	
SYMTBL3.DAT	Symbol table database record file.	

Table 3-1. Batch Data Manager Files (continued)

Directory	File	Description
HELP	BDM.HLP	Batch data manager help file.
	SYNTAX.HLP	Batch 90 and UDF syntax help file.
SYSTEM	GLIB.DLL	Shared executable file.
	MSVC.DLL	Shared executable file.
	RDM.DLL	Shared executable file.
<i>project_name</i>	LOCKFILE.LOK	Project multi-user locking file.
	PROJECT.DBD	Project database schema file.
	PROJECT.KEY	Project database key file.
	PROJECT1.DAT	Project database record file.
ARCHLOG	LOCKFILE.LOK	Archive log multi-user locking file.
	ARCHLOG.DBD	Archive log database schema file.
	ARCHLOG.KEY	Archive log database key file.
	ARCHLOG.DAT	Archive log database record file.
<i>unit_procedure_name</i>	LOCKFILE.LOK	Unit procedure multi-user locking file.
	RECIPE.DBD	Unit procedure database schema file.
	RECIPE.KEY	Unit procedure database key file.
	RECIPE1.DAT	Unit procedure database record file.
	RECIPE2.DAT	Unit procedure database record file.
	RECIPE3.DAT	Unit procedure database record file.
<i>master_recipe_name</i>	LOCKFILE.LOK	Master recipe multi-user locking file.
	RECIPE.DBD	Master recipe database schema file.
	RECIPE.KEY	Master recipe database key file.
	RECIPE1.DAT	Master recipe database record file.
	RECIPE2.DAT	Master recipe database record file.
	RECIPE3.DAT	Master recipe database record file.

Table 3-1. Batch Data Manager Files (continued)

Directory	File	Description
<i>class_name</i>	<i>program.B90</i>	Class Batch 90 source file.
	<i>program.OBJ</i>	Compiled class Batch 90 source file.
	<i>program.LST</i>	Class Batch 90 listing
	<i>program.LIS</i>	Class Batch 90 listing file with syntax highlighting
	<i>program.SYM</i>	Class Batch 90 symbol table file.
	<i>program.TEM</i>	Unit template file.
	CLASS.DEF	Class program location file.
	ADDRESS.DEF	Address specification file.
	<i>unit.UNT</i>	Unit definition source file.
	<i>unit.LST</i>	Unit definition listing file.
	<i>unit.LIS</i>	Unit definition listing file with syntax highlighting.
	<i>unit.OBJ</i>	Compiler unit definition file.
	LOCKFILE.LOK	Symbol table multi-user locking file.
	SYMTBL.DBD	Symbol table database schema file.
	SYMTBL.KEY	Symbol table database key file.
	SYMTBL1.DAT	Symbol table database record file.
	SYMTBL2.DAT	Symbol table database record file.
SYMTBL3.DAT	Symbol table database record file.	
EXPORTS	LOCKFILE.LOK	Export multi-user locking file.
	RECIPE.DBD	Export database schema file.
	RECIPE.KEY	Export database key file.
	RECIPE1.DAT	Export database record file.
	RECIPE2.DAT	Export database record file.
	RECIPE3.DAT	Export database record file.

SECTION 4 - PROJECTS

INTRODUCTION

The first step in developing any Batch 90 or UDF process control application is to start a project. A project is the largest grouping of information concerning the process control application. This section provides instructions on how to add, close, open, copy, delete, rename, edit user list of, and view properties of a project. How to refresh the projects window is also explained.

ADD

To add or create a project:

1. Verify no project is open.
2. In the **projects window**, right click on the designator of the drive that will contain the project.
3. Select *Add B90 Project...* or *Add UDF Project...* from the pop-up menu.
4. Enter the project name in the add project dialog box.

Project names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of eight characters can be used. Valid characters include letters A through Z, numbers zero through nine, and underscores.

5. Click on *Next*.
6. Specify if a new class library is to be created or an existing class library is to be referenced.
7. Click on *Next*.
8. If a new class library is to be created, enter the name of the subdirectory that will contain the class library and specify the class library path. If an existing class library is to be referenced, select the directory containing the class library.
9. Click on *Next*.
10. Verify the project name, project location, and class library name displayed in the add project dialog box is correct.
11. Click on *Next*.
12. Click on *Finish*.

CLOSE

To close a project:

1. In the **projects window**, right click on the name of the project to be closed.
2. Select *Close* from the pop-up menu.

All batch data manager application windows except the projects window will close.

OPEN

To open a project:

1. Verify no project is open.
2. In the **projects window**, right click on the name of the project to be opened.
3. Select *Open...* from the pop-up menu.

The unit procedures, master recipes (optional) and class library windows will appear for batch projects. The UDF window will appear for UDF projects. Refer to Figure 4-1 and Figure 4-2 for example new project windows.

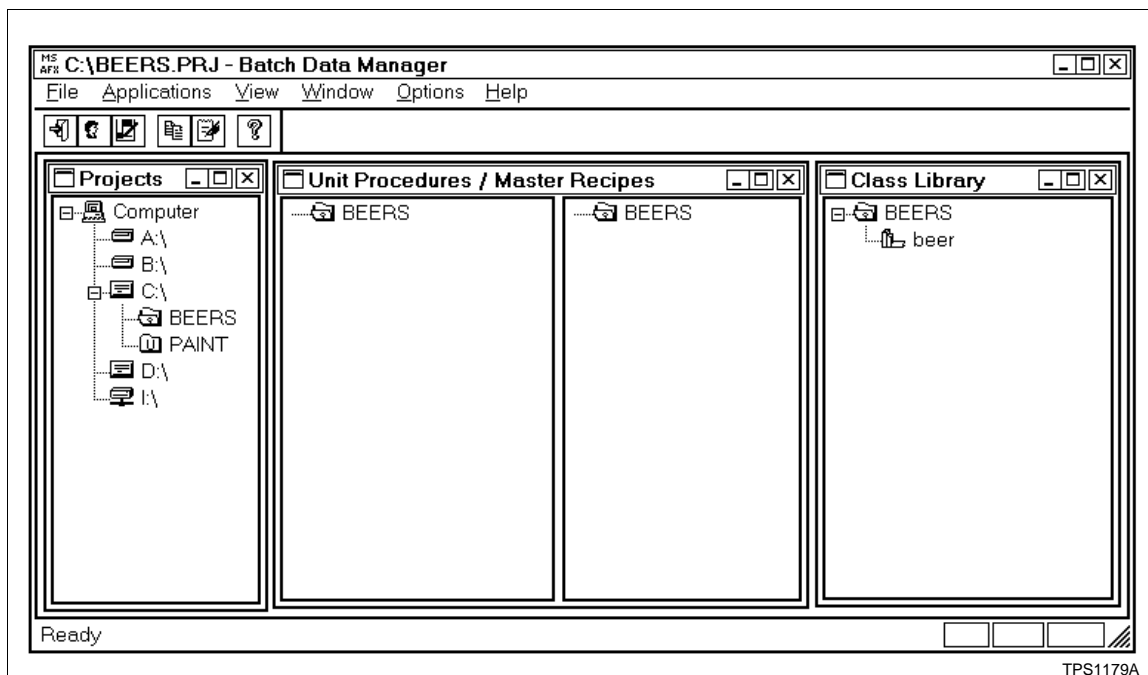


Figure 4-1. New Batch Project Windows Example

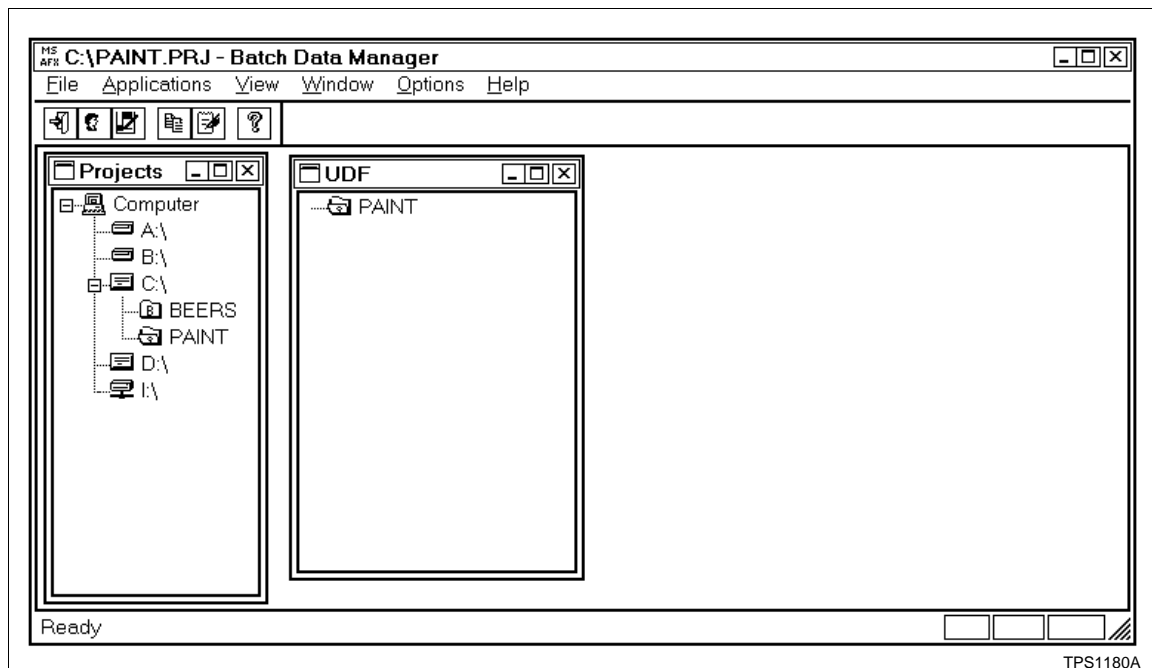


Figure 4-2. New UDF Project Window Example

COPY

To copy and rename an existing project:

1. In the **projects window**, right click on the project name.
2. Select *Copy...* from the pop-up menu.
3. Select the destination drive designator and enter the new project name in the copy project dialog box.
4. Click on *Next*.
5. Specify if unit procedures, master recipes and class libraries are to be copied also.
6. Click on *Next*.
7. Click on *Finish*.

DELETE

To delete a project:

1. Verify the project to be deleted is closed.
2. In the **projects window**, right click on the project name.
3. Select *Delete* from the pop-up menu.

- Click on **Yes** in the confirm project delete dialog box.

RENAME

To give a project a new name:

- Verify the project to be renamed is closed.
- In the **projects window**, right click on the project name.
- Select *Rename...* from the pop-up menu.
- Enter the new project name in the rename project dialog box.
- Click on **OK**.

EDIT USERS LIST

The *Edit Users List* option opens the selected project and displays the users list window. This window contains the names of the individual users that have some level of access to the selected project. Refer to Figure 4-3 for an example users list window. To display the users list window:

- Verify no project is open.

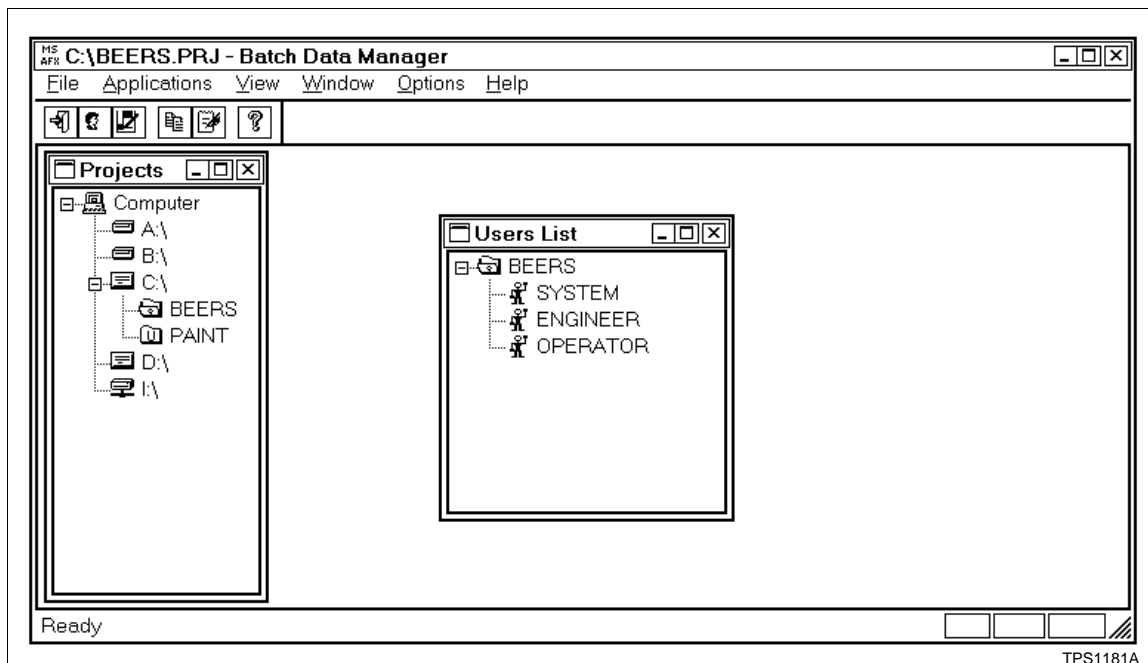


Figure 4-3. Users List Window Example

- In the **projects window**, right click on the name of the project containing the desired users list.

3. Select *Edit Users List* from the pop-up menu.

From the users list window, user names can be added or deleted and the properties of the list can be viewed. User privileges can be changed from this window also.

NOTE: Only the SYSTEM user can add or delete user names, modify user privileges, and reset user passwords.

Add User

To add the name and specify privileges of a new user to a project:

1. In the **users list window**, right click on the project name.
2. Select *Add User...* from the pop-up menu.
3. Enter the name of the new user in the user privileges dialog box.

User names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of eight characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. For batch projects, specify the level of access (view, edit, download, and debug) granted to the user for batch program, master recipe (optional), unit procedure, and unit definition files. For UDF projects, specify the level of access granted to the user for UDF program files.
5. Click on *OK*.

Delete All Users

To delete all user names (except the SYSTEM user) from the user list:

1. In the **users list window**, right click on the project name.
2. Select *Delete All Users* from the pop-up menu.
3. Click on *Yes* in the confirm user delete dialog box.

Refresh

To update the users list:

1. In the **users list window**, right click on the project name.
2. Select *Refresh* from the pop-up menu.

Delete

To delete a single user name (except the SYSTEM user) from the users list:

1. In the **users list window**, right click on the user name to be deleted.
2. Select *Delete* from the pop-up menu.
3. Click on *OK* in the confirm user delete dialog box.

Modify

To modify the access privileges of a user (except the SYSTEM user):

1. In the **users list window**, right click on the user name whose access privileges are to be modified.
2. Select *Modify...* from the pop-up menu.

The user privileges dialog box will appear.

3. For batch projects, edit the level of access (view, edit, download, and debug) granted to the user for batch program, master recipe (optional), unit procedure, and unit definition files. For UDF projects, edit the level of access granted to the user for UDF program files.
4. Click on *OK*.

Reset Password

Only the SYSTEM user can reset the password of other project users. The password will be reset to the user name (in upper case). Call the Technical Support department for information about resetting the SYSTEM user password.

To reset the password of a user:

1. In the **users list window**, right click on the user name whose password is to be reset.
2. Select *Reset Password...* from the pop-up menu.
3. Click on *Yes* in the confirm password reset dialog box.

PROPERTIES

There are three types of property displays available from the projects window. They are the drive, project, and users list properties.

Drive

The properties of a drive consist of:

- Type.
- Size.
- Number of projects contained.
- Label.
- Amount of space available.
- File system used.

To view the properties of a drive:

1. In the **projects window**, right click on the drive designator.
2. Select *Properties* from the pop-up menu.
3. View the desired properties in the properties dialog box.
4. Click on *OK*.

Project

The properties of a project consist of:

- Type (batch or UDF) of project.
- Location (drive and directory).
- Date created.
- Size of project directory and subdirectories in bytes.
- Number of unit procedures (batch only), master recipes (batch only), and programs (UDF only) contained in the project.
- Version number of the project.
- Class library location (batch only).
- A count of the individual users that have access to the project.
- Comments

To view or edit the properties of a project:

1. Right click on the project name in the **projects, unit procedures, master recipes** (optional), **class library**, or **UDF window**.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Users List

The properties of a user list include:

- Name.
- Location.
- Number of users on the list.

To view or edit the properties of a users list:

1. In the **users list window**, right click on the project name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

REFRESH

To refresh the listing of projects:

1. In the **projects window**, right click on the project name.
2. Select *Refresh* from the pop-up menu.

SECURITY

Security within the batch data manager consists of user names and passwords applied to projects on a per project basis. User names and passwords reside within a project and restrict or allow access to only that project. The batch data manager contains the default user name SYSTEM which has a password of BAILEY. This user should be the system administrator or other person responsible for the recipes. The system administrator should create the projects and edit the users list for each project to include the names of the necessary personnel. The access privileges of the users should be set by the system administrator.

Any user can create a new project and immediately become a defined user for that project. Only the SYSTEM user for a project can edit the users list for that project, edit the access privileges of users, rename the project, and delete the project.

User names are utilized as default passwords. Passwords can be changed by that user at any time. No user, not even the SYSTEM user can access the password of another user. However, the SYSTEM user can reset a users password to the user name. Refer to **Reset Password** in this section for more information about resetting a user password. Passwords can contain a maximum of eight ASCII characters. Passwords are case sensitive. To change a password:

1. Right click on the project name in the **projects window**.
2. Select *Open...* from the pop-up menu.
3. Right click on the project name in the **projects, unit procedures, master recipes, class library, or UDF window**.
4. Select *Change Password...* from the pop-up menu.
5. Enter the current password, new password, and the new password for a second time in the change user password dialog box.
6. Click on *OK*.

SECTION 5 - CLASS LIBRARIES AND CLASSES

INTRODUCTION

A class library is a directory that contains classes, each of which resides in a separate subdirectory. A class is the Batch 90 program and associated files that defines a sequence of operations for a specific type of process equipment. Units (discussed later in this instruction) define the individual pieces of equipment. For example, assume a plant has three machines of the same model and manufacturer that perform the process operations. The plant would require one class to define the equipment phases for any of the machines and three units to define each machine.

NOTE: Classes and class libraries are used in batch projects only.

CLASS LIBRARIES

Only one class library can be specified for each project. This is done during project creation. However, once a project has been created the class library associated with that project can be changed. To change the class library, the class library must be deleted or the reference to it must be removed from the project. Add the new class library to complete the change operation. It is also possible to rename and copy a class library.

Delete

To delete a class library and all associated files from the hard disk drive:

1. In the ***class library window***, right click on the class library name.
2. Select *Delete Class Library* from the pop-up menu.
3. Click on *Yes* in the confirm class library delete dialog box.

NOTE: Verify the class library is not used by any other project before deleting the class library.

Remove Reference

To remove the reference to a class library from a project:

1. In the ***class library window***, right click on the class library name.
2. Select *Remove Reference* from the pop-up menu.

3. Click on *Yes* in the confirm class library removal dialog box.

Add

To add a class library to a project:

1. Verify the project does not contain a class library.
2. In the **class library window**, right click on the project name.
3. Select *Add Class Library...* from the pop-up menu.
4. Specify if a new class library is to be created or an existing class library is to be referenced in the add class library dialog box.
5. Click on *Next*.
6. If a new class library is to be created, enter the name of the subdirectory that will contain the class library and specify the class library path. If an existing class library is to be referenced, select the directory containing the class library.

Class library names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers zero through nine, and underscores.

7. Click on *Next*.
8. Verify the class library name displayed in the add class library dialog box is correct.
9. Click on *Next*.
10. Click on *Finish*.

Rename

To rename a class library:

1. In the **class library window**, right click on the class library name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new name of the class library in the rename class library dialog box.

4. Click on *OK*.

Copy

To copy a class library from the current location to another location:

1. In the **class library window**, right click on the class library name.
2. Select *Copy Out...* from the pop-up menu.

The copy class library out dialog box will appear.

3. Set the batch copy options by specifying if symbol table, listing, object code, and include files are to be copied.
4. Set the unit copy options by specifying if unit, object code, and listing files are to be copied.
5. Specify if exports (explained in [Section 7](#) and [Section 8](#)) are to be copied.
6. Specify the destination directory.
7. Click on *OK*.

CLASSES

Classes can be added, copied within a library, copied to or from another class library directory, renamed, and deleted. Figure 5-1 shows an example class library window after classes are added to a class library.

Add

To add a new class to a class library directory:

1. In the **class library window**, right click on the class library name.
2. Select *Add Class...* from the pop-up menu.
3. Enter the class name in the add class dialog box.

Class names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of eight characters can be used. Valid characters include letters A through Z, numbers zero through nine, and underscores.

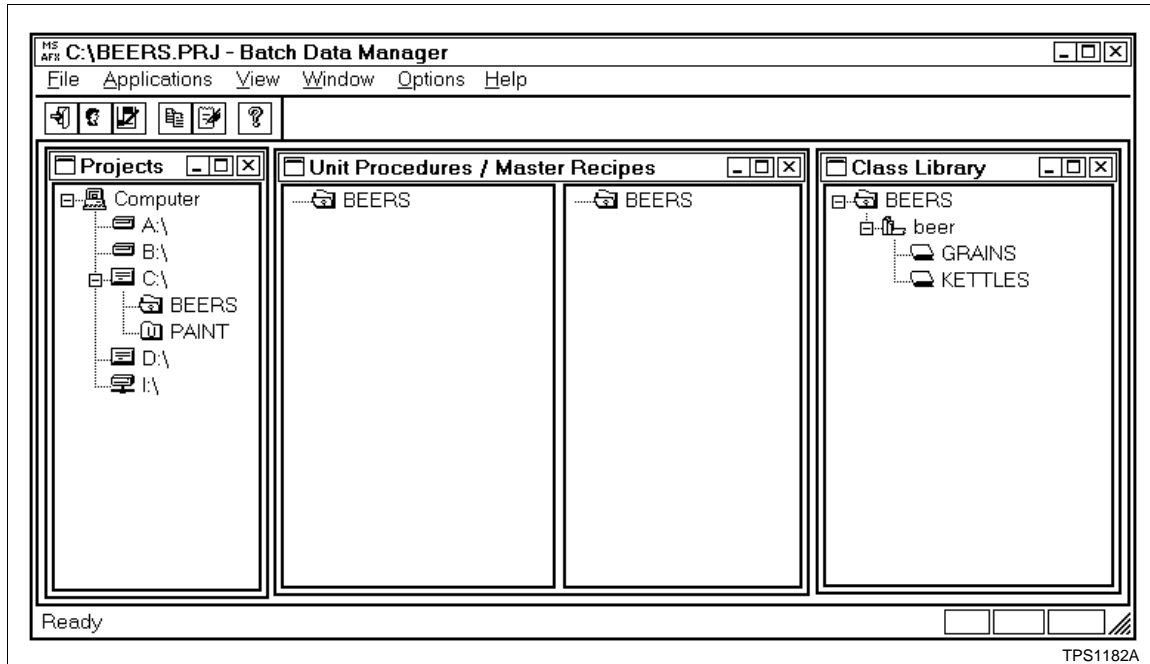


Figure 5-1. Class Library Window Example Containing Classes

4. Click on *Next*.
5. Specify if this class will use a new batch program (*New*), copy an existing program (*Copy*), or reference a program in another class library directory (*Link*). If a new program is used, select the unit data box to have a unit data section put in the program shell by the BDM software.
6. Click on *Next*.
7. If a new class batch program is to be created, proceed to Step 9. If an existing program is to be copied or referenced, select *Browse >>>* from the dialog box and specify the Batch 90 source program in the standard Windows open file dialog box.
8. Click on *Next*.
9. Verify the class name, class location, and batch source program location are correct.
10. If desired, click in the edit box and click on *Next* to edit the batch source program. Edit the batch program and select *Save* from the File menu of the Batch 90 editor window.
11. Click on *Finish*.

Copy

To make a copy of a class within the same class library directory and give it a new name:

1. In the **class library window**, right click on the class name.
2. Select *Copy...* from the pop-up menu.
3. Enter the name of the new class in the copy class dialog box.
4. Click on *OK*.

To copy all or part of a class from the current class library directory to another directory:

1. In the **class library window**, right click on the class name.
2. Select *Copy Out...* from the pop-up menu.

The copy class out dialog box will appear.

3. Set the batch copy options by specifying if symbol table, listing, object code, and include files are to be copied.
4. Set the unit copy options by specifying if source, object code, and listing files are to be copied.
5. Specify if exports (explained in [Section 7](#) and [Section 8](#)) are to be copied.
6. Specify the destination directory.
7. Click on *OK*.

To copy all or part of a class from a directory into the current class library directory:

1. In the **class library window**, right click on the destination class library name.
2. Select *Copy Class In...* from the pop-up menu.
3. Select the source directory and class name in the copy class in dialog box.
4. Enter a new name for the class (optional).
5. Set the batch copy options by specifying if symbol table, listing, object code, and include files are to be copied.

6. Set the unit copy options by specifying if source, object code, and listing files are to be copied.
7. Specify if exports (explained in [Section 7](#) and [Section 8](#)) are to be copied.
8. Click on *OK*.

Rename

To rename a class:

1. In the **class library window**, right click on the class name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new class name in the rename class dialog box.
4. Click on *OK*.

Delete

To delete a class from the class library directory:

1. In the **class library window**, right click on the class name.
2. Select *Delete* from the pop-up menu.
3. Click on *Yes* in the please confirm class delete dialog box.

To delete one or more classes from the class library directory:

1. In the **class library window**, right click on the class library name.
2. Select *Delete Selected...* from the pop-up menu.
3. Select the classes to be deleted in the delete classes dialog box.
4. Click on *OK*.

Change Class Program

To change the batch program associated with a class:

1. In the **class library window**, right click on the class name.
2. Select *Change B90 Program* from the pop-up menu.

3. Select the program in the standard Windows open dialog box.
4. Click on *Open*.

PROPERTIES

The class library and class property displays are available from the class library window. The properties of a class are also available from the Batch 90 editor window when editing a unit.

Class Library

The properties of a class library consist of:

- Type.
- Location and name.
- Size (in bytes).
- Number of classes contained in the class library.

To view or edit the properties of a class library:

1. In the **class library window**, right click on the class library name.
2. Select *Properties* from the pop-up menu.
3. View the properties in the properties dialog box.
4. Click on *OK*.

Class

The properties of a class include:

- Type.
- Location and name.
- Size of the class library directory (in bytes).
- Number of unit definition files associated with the class.
- Batch program location and name.
- Date the program was last compiled.
- Unit relative status (*YES* indicates the class is not specific to any unit, *NO* indicates the class is specific to a unit).
- Firmware revision number of the MFP module for which the class was compiled.

- Indication of usage of the #HIHI and #MAXPARALLEL directives.
- Program ID type.
- Program ID.
- A notice if the class has not yet been compiled.
- Address of units the class is specific to (non-unit relative classes only).
- Comment.

To view or edit the properties of a class from the **class library window**:

1. Right click on the class name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit the properties of a class while editing a unit of the class:

1. In the **Batch 90 editor window**, select *Properties* from the Class menu.
2. View or edit (if the current user has access) the desired properties in the properties dialog box.
3. Click on *OK*.

SECTION 6 - BATCH 90 AND UDF PROGRAMS

INTRODUCTION

This section describes how to create and manipulate Batch 90 programs, unit definitions, and UDF programs.

NOTE: Units (unit definitions) are used in batch projects only.

BATCH 90 PROGRAMS

The next step after adding a class to the class library (which specifies an existing or a new batch program) is to edit the program. This program will specify all the operations possible by the equipment controlled by this class. It is also possible to compile and print a batch program. Adding batch include files (described later in this section) to a class should be done at this time.

Edit

To edit an existing batch program for a class:

1. In the **class library window**, right click on the class containing the program.
2. Select *Edit* from the pop-up menu.

The Batch 90 editor window containing the opened program file will appear. Refer to Figure 6-1 for an example editor window.

3. Edit the program file as required.
4. Select *Save* from the File menu.

Compile

Settings

To view or edit the class settings used by the batch compiler when compiling a program file:

1. In the **Batch 90 editor window**, open the program file or verify it is the active program file.
2. Select *Settings* from the Class menu.

The compiler options dialog box will appear.

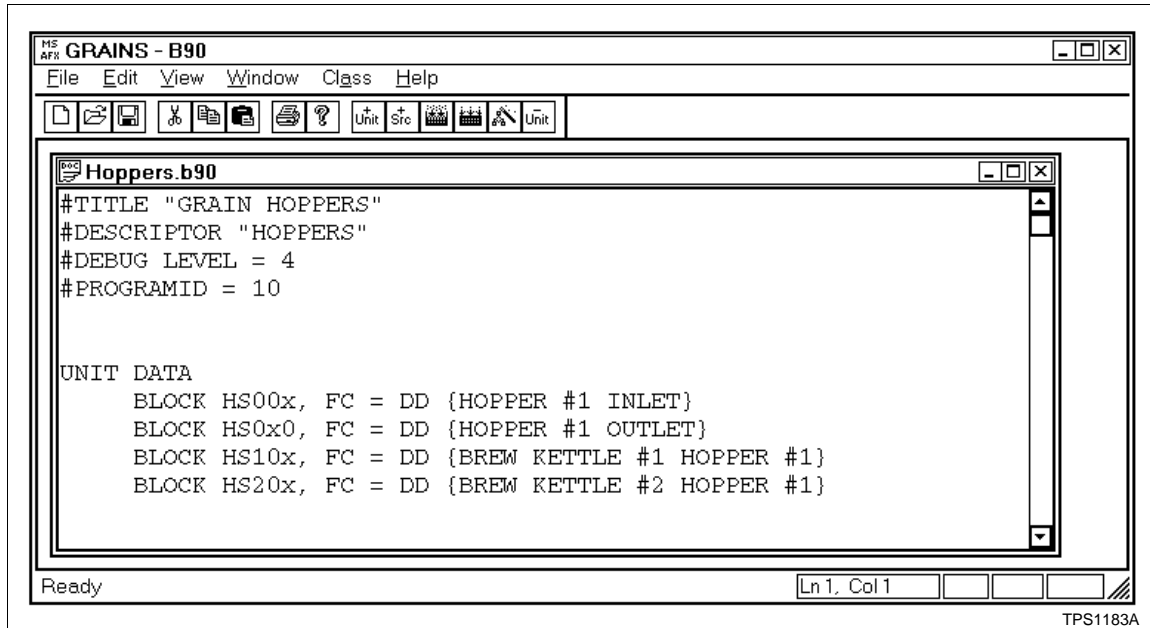


Figure 6-1. Batch 90 Editor Window Example

3. Under the main options tab, specify if a cross reference of program elements to sections of the program is to be put in the listing file.
4. Specify if the listing file should be left justified.
5. Specify if warnings should be suppressed from the listing file.
6. Specify if ISA-SP88.01-1995 standard terms are being used in the program.
7. Under the list options tab, specify if a hexadecimal version of the program with comments included should be downloaded into the listing file.
8. Specify if a pure hexadecimal version of the program is to be included in the listing file.
9. Specify if the contents of the symbol table file are to be included in the listing file.
10. Specify if the maximum stack size used is to be included in the listing file.
11. Under the versions tab, specify the firmware revision used when compiling.
12. Click on OK.

Compile To compile a batch program:

1. In the **Batch 90 editor window**, open the program file or verify it is the active program file.
2. Select *Compile* from the Class menu.

When the compiling operation is complete, the output window will display the number of error and warnings generated, time required to compile, value for specifications 11 and 12 of the batch sequence function block, and the minimum firmware version of the Batch 90 software used in the MFP module.

A listing file (**.lst** file extension) is created when the compiling operation completes. This file contains diagnostic information useful in finding the source of errors and warnings.

3. Correct the source of any errors and recompile the program file.

Print

To print a batch program:

1. In the **Batch 90 editor window**, open the program file or verify it is the active program file.
2. Select *Print...* from the File menu.
3. Specify the printer, print range, and copies settings in the standard Windows print dialog box.
4. Click on *OK*.

Source File

To add a batch include file to a class:

1. In the **class library window**, right click on the class name.
2. Select *Edit* from the pop-up menu.

The Batch 90 editor window containing the opened program file will appear.

3. Select *Add Source File* from the Class menu.

NOTE: A #INCLUDE directive is required in the program for it to locate and access the include file.

4. Select the *.B90* or *.INC* file extension in the add source file dialog box.

5. Click on *OK*.

An include file window will appear within the Batch 90 editor window.

6. Edit the include file as required.
7. Select *Save As* from the File menu.
8. Specify a destination directory and file name or accept the default values in the standard Windows save as dialog box.
9. Click on *Save*.

UDF PROGRAMS

After creating a project, the next step in creating a UDF project is to specify an existing or create a new UDF program. This program will specify the sequential operation of the equipment controlled by this project. UDF programs can be copied within a project, copied to or from other project directories, compiled, renamed, printed, and deleted. Adding UDF include files to a project can be done at this time.

Add

To add a new UDF program to a project:

1. In the **UDF window**, right click on the name of the project that will receive the new program.
2. Select *Add UDF...* from the pop-up menu.
3. Select the source file type and enter the name of the UDF program file in the add source file dialog box.

UDF program names can be any valid DOS file name.

4. Click on *OK*.

The UDF editor window containing the opened program file will appear. Refer to Figure 6-2.

5. Edit the program file as required.
6. Select *Save* from the File menu.

Edit

To edit an existing UDF program within a project:

1. In the **UDF window**, right click on the program file name.

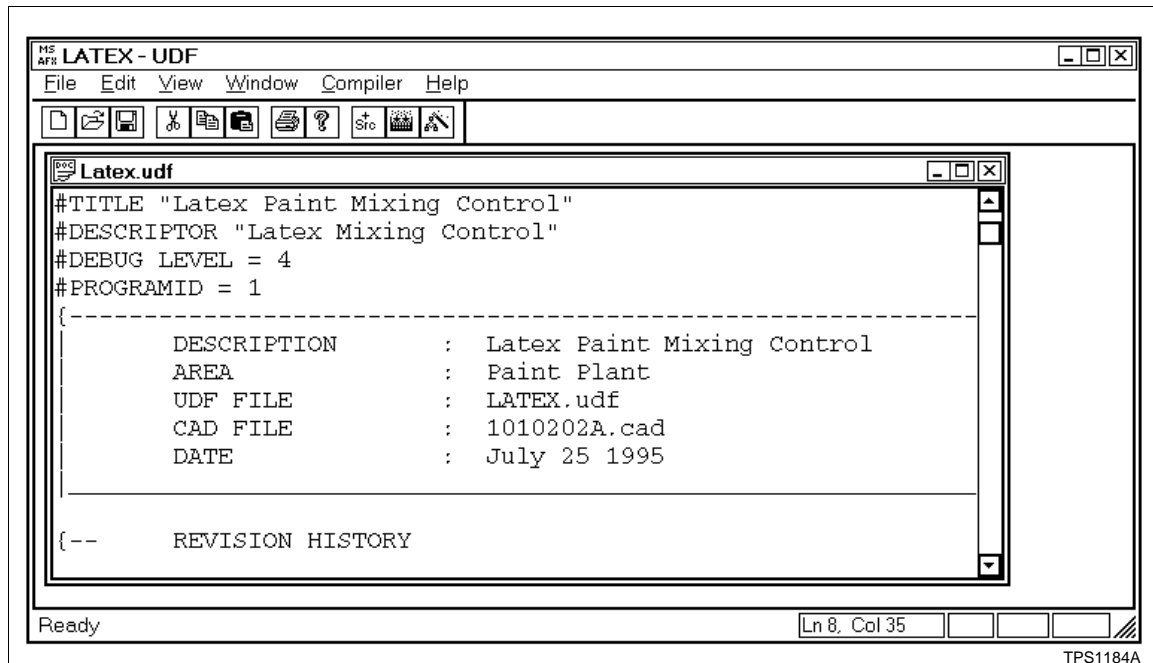


Figure 6-2. UDF Editor Window Example

2. Select *Edit* from the pop-up menu.

The UDF editor window containing the opened program file will appear.

3. Edit the program file as required.
4. Select *Save* from the File menu.

Copy

To make a copy a UDF program in the same program directory and give it a new name:

1. In the **UDF window**, right click on the program file name.
2. Select *Copy...* from the pop-up menu.
3. Enter the name of the new program file in the copy UDF dialog box.
4. Click on *OK*.

To copy all or part of a UDF program from the current program directory to another directory:

1. In the **UDF window**, right click on the name of the program file to be copied.

2. Select *Copy Out...* from the pop-up menu.

The copy UDF out dialog box will appear.

3. Set the UDF copy options by specifying if debugger symbol table, listing, and MFP object files are to be copied.
4. Select the destination directory.
5. Click on *OK*.

To copy all or part of a UDF program from another directory to the current project directory:

1. In the **UDF window**, right click on the name of the project that will receive the program file.
2. Select *Copy UDF In...* from the pop-up menu.
3. Select the source directory and program name in the copy UDF in dialog box.
4. Enter a new name for the program (optional).
5. Set the UDF copy options by specifying if symbol table, listing, and object files are to be copied.
6. Click on *OK*.

Compile

Options To view or edit the settings used by the UDF compiler when compiling a program file:

1. In the **UDF editor window**, open the program file or verify it is the active program.
2. Select *Settings* from the Compiler menu.

The compiler options dialog box will appear.

3. Under the main options tab, specify if a cross reference of program elements to sections of the program is to be put in the listing file.
4. Specify if the listing file should be left justified.
5. Specify if warnings should be suppressed from the listing file.
6. Under the listing options tab, specify if a hexadecimal version of the program with comments included should be downloaded into the listing file.

7. Specify if a pure hexadecimal version of the program is to be included in the listing file.
8. Specify if the contents of the symbol table file are to be included in the listing file.
9. Specify if the maximum stack size used is to be included in the listing file.
10. Under the versions tab, specify the firmware revision used when compiling.
11. Click on *OK*.

Compile To compile a UDF program:

1. In the ***UDF editor window***, open the program file or verify it is the active program file.
2. Select *Compile* from the Compiler menu.

When the compiling operation is complete, the output window will display the number of errors and warnings generated, time required to compile, program size, size of data for the UDF function block, and the minimum firmware version of the UDF software that must be used in the MFP module for the program to execute.

A listing file (***.lst*** file extension) is created when the compiling operation completes. This file contains diagnostic information useful in finding the source of errors and warnings.

3. Correct the source of any errors and recompile the program.

Delete

To delete a UDF program from a project:

1. In the ***UDF window***, right click on the name of the program file to be deleted.
2. Select *Delete* from the pop-up menu.
3. Click on *Yes* in the confirm UDF delete dialog box.

Rename

To rename a UDF program:

1. In the ***UDF window***, right click on the program name to be changed.

2. Select *Rename...* from the pop-up menu.
3. Enter the new name of the program in the rename UDF dialog box.
4. Click on *OK*.

Print

To print a UDF program:

1. In the **UDF editor window**, open the program file or verify it is the active program file.
2. Select *Print...* from the File menu.
3. Specify the printer, print range, and copies settings in the standard Windows print dialog box.
4. Click on *OK*.

Source File

To add a UDF include file to a UDF project:

1. In the **UDF window**, right click on a name of a program contained in the project.
2. Select *Edit* from the pop-up menu.

The UDF editor window containing the opened program file will appear.

3. Select *Add Source File* from the Compiler menu.

An include file window will appear within the UDF editor window.

4. Edit the include file as required.
5. Select *Save As...* from the File menu.
6. Specify a destination directory and file name or accept the default values in the standard Windows save as dialog box.
7. Click on *Save*.

UNITS

The next step after successfully compiling the batch program for a class is to add a unit (in the case of unit relative programs) for each piece of equipment using the class. The unit source file must be successfully compiled for the unit to

properly control the equipment. It is also possible to print and delete unit files. Figure 6-3 shows an example of units added to classes.

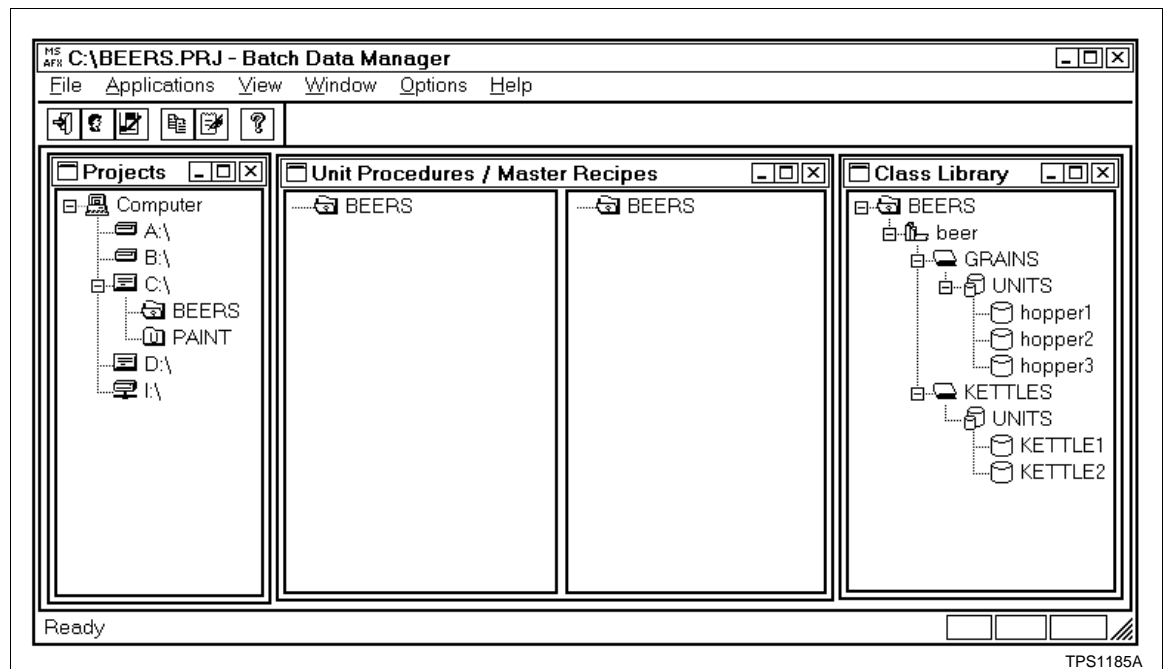


Figure 6-3. Class Library Window Example Containing Units

Add

To add a new unit to a class:

1. In the **Batch 90 editor window**, select *Add Unit...* from the Class menu.
2. In the unit file window, specify the unit source file title, descriptor, location (loop, PCU, module, and batch sequence function block number), and unit data.

Unit source file names can be any valid DOS file name.

3. Select *Save* from the File menu.

Compile

Settings

To view or edit the class settings used by the batch compiler when compiling a unit:

1. In the **Batch 90 editor window**, open the unit source file or verify it is the active file.
2. Select *Settings* from the Class menu.

The compiler options dialog box will appear.

3. Under the main options tab, specify if a cross reference of program elements to sections of the program is to be put in the listing file.
4. Specify if the listing file should be left justified.
5. Specify if warnings should be suppressed from the listing file.
6. Under the listing options tab, specify if a hexadecimal version of the program with comments included should be downloaded into the listing file.
7. Specify if a pure hexadecimal version of the program is to be included in the listing file.
8. Specify if the contents of the symbol table file are to be included in the listing file.
9. Specify if the maximum stack size used is to be included in the listing file.
10. Under the versions tab, specify the firmware revision used when compiling.
11. Click on *OK*.

Compile To compile a unit:

1. In the ***Batch 90 editor window***, open the unit source file or verify it is the active file.
2. Select *Compile* from the Class menu.

When the compiling operation is complete, the output window will display the number of errors and warnings generated, time required to compile, and the minimum firmware revision of the Batch 90 software that must be used in the MFP module.

A listing file (***.lst*** file extension) is created when the compiling operation completes. This file contains diagnostic information useful in finding the source of errors and warnings.

3. Correct the source any errors and recompile the unit source file.

Print

To print a unit source file:

1. In the **Batch 90 editor window**, open the unit source file or verify it is the active file.
2. Select *Print...* from the File menu.
3. Specify the printer, print range, and copies settings in the standard Windows print dialog box.
4. Click on *OK*.

Delete

To delete a unit from a class:

1. In the **Batch 90 editor window**, select *Delete Unit* from the Class menu.
2. Select the unit to be deleted in the delete unit dialog box.

Properties

The properties of a unit include:

- Type.
- Location and name.
- Size (in bytes).
- Number of unit recipes associated with this unit.
- Time and date when the unit was last compiled.
- Name of the class containing this unit.
- Time and date when the batch program associated with the class was last compiled.
- Name of the class library containing this unit.
- Firmware revision number of the MFP module for which this unit was compiled.

- Loop, module, PCU, and function block address of the unit source file.
- Indication if the class is unit relative or not.
- Batch program ID.
- Unit descriptor.

To view or edit the properties of a unit from the **class library window**:

1. Right click on the unit name.
2. Select *Properties* from the pop-up menu.
3. View (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit the properties of a unit from the **unit procedures window**:

1. Right click on the unit name.
2. Select *Properties* from the pop-up menu.
3. View (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit the properties of a unit from the **overview window** of the unit procedure editor:

1. Right click on the unit name.
2. Select *Properties* from the pop-up menu.
3. View (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

REBUILD

To compile the main batch program and all units of a class:

1. In the **Batch 90 editor window**, select *Rebuild All* from the Class menu.

The compiling operation output for the program and every unit of the class will be listed.

2. Correct the source of any errors and recompile that program or unit.

SECTION 7 - UNIT PROCEDURES AND UNIT RECIPES

INTRODUCTION

The next step after adding units to a class is to create a unit procedure, edit a unit procedure, and create a unit recipe. A unit procedure specifies how the equipment designated by the class is operated. The unit procedure and unit are merged to create a unit recipe that is downloaded into the MFP module. For example, a unit procedure might control how one of several possible chemical compounds is made using a certain type of reactor. The unit provides specific data about the particular reactor being used. The resulting unit recipe is specific to the type of chemical compound and the particular reactor being used. This section describes how to add, edit, copy within the project, copy to or from another project directory, rename, print, and delete a unit procedure or unit recipe. Figure 7-1 shows an example unit procedures window after unit procedures, units, and unit recipes have been added.

NOTE: Unit procedures and unit recipes are used in batch projects only.

UNIT PROCEDURES

More than one unit procedure can be added to a project. It is

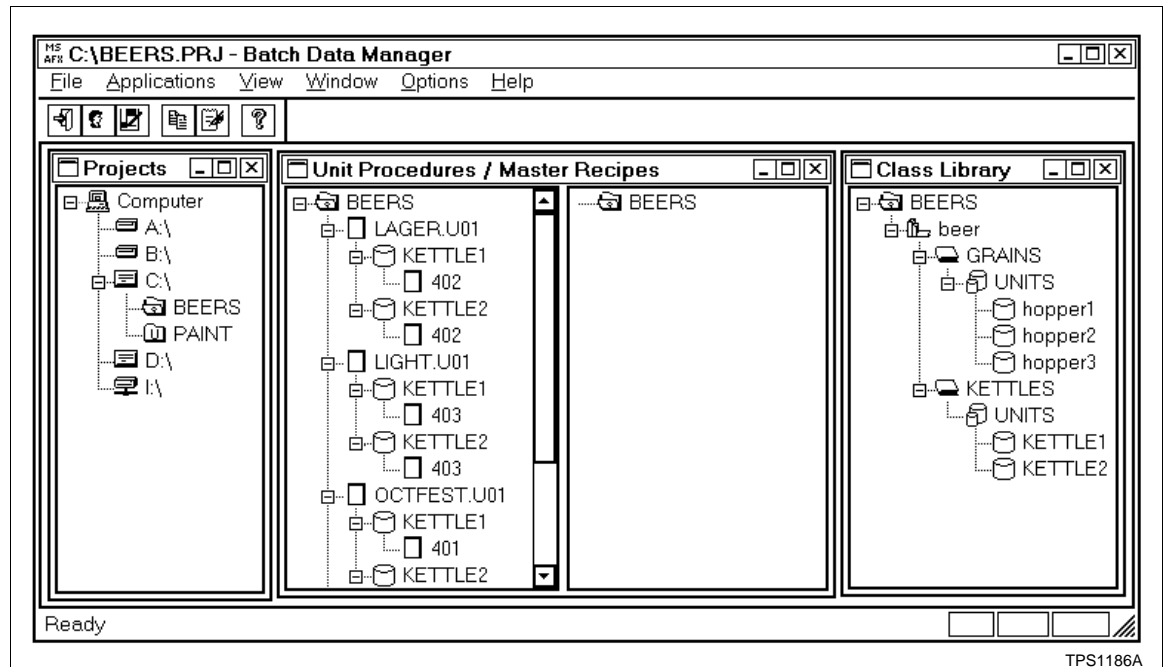


Figure 7-1. Unit Procedures Window Example

possible to add, edit, copy within the project, copy to or from another project directory, rename, print, and delete a unit procedure.

Add

To add a new unit procedure to a project:

1. In the **unit procedures window**, right click on the project name.
2. Select *Add Unit Procedure...* from the pop-up menu.
3. Enter the unit procedure name and select the class type under the unit procedure tab of the add unit procedure dialog box.

Unit procedure names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of eight characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Enter any comments under the comment tab.
5. Enter description information under the description tab.
6. Click on *OK*.

Edit

To edit a unit procedure:

1. In the **unit procedures window**, right click on the unit procedure name.
2. Select *Edit* from the pop-up menu.

The unit procedure editor window containing the unit procedure, overview, and output windows will appear. Figure 7-2 shows an example unit procedure editor window of a new and unedited unit procedure. Figure 7-3 shows an example unit procedure editor window of an fully edited unit procedure. The unit procedure window shows the unit procedure structure on top and a listing of the arguments associated with selected operation or phase on the bottom. The overview window displays all the unit recipes associated with the unit procedure. The output window displays general output messages.

3. Edit the unit procedure as required.
4. Select *Save* from the File menu.
5. Select *Exit* from the File menu.

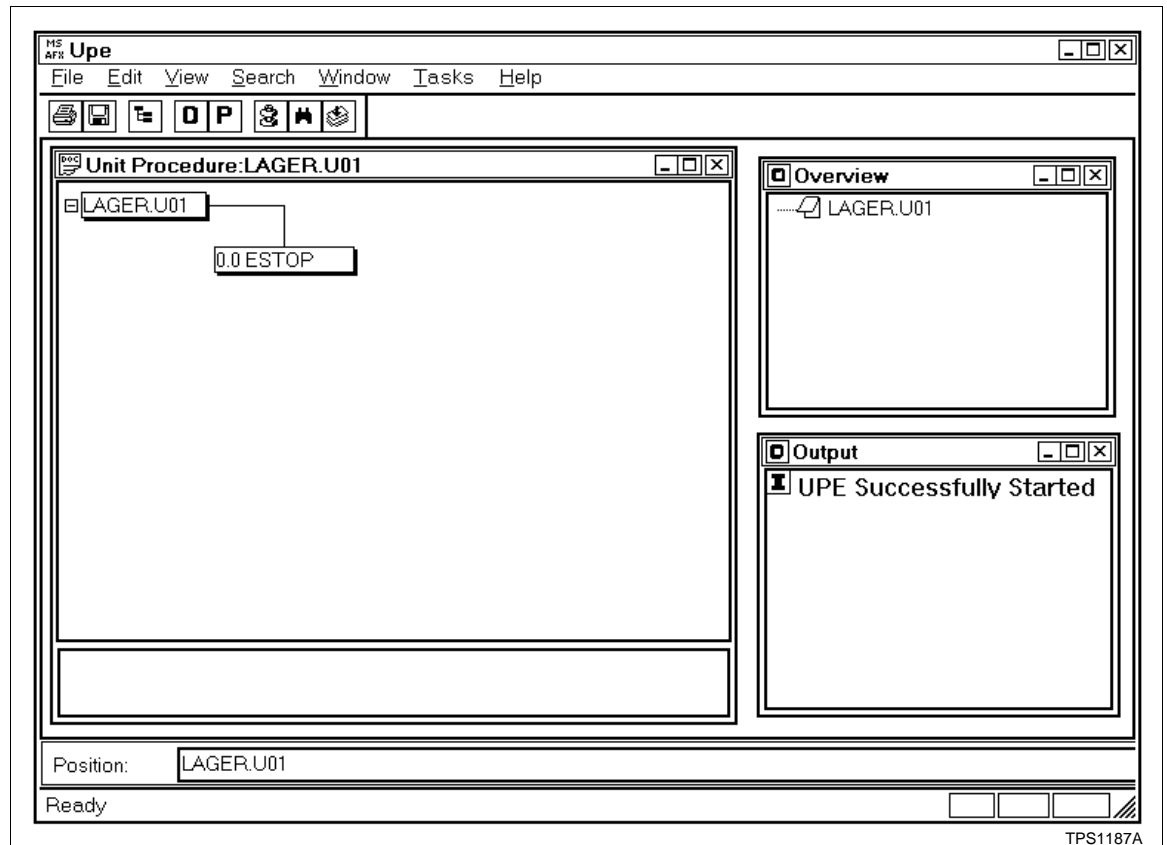


Figure 7-2. Unit Procedure Editor Window (Unedited) Example

The editing functions available in the unit procedure editor consists of:

- Inserting operations and phases.
- Exporting unit procedures operations and phases.
- Importing unit procedures, operations, and phases.
- Creating unit recipes.
- Renaming operations.
- Deleting operations and phases.
- Printing unit procedures.
- Changing write protection status.
- Contract or expand unit procedure structures.
- Searching for operations, phases, and arguments.
- Refreshing the internal class library RAM memory image.

Write Protect

When created, a unit procedure can be edited by any user with the appropriate access privileges. At any time, any user can change the unit procedure status to read only and become the owner of it. No user can edit the unit procedure until the owner changes the write protect status to the default state. The name of the user who enabled the read only status is displayed in the properties dialog box for as long as the reads only status is enabled.

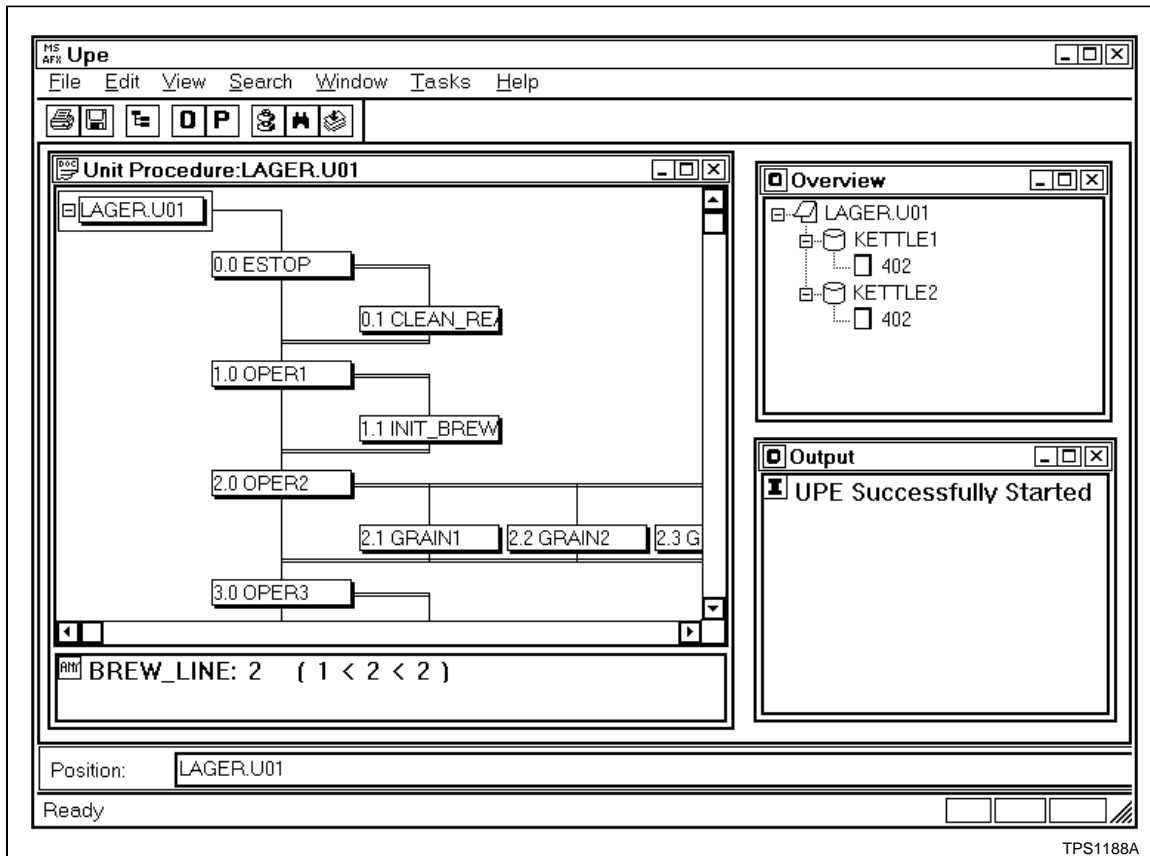


Figure 7-3. Unit Procedure Editor Window (Edited) Example

To change the write protect status of a unit procedure from the **unit procedure window** of the unit procedure editor:

1. Right click on the unit procedure name.
2. Select *Write Protect* from the pop-up menu.

Contract or Expand

To contract or expand the unit procedure structure (like a directory listing) from the **unit procedure window** of the unit procedure editor:

1. Right click on the unit procedure name.
2. Select *Contract/Expand* from the pop-up menu.

Export Unit Procedure

To send a copy of a unit procedure to an export subdirectory of the class directory from the **unit procedure window** or **overview window** of the unit procedure editor:

1. Right click on the unit procedure name.
2. Select *Export...* from the pop-up menu.

3. Enter the name of the unit procedure as it will appear in the export directory in the export dialog box.

4. Click on *OK*.

Import Unit Procedure

To import a copy of a unit procedure from the export directory, change the name to match the current unit procedure, and delete the current unit procedure from the ***unit procedure window*** of the unit procedure editor:

1. Right click on the name of the unit procedure that will be replaced by the imported unit procedure.

2. Select *Import...* from the pop-up menu.

3. Select the name of the unit procedure to be imported in the import unit procedure dialog box.

4. Click on *OK*.

Insert Operation

To insert a new operation into the current unit procedure from the menu bar of the unit procedure editor:

1. Verify the ***unit procedure window*** is the active window.

2. Select *Insert Operation...* from the Edit menu.

3. Enter the name of the inserted operation under the operation tab of the insert operation dialog box.

Operation names are forced into upper case, are not case sensitive, and can not begin with an underscore. A maximum of 32 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Specify the placement of the operation by selecting an operation and a modifier (before or after).

NOTE: Operations can not be inserted before the first operation (operation 0) of a unit procedure.

5. Enter any comments under the comment tab.

6. Click on *OK*.

To insert a new operation into the current unit procedure from the ***unit procedure window*** of the unit procedure editor:

1. Right click on an existing operation name or the name of the unit procedure that will receive the inserted operation.

2. Select *Insert Operation...* from the pop-up menu.

3. Enter the name of the inserted operation under the operation tab of the insert operation dialog box.

4. Specify the placement of the operation by selecting an operation and a modifier (before or after).

NOTE: Operations can not be inserted before the first operation (operation 0) of a unit procedure.

5. Enter any comments under the comment tab.

6. Click on *OK*.

Export Operation

To send a copy of an operation to an export subdirectory of the class directory from the ***unit procedure window*** of the unit procedure editor:

1. Right click on the operation name.

2. Select *Export...* from the pop-up menu.

3. Enter the name of the operation as it will appear in the export directory in the export dialog box.

4. Click on *OK*.

Import Operation

To import a copy of an operation from the export directory and add it to the current unit procedure from the ***unit procedure window*** of the unit procedure editor:

1. Right click on an existing operation name or the name of the unit procedure that will receive the imported operation.

2. Select *Import Operation...* from the pop-up menu.

3. Select the name of the imported operation as it will appear in the unit procedure in the import dialog box.

4. Select the name of the operation to be imported.

5. Specify the placement of the operation by selecting an operation and a modifier (before or after).

6. Click on *OK*.

Rename Operation

To give an operation a new name from the ***unit procedure window*** of the unit procedure editor:

1. Right click on the name of the operation receiving the new name.

2. Select *Rename...* from the pop-up menu.

3. Enter the new name in the rename operation dialog box.
4. Click on *OK*.

Add Phase

To insert a new phase into the current operation from the menu bar of the unit procedure editor:

1. Verify the ***unit procedure window*** is the active window.
2. Select *Add Phase...* from the Edit menu.
3. Specify the type and number of the inserted phase under the phase tab of the add phase dialog box.
4. Valid phase numbers range from one to the MAXPARALLEL value defined in the source program file for the class.
5. Specify the operation that will contain the phase.
6. If necessary, select the operation in which the phase will terminate under the superphase tab.

NOTE: Phases within the first operation (operation 0) of a unit procedure must terminate within the first operation.

7. Enter any comments under the comment tab.
8. Click on *OK*.

To add a new phase to an operation from the ***unit procedure window*** of the unit procedure editor:

1. Right click on an existing phase or the name of the operation that will receive the new phase.
2. Select *Add Phase...* from the pop-up menu.
3. Specify the type and number of the inserted phase under the phase tab of the add phase dialog box.
4. Specify the operation that will contain the phase.
5. If necessary, select the operation in which the phase will terminate under the superphase tab.

NOTE: Phases within the first operation (operation 0) of a unit procedure must terminate within the first operation.

6. Enter any comments under the comment tab.
7. Click on *OK*.

Export Phase To send a copy of a phase to an export subdirectory of the class directory from the **unit procedure window** of the unit procedure editor:

1. Right click on the phase name.
2. Select *Export...* from the pop-up menu.
3. Enter the name of the phase as it will appear in the export directory in the export dialog box.
4. Click on *OK*.

Import Phase To import a copy of a phase from the export directory and add it to the current operation from the **unit procedure window** of the unit procedure editor:

1. Right click on an existing phase name or the name of the operation that will be receive the imported phase.
2. Select *Import Phase...* from the pop-up menu.
3. Select the name of the phase, desired phase number, and the operation receiving the phase under the phase tab of the import dialog box.
4. If necessary, select the operation in which the phase will terminate under the superphase tab.

NOTE: Phases within the first operation (operation 0) of a unit procedure must terminate within the first operation.

5. Click on *OK*.

Terminate To change the operation in which a phase will terminate from the **unit procedure window** of the unit procedure editor:

1. Right click on the phase.
2. Select *Terminate...* from the pop-up menu.
3. Select the operation in which the phase will terminate in the terminate phase dialog box.

NOTE: Phases within the first operation (operation 0) of a unit procedure must terminate within the first operation.

4. Click on *OK*.

Delete To delete an operation or phase from the **unit procedure window** of the unit procedure editor:

1. Delete or change the termination operation of any super-phase that passes through the operation.

2. Right click on the operation or phase name.

NOTE: The first operation (operation 0) of a unit procedure can not be deleted.

3. Select *Delete* from the pop-up menu.
4. Click on *Yes* in the confirm phase delete dialog box.

Search

To search for a specific operation, phase, or argument from the **unit procedure window** of the unit procedure editor:

1. Select *Operation*, *Phase*, or *Argument* from the Search menu.
2. In the search dialog box, specify where to search and the name of the operation, phase, or argument.
3. Click on *Find Next* or *Find Previous*.

A box will appear around the found operation, phase, or argument.

4. Click on *OK* to exit the search dialog box.

Refresh Class Library

To update the RAM memory image of the class library, select *Refresh Class Library* from the Tasks menu of the **unit procedure editor window**.

Exiting Unit Procedure Editor

Save the edited unit procedure and select *Exit* from the File menu. If edited unit procedures are not saved before selecting *Exit*, the exit dialog box appears. In this dialog box, confirm the creation a new revision, overwriting the current revision, not saving, or creation a new unit procedure. When creating a new unit procedure, enter the new name.

Copy

To make a copy of a unit procedure in the same project directory and give it a new name or revision number:

1. In the **unit procedures window**, right click on the unit procedure name.
2. Select *Copy...* from the pop-up menu.
3. In the copy unit procedure dialog box, specify the creation of a new revision or an entirely new unit procedure. Enter the new unit procedure name when creating a new unit procedure.
4. Click on *OK*.

To copy all or part of a unit procedure from the current project directory to another directory:

1. In the **unit procedures window**, right click on the unit procedure name.
2. Select *Copy Out...* from the pop-up menu.
3. Select the destination directory in the copy unit procedure out dialog box.
4. Specify if the unit recipes are to be copied also.
5. Click on *OK*.

To copy a unit procedure from another directory to the current project directory:

1. In the **unit procedures window**, right click on the destination project name.
2. Select *Copy Unit Procedure In...* from the pop-up menu.
3. In the copy unit procedure in dialog box, select the source project directory and name of the unit procedure to be copied.
4. Enter the new name for the unit procedure.
5. Click on *OK*.

Rename

To give a unit procedure a new name:

1. In the **unit procedures window**, right click on the unit procedure name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new name of the unit procedure in the rename unit procedure dialog box.
4. Click on *OK*.

Print

To print all or part of a unit procedure:

1. In the **unit procedure window** of the unit procedure editor, select *Print...* from the File menu.

The unit procedure print options dialog box will appear.

2. Specify if the comments and unit parameters should be printed under the unit procedures tab. When printing unit parameters, specify if the default values, recipe values, and types are to be printed.
3. Specify how many spaces to indent the printout of unit procedures, comments, and unit parameters.
4. Under the operations tab, specify if the operations and operation comments are to be printed.
5. Specify how many spaces to indent the printout of operations and operation comments.
6. Under the phases tab, specify if the phases and phase comments are to be printed.
7. Specify how many spaces to indent the printout of the phases and phase comments.
8. Under the arguments tab, specify if the arguments, default values, recipe values, and types are also to be printed.
9. Specify how many spaces to indent the printout of arguments, default values, recipe values, and type.
10. Under the print profiles tab, specify if the print profile just configured is to be saved or an existing print profile is to be loaded. When saving a new print profile, enter a print profile name. When loading an existing print profile, select the print profile name from the list.
11. Click on *OK*.
12. Specify the printer, print range, and copies settings in the standard Windows print dialog box.
13. Click on *OK*.
14. In the printing stats dialog box, click on *OK* to confirm the number of bytes sent to the printer and total page count.

Delete

To delete a unit procedure from a project:

1. In the **unit procedures window**, right click on the unit procedure name.
2. Select *Delete* from the pop-up menu.
3. In the delete dialog box, specify if all revisions of the unit procedure should be deleted or just this revision.

4. Click on *OK*.

UNIT RECIPE

After the unit procedure is edited, a unit recipe should be created by merging the unit procedure and unit. It is possible to edit, copy, rename, and delete unit recipes also.

Add

To generate a new unit recipe and add it to the unit procedure from the menu bar of the unit procedure editor:

1. Verify the **unit procedure window** is the active window.
2. Select *Add Unit Recipe* from the File menu.
3. In the add unit recipe dialog box, enter a name or ID number for the unit recipe or select one from the list provided.

Unit recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric unit recipe names are one through 65,536.

4. Select the unit to be used when merging.
5. Click on *OK*.
6. If necessary, click on any error shown in the overview window. All windows of the unit procedure editor window will show conditions at the time of the error. Correct the error and repeat the merging process.

To generate a new unit recipe from the **unit procedure window** of the unit procedure editor:

1. Right click on the name of the unit procedure being used to create the unit recipe.
2. Select *Add Unit Recipe* from the pop-up menu.
3. In the add unit recipe dialog box, enter a name or ID number for the unit recipe or select one from the list provided.

Unit recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric unit recipe names are one through 65,536.

4. Select the unit to be used when merging.
5. Click on *OK*.
6. If necessary, click on any error shown in the overview window. All windows of the unit procedure editor window will show conditions at the time of the error. Correct the error and repeat the merging process.

To generate a new unit recipe from the **overview window** of the unit procedure editor:

1. Right click on the name of the unit procedure being used to create the unit recipe.
2. Select *Add Unit Recipe* from the pop-up menu.
3. In the add unit recipe dialog box, enter a name or ID number for the unit recipe or select one from the list provided.

Unit recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric unit recipe names are one through 65,536.

4. Select the unit to be used when merging.
5. Click on *OK*.
6. If necessary, click on any error shown in the overview window. All windows of the unit procedure editor window will show conditions at the time of the error. Correct the error and repeat the merging process.

Edit

To edit a unit recipe from the **unit procedures window**:

1. Right click on the unit recipe name.
2. Select *Edit...* from the pop-up menu.

The unit procedure editor window containing the unit recipe, overview, and output windows will appear. The unit recipe window shows the unit procedure structure on top and a listing of the arguments associated with selected unit procedure or phase on the bottom. The overview window displays the structure of the unit procedure after it has been compiled to create a unit recipe. The output window displays general output messages.

3. Edit the unit recipe as required.
4. Select *Save* from the File menu.
5. Select *Exit* from the File menu.

To edit a unit recipe from the unit procedure editor:

1. Verify that the **overview window** is the active window.
2. Right click on the unit recipe name.
3. Select *Edit...* from the pop-up menu.

The unit procedure editor window containing the unit recipe, overview, and output windows will appear.

4. Edit the unit recipe as required.
5. Select *Save* from the File menu.
6. Select *Exit* from the File menu.

The only editing function available in the unit recipe is the editing of unit procedure and phase arguments. To edit arguments:

1. In the **unit recipe window** of the unit procedure editor, click the unit procedure or phase containing the argument.
2. Double click on the argument.
3. Edit the argument as necessary in the edit dialog box.
4. Click on *OK*.

Copy

To make a copy of a unit recipe in the same project directory and give it a new name from the **unit procedures window**:

1. Right click on the unit recipe name.
2. Select *Copy...* from the pop-up menu.
3. Enter the new unit recipe name in the copy unit procedure dialog box.
4. Click on *OK*.

To make a copy of a unit recipe in the same project directory and give it a new name from the unit procedure editor:

1. Verify the **overview window** is the active window.
2. Right click on the unit recipe name.
3. Select *Copy...* from the pop-up menu.
4. Enter the new unit recipe name in the copy unit recipe dialog box.
5. Click on *OK*.

Rename

To rename a unit recipe from the **unit procedures window**:

1. Right click on the unit recipe name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new unit recipe name in the rename unit recipe dialog box.
4. Click on *OK*.

To rename a unit recipe from the unit procedure editor:

1. Verify the **overview window** is the active window.
2. Right click on the unit recipe name.
3. Select *Rename...* from the pop-up menu.
4. Enter the new unit recipe name in the rename unit recipe dialog box.
5. Click on *OK*.

Delete

To delete a unit recipe from the **unit procedures window**:

1. Right click on the unit recipe name.
2. Select *Delete...* from the pop-up menu.
3. Click on *Yes* in the confirm unit recipe delete dialog box.

To delete a unit recipe from the unit procedure editor:

1. Verify the **overview window** is the active window.
2. Right click on the unit recipe name.
3. Select *Delete...* from the pop-up menu.
4. Click on *Yes* in the confirm unit recipe delete dialog box.

To delete all unit recipes associated with a class from the **unit procedures window**:

1. Right click on the unit name.
2. Select *Delete All* from the pop-up menu.
3. Click on *Yes* in the confirm unit recipe delete all dialog box.

To delete all unit recipes associated with a unit from the unit procedure editor:

1. Verify the **overview window** is the active window.
2. Right click on the unit name.
3. Select *Delete All* from the pop-up menu.
4. Click on *Yes* in the confirm unit recipe delete all dialog box.

PROPERTIES

The unit procedure, operation, unit procedure editor, and unit recipe property displays are available from the **unit procedures window**.

Unit Procedure

The properties of a unit procedure include:

- Type.
- Location and name.
- Date the unit procedure was created.
- Size (in bytes).
- Number of units and unit recipes associated with this unit procedure.
- Date the unit procedure was last compiled.

- Type of unit procedure ID used.
- Write protection status.
- Indication if the unit procedure was merged.
- Notification if the unit procedure is currently opened.
- Name of the class associated with this unit procedure.
- Date the batch program for the class was last compiled.
- Name of the class library containing the class used by this unit procedure.
- Unit relative status.
- Firmware revision number of the module for which the unit recipe was merged.
- Comments.
- Unit procedure description.

To view or edit the properties of a unit procedure from the **unit procedures window**:

1. Right click on the unit procedure name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit the properties of a unit procedure from the **overview window** of the unit procedure editor:

1. Right click on the unit procedure name.
2. Select *Properties...* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit a subset of the properties of a unit procedure from the **unit procedure window** of the unit procedure editor:

1. Right click on the unit procedure name.

2. Select *Properties...* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Operation

The properties of an operation include:

- Name of the unit procedure that contains this operation.
- Number phases within the operation.
- Comment.

To view or edit the properties of an operation from the **unit procedure window** of the unit procedure editor:

1. Right click on the operation name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Phase

The properties of a phase include:

- Type.
- Name of operation containing this phase.
- Number of arguments used in this phase.
- Comment.

To view or edit the properties of a phase from the **unit procedure window** of the unit procedure editor:

1. Right click on the phase name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Unit Procedure Editor

The properties of the unit procedure editor include:

- Color of all objects used.
- Choice of displaying property name, number, or both.
- Option to change font size when object size changes.
- Width and height of objects.

To view or edit the properties of the unit procedure editor from the unit procedure editor:

1. Select *Properties* from the Edit menu or right click on any empty space in the **unit procedure window**.
2. View or edit (if the current user has access) the desired properties in the properties dialog box.
3. Click on *OK*.

Unit Recipes

The properties of a unit recipe include:

- Type.
- Location and name.
- Date first compiled.
- Size (in bytes).
- Date last compiled.
- ID type (numeric or alphanumeric).
- MaxParallel number.
- Address (loop, module, PCU, block).
- Recipe ID.
- Class associated with unit recipe.
- Program ID of the class batch program.
- Date the class batch program was compiled.
- Class library name.
- Designation if the unit recipe is unit relative or not.
- Firmware revision number of module containing this unit.
- Comment.
- Description.

To view or edit the properties of a unit recipe from the **unit procedures window**:

1. Right click on the unit recipe name.
2. Select *Properties* from the pop-up menu.

3. View or edit (if the current user has access) the desired properties in the properties dialog box.

4. Click on *OK*.

To view or edit the properties of a unit recipe from the unit procedure editor:

1. Right click on the unit recipe name in the ***overview window***.

2. Select *Properties* from the pop-up menu.

3. View or edit (if the current user has access) the desired properties in the properties dialog box.

4. Click on *OK*.

SECTION 8 - MASTER RECIPES AND CONTROL RECIPES

INTRODUCTION

The optional Master Recipe Editor software package (WMRE40) provides master recipe management features to the INFI-BATCH tool system. The ability to create master recipes is not required in order to make a product. Master recipes simplify the creation of the sequence of operations necessary to make a product.

NOTE: Master recipes and control recipes are used in batch projects only.

A master recipe is a collection of unit procedures that are linked together into a sequence of operations that control the making of a product. This sequence of operations or recipe, unlike unit procedures, can control multiple classes of equipment. The unit procedures that make up the master recipe and the units (specified in trains) are compiled to create control recipes which are downloaded into the MFP module. For example, a master recipe might control how one of several possible chemical compounds is made from beginning to end. The unit procedures within the master recipe would control how the chemical compounds are processed in each of the types of equipment (raw material loaders, mixers, packagers, etc.). Trains, which specify units, provide the specific data about a particular piece of equipment (raw material loader #5, mixer #7, packager #2, etc.). The resulting control recipes are specific to the type of chemical compound and the particular piece of equipment used. This section describes how to add, edit, copy within the project, copy to or from another project directory, rename, print, and delete a master recipe or control recipe. Figure 8-1 shows an example master recipe window after a master recipe, train, and control recipe have been added.

MASTER RECIPES

More than one master recipe can be added to a project. It is possible to add, edit, copy within the project, copy to or from another project directory, rename, print, and delete a master recipe.

Add

To add a new master recipe to a project:

1. In the **master recipes window**, right click on the project name.

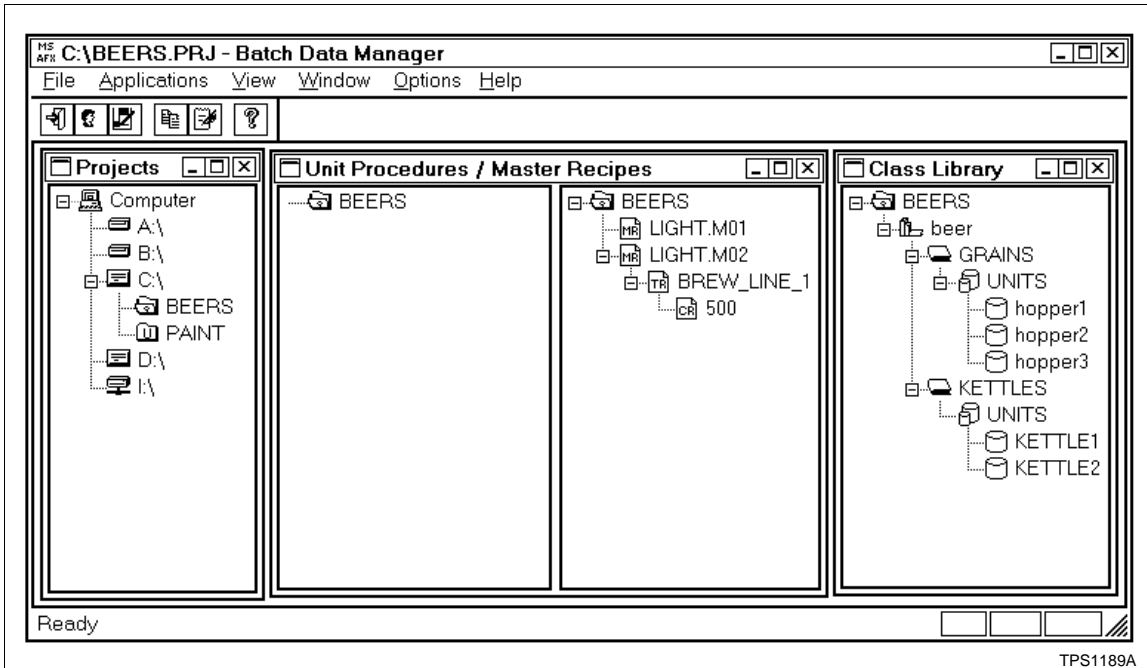


Figure 8-1. Master Recipes Window Example

2. Select *Add Master Recipe...* from the pop-up menu.
3. Enter the master recipe name, select the type of recipe ID, and enter any description information under the master recipe tab of the add master recipe dialog box.

Master recipe names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of eight characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Enter any comments under the comment tab.
5. Click on *OK*.

Edit

To edit a master recipe:

1. In the **master recipes window**, right click on the master recipe name.
2. Select *Edit* from the pop-up menu.

The master recipe editor window containing the master recipe, overview, and output windows will appear. Figure 8-2 shows an example master recipe editor window of a new and unedited

master recipe. Figure 8-3 shows an example master recipe editor window of a fully edited master recipe. The master recipe window shows the master recipe structure on top and a listing of the arguments associated with selected operation or phase on the bottom. The overview window displays the structure of the master recipe after it has been compiled to create a control recipe. The output window displays general output messages.

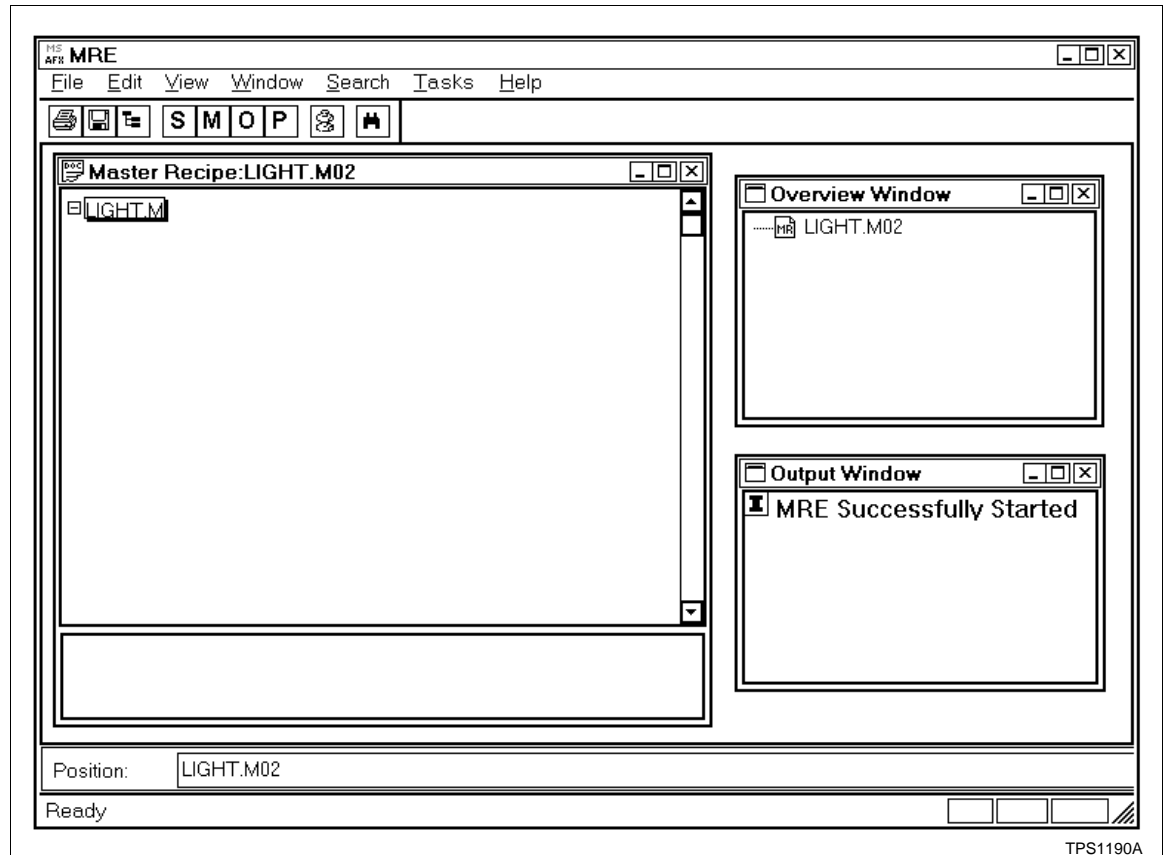


Figure 8-2. Master Recipe Editor Window (Unedited) Example

3. Edit the master recipe as required.
4. Select *Save* from the File menu.
5. Select *Exit* from the File menu.

The editing functions available in the master recipe editor consists of:

- Inserting segments, operations, and phases.
- Exporting unit procedures operations and phases.
- Importing unit procedures, operations, and phases.

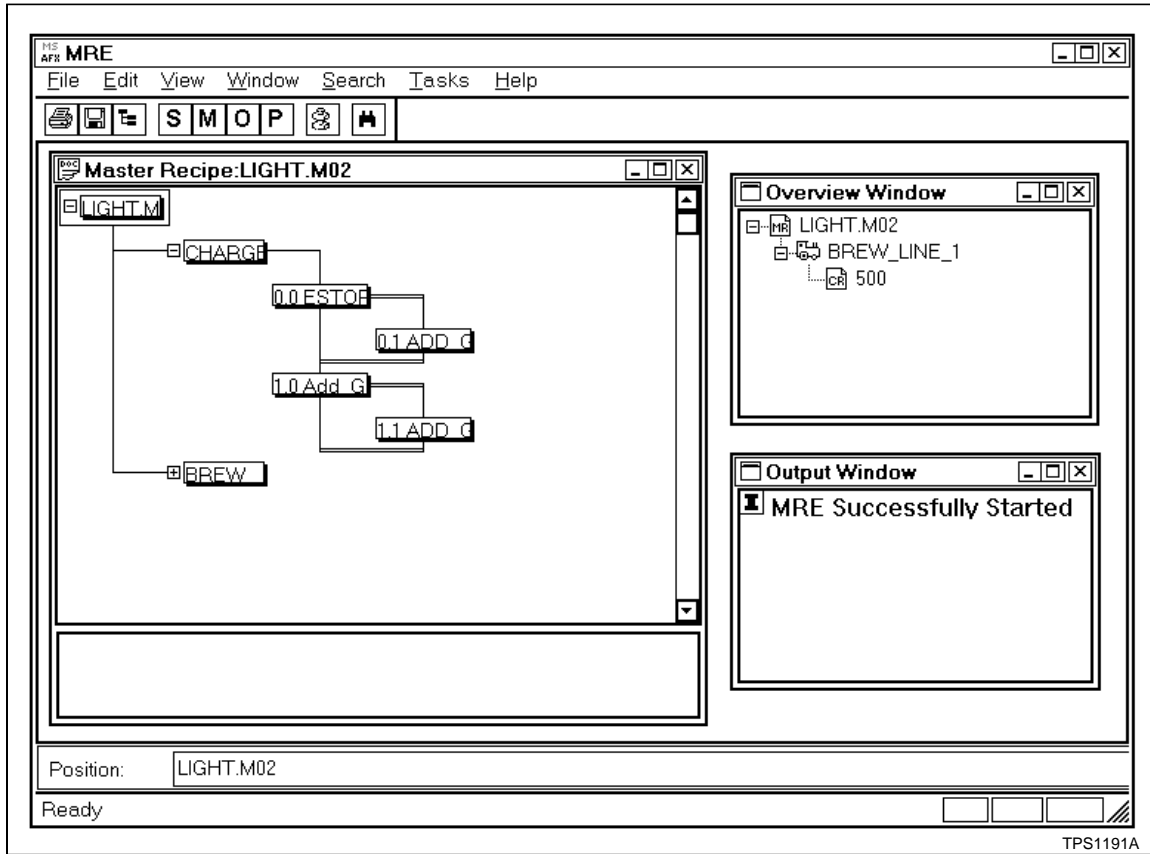


Figure 8-3. Master Recipe Editor Window (Edited) Example

- Checking unit procedures for errors.
- Creating control recipes.
- Renaming segments, operations, and trains.
- Deleting master recipes, segments, operations, and phases.
- Printing master recipes and unit procedures.
- Changing master recipe write protection status.
- Contract or expand master recipe, segment, and unit procedure structures.
- Searching for segments, operations, phases, and arguments.
- Refreshing the RAM memory image of the class library.

Write Protect

When created, a master recipe can be edited by any user with the appropriate access privileges. At any time, any user can change the master recipe to read only status and become the owner of it. No user can edit the master recipe until the owner changes the write protect status to the default state. The name of the user who enabled the read only state will be displayed in the properties dialog box for as long as the read only state is enabled.

To change the write protect status from the master recipe editor:

1. In the **master recipe window**, right click on the master recipe name.
2. Select *Write Protect* from the pop-up menu.

Contract or Expand

To contract or expand the master recipe, segment, or unit procedure structure (like a directory listing) from the master recipe editor:

1. In the **master recipe window**, right click on the master recipe, segment, or unit procedure name.

NOTE: Only one unit procedure can be expanded at a time.

2. Select *Contract/Expand* from the pop-up menu.

Insert Unit Procedure

To insert a new unit procedure into the current master recipe from the menu bar of the master recipe editor:

1. Verify the **master recipe window** is active.
2. Select *Insert Unit Procedure* from the Edit menu.
3. Enter the name of the inserted unit procedure under the unit procedure tab of the add unit procedure dialog box.

Unit procedure names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of eight characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Specify the class to be associated with this unit procedure from the list provided.
5. Specify the placement of the unit procedure by selecting a unit procedure and a modifier (before or after).
6. Enter any comments under the comment tab.

7. Enter description information under the description tab.
8. Click on *OK*.

To insert a new unit procedure into the current master recipe from the **master recipe window** of the master recipe editor:

1. Right click on the name of the master recipe (one segment used) or segment (multiple segments used) that will receive the inserted unit procedure.
2. Select *Insert Unit Procedure* from the pop-up menu.
3. Enter the name of the inserted unit procedure under the unit procedure tab of the add unit procedure dialog box.

Unit procedure names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of eight characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Specify the class to be associated with this unit procedure from the list provided.
5. Specify the placement of the unit procedure by selecting a unit procedure and a modifier (before or after).
6. Enter any comments under the comment tab.
7. Enter description information under the description tab.
8. Click on *OK*.

Export Unit Procedure

To send a copy of a unit procedure to an export subdirectory of the class directory from the master recipe editor:

1. In the **master recipe window**, expand the unit procedure structure.
2. Right click on the unit procedure name
3. Select *Export...* from the pop-up menu.
4. Enter the name of the unit procedure as it will appear in the export directory in the export dialog box.

Unit procedure names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of eight characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

5. Click on *OK*.

- Import Unit Procedure** To import a copy of a unit procedure from the export directory, change the name to match the current unit procedure, and delete the current unit procedure from the master recipe editor:
1. In the **master recipe window**, expand the unit procedure structure of the unit procedure that will be replaced by the imported unit procedure.
 2. Right click on the name of the unit procedure.
 3. Select *Import...* from the pop-up menu.
 4. Select the name of the unit procedure to be imported in the import dialog box.
 5. Click on *OK*.
- Print Unit Procedure** To print a unit procedure from the **master recipe editor window**:
1. Right click on the unit procedure name.
 2. Select *Print...* from the pop-up menu.
- The unit procedure print options dialog box will appear.
3. Under the unit procedures tab, specify if the unit procedure, comments, and unit parameters should be printed. When printing unit parameters, specify if default values, recipe values, and types are to be printed.
 4. Specify how many spaces to indent the printout of unit procedures, comments, and unit parameters.
 5. Under the operations tab, specify if the operations and operation comments are to be printed.
 6. Specify how many spaces to indent the printout of operations and operation comments.
 7. Under the phases tab, specify if phases and phase comments are to be printed.
 8. Specify how many spaces to indent the printout of phases and phase comments.
 9. Under the arguments tab, specify if argument default values, recipe values, and types are also to be printed.
 10. Specify how many spaces to indent the printout of arguments, default values, recipe values, and type.

11. Under the print profiles tab, specify if the print profile just configured is to be saved or an existing print profile to be loaded. When saving a new print profile, enter a print profile name. When loading an existing print profile, select the print profile name from the list.

12. Under the segments tab, specify if segments are to be printed.

13. Specify how many spaces to indent the printout of the segments.

14. Click on *OK*.

15. Specify the printer, print range, and copies settings in the standard Windows print dialog box.

16. Click on *OK*.

17. In the printing stats dialog box, click on *OK* to confirm the number of bytes sent to the printer and total page count.

Insert Operation

To insert a new operation into the current unit procedure from the menu bar of the ***master recipe editor window***:

1. Verify the unit procedure receiving the operation is expanded.
2. Select *Insert Operation* from the Edit menu.
3. Enter the name of the inserted operation under the operation tab of the insert operation dialog box.

Operation names are forced into upper case, are not case sensitive, and can not begin with an underscore. A maximum of 32 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Specify the placement of the operation by selecting an operation and a modifier (before or after).

NOTE: Operations can not be inserted before the first operation (operation 0) of a unit procedure.

5. Enter any comments under the comment tab.
6. Click on *OK*.

To insert a new operation into the current unit procedure from the ***master recipe window*** of the master recipe editor:

1. Verify the unit procedure receiving the operation is expanded.

2. Right click on an existing operation name or the name of the unit procedure that will receive the inserted operation.
3. Select *Insert Operation...* from the pop-up menu.
4. Enter the name of the inserted operation under the operations tab of the insert operation dialog box.

Operation names are forced into upper case, are not case sensitive, and can not begin with an underscore. A maximum of 32 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

5. Specify the placement of the operation by selecting an operation and a modifier (before or after).

NOTE: Operations can not be inserted before the first operation (operation 0) of a unit procedure.

6. Enter any comments under the comment tab.
7. Click on *OK*.

Export Operation

To send a copy of an operation to an export subdirectory of the class directory from the master recipe editor:

1. In the ***master recipe window***, right click on the operation name.
2. Select *Export...* from the pop-up menu.
3. Enter the name of the operation as it will appear in the export directory in the export dialog box.

Operation names are forced into upper case, are not case sensitive, and can not begin with an underscore. A maximum of 32 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Click on *OK*.

Import Operation

To import a copy of an operation from the export directory and add it to the current unit procedure from the master recipe editor:

1. Verify the unit procedure receiving the operation is expanded.
2. In the ***master recipe window***, right click on an existing operation name or the name of the unit procedure that will receive the imported operation.
3. Select *Import Operation...* from the pop-up menu.

4. Select the name of the operation to be imported in the import dialog box.
5. Specify the placement of the operation by selecting an operation and a modifier (before or after).

NOTE: Operations can not be inserted before the first operation (operation 0) of a unit procedure.

6. Click on *OK*.

Rename Operation

To give an operation a new name from the master recipe editor:

1. In the ***master recipe window***, right click on the name of the operation receiving the new name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new name in the rename operation dialog box.

Operation names are forced into upper case, are not case sensitive, and can not begin with an underscore. A maximum of 32 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Click on *OK*.

Add Phase

To insert a new phase into the current operation from the menu bar of the master recipe editor:

1. Verify the ***master recipe window*** is the active window.
2. Select *Add Phase* from the Edit menu.
3. Select the name of the inserted phase from the list provided under the phase tab of the add phase dialog box.
4. Specify the operation that will contain the phase.
5. Under the superphase tab, select the operation in which the phase will terminate.

NOTE: Phases within the first operation (operation 0) of a unit procedure must terminate in the first operation.

6. Enter any comments under the comment tab.
7. Click on *OK*.

To add a new phase to an operation from the **master recipe window** of the master recipe editor:

1. Right click on an existing phase or the name of the operation that will receive the new phase.
2. Select *Add Phase...* from the pop-up menu.
3. Select the name of the inserted phase from the supplied list under the phase tab of the add phase dialog box.
4. Specify the operation that will contain the phase.
5. Under the superphase tab, select the operation in which the phase will terminate.

NOTE: Phases within the first operation (operation 0) of a unit procedure must terminate in the first operation.

6. Enter any comments under the comment tab.
7. Click on *OK*.

Export Phase

To send a copy of a phase to an export subdirectory of the class directory from the master recipe editor:

1. In the **master recipe window**, right click on the phase name.
2. Select *Export...* from the pop-up menu.
3. Enter the name of the phase as it will appear in the export directory in the export dialog box.
4. Click on *OK*.

Import Phase

To import a copy of a phase from the export directory and add it to the current operation from the master recipe editor:

1. In the **master recipe window**, right click on an existing phase name or the name of the operation that will receive the imported phase.
2. Select *Import Phase...* from the pop-up menu.
3. Select the name of the phase to be imported in the import dialog box.

4. Under the superphase tab, select the operation in which the phase will terminate.

NOTE: Phases within the first operation (operation 0) of a unit procedure must terminate in the first operation.

5. Click on *OK*.

Terminate To change the operation in which a phase will terminate from the master recipe editor:

1. In the **master recipe window**, right click on the phase name.
2. Select *Terminate...* from the pop-up menu.
3. Select the operation in which the phase will terminate in the terminate phase dialog box.

NOTE: Phases within the first operation (operation 0) of a unit procedure must terminate in the first operation.

4. Click on *OK*.

Insert Segment To insert a new segment into the current master recipe from the menu bar of the master recipe editor:

1. Verify the **master recipe window** is the active window.
2. Select *Insert Segment* from the Edit menu.
3. Enter the name of the inserted segment in the insert segment dialog box.

Segment names are forced into upper case, are not case sensitive, and can not begin with an underscore. A maximum of 32 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Specify the placement of the segment by selecting a segment and a modifier (before or after).

5. Click on *OK*.

To insert a new segment into the current master recipe from the **master recipe window** of the master recipe editor:

1. Right click on an existing segment name or the name of the master recipe that will receive the inserted segment.
2. Select *Insert Segment...* from the pop-up menu.
3. Enter the name of the inserted segment under the segment tab of the insert segment dialog box.

Segment names are forced into upper case, are not case sensitive, and can not begin with an underscore. A maximum of 32 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Specify the placement of the segment by selecting a segment and a modifier (before or after).
5. Enter any comments under the comment tab.
6. Click on *OK*.

Rename Segment

To give a segment a new name from the master recipe editor:

1. In the ***master recipe window***, right click on the name of the segment receiving the new name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new name in the rename segment dialog box.

Segment names are forced into upper case, are not case sensitive, and can not begin with an underscore. A maximum of 32 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

4. Click on *OK*.

Add Train

To add a new train to a master recipe from the ***master recipe window*** of the master recipe editor:

1. Verify the ***master recipe window*** is the active window.
2. Right click on the name of the master recipe that will contain the train.
3. Select *Add Train...* from the pop-up menu.
4. Enter a name under the train tab of the add train dialog box.

Train names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

5. Enter any comments under the comment tab.
6. Enter description information under the description tab.
7. Click on *OK*.
8. Select the unit to be tied to the train in the edit train dialog box.

9. Click on *OK*.

To add a new train to a master recipe from the **overview window** of the master recipe editor:

1. Verify the **overview window** is the active window.
2. Right click on the master recipe name.
3. Select *Add Train* from the pop-up menu.
4. Enter a name under the train tab of the add train dialog box.

Train names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores.

5. Enter any comments under the comment tab.
6. Enter description information under the description tab.
7. Click on *OK*.
8. Select the unit to be tied to the train in the edit train dialog box.
9. Click on *OK*.

Edit Train To edit a train from the master recipe editor:

1. Verify that the **overview window** is the active window.
2. Right click on the train name.
3. Select *Edit Train* from the pop-up menu.
4. Select the unit to be tied to the train in the edit train dialog box.
5. Click on *OK*.

Copy Train To make a copy of a train and give it a new name under the same master recipe from the master recipe editor:

1. Verify the **overview window** is the active window.
2. Right click on the train name.
3. Select *Copy...* from the pop-up menu.

4. Enter the name of the copy in the copy train dialog box.
5. Click on *OK*.

To make a copy of a train and give it a new name under the same master recipe from the ***master recipes window***:

1. Right click on the train name.
2. Select *Copy...* from the pop-up menu.
3. Enter the name in the copy train dialog box.
4. Click on *OK*.

Delete Train To delete a train from the master recipe editor:

1. In the ***overview window***, right click on the train name.
2. Select *Delete* from the pop-up menu.
3. Click on *Yes* in the delete dialog box.

To delete a train from the ***master recipes window***:

1. Right click on the train name.
2. Select *Delete* from the pop-up menu.
3. Click on *Yes* in the delete dialog box.

Rename Train To give a train a new name from the master recipe editor:

1. In the ***overview window***, right click on the current train name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new name in the rename train dialog box.
4. Click on *OK*.

To give a train a new name from the ***master recipes window***:

1. Right click on the current train name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new name in the rename train dialog box.
4. Click on *OK*.

Delete To delete a segment, unit procedure, operation, or phase from the master recipe editor:

1. In the **master recipe window**, right click on the segment, unit procedure, operation, or phase name.

NOTES:

1. When deleting an operation, delete or change the termination operation of any superphase that passes through the operation.
 2. The first operation (operation 0) of a unit procedure can not be deleted.
2. Select *Delete* from the pop-up menu.
 3. Click on *Yes* in the confirm delete dialog box.

Search To search for a specific segment, operation, phase, or argument from the master recipe editor:

1. Verify the **master recipe window** is the active window.
2. Select *Segment, Unit Procedure, Operation, or Phase* from the Search menu.
3. In the search dialog box, specify where to search and the name of the segment, unit procedure, operation, or phase.
4. Click on *OK*.

A box will appear around the found operation, phase, or argument.

5. Click on *OK* to exit the search dialog box.

Refresh Class Library To update the RAM memory image of the class library, select *Refresh Class Library* from the Tasks menu of the **master recipe editor window**.

Exiting Master Recipe Editor Save any edited master recipes and select *Exit* from the File menu. If edited master recipes are not saved before selecting *Exit*, the exit dialog box appears. In this dialog box, confirm the creation of a new revision, creation of a new master recipe, overwriting the current revision, or not saving. When creating a new master recipe, enter the new name.

Copy

To make a copy of a master recipe in the same project directory and give it a new name or revision number:

1. In the **master recipes window**, right click on the master recipe name.

2. Select *Copy...* from the pop-up menu.
3. In the copy master recipe dialog box, specify the creation of a new revision or an entirely new master recipe. Enter the new master recipe name when creating a new master recipe.
4. Click on *OK*.

To copy all or part of a master recipe from the current project directory to another directory:

1. In the ***master recipes window***, right click on the master recipe name.
2. Select *Copy Out...* from the pop-up menu.
3. Select the destination directory in the copy master recipe out dialog box.
4. Specify if the control recipes are to be copied also.
5. Click on *OK*.

To copy a master recipe from another directory to the current project directory:

1. In the ***master recipes window***, right click on the destination project name.
2. Select *Copy Master Recipe In...* from the pop-up menu.
3. In the copy master recipe in dialog box, select the project directory and name of the master to be copied.
4. Enter the new name for the master recipe.
5. Click on *OK*.

Rename

To give a master recipe a new name:

1. In the ***master recipes window***, right click on the master recipe name.
2. Select *Rename...* from the pop-up menu.
3. Enter the new name of the master recipe in the rename master recipe dialog box.
4. Click on *OK*.

Print

To print all or part of a master recipe from the **master recipe window** of the master recipe editor:

1. Verify the **master recipe window** is the active window.
2. Select *Print...* from the File menu.

The master recipe print options dialog box will appear.

3. Under the segments tab, specify if segments are to be printed.
4. Specify how many spaces to indent the printout of segments.
5. Under the unit procedures tab, specify if the unit procedure, comments, and unit parameters should be printed. When printing unit parameters, specify if default values, recipe values, and types are to be printed.
6. Specify how many spaces to indent the printout of unit procedures, comments, and unit parameters.
7. Under the operations tab, specify if the operations and operation comments are to be printed.
8. Specify how many spaces to indent the printout of operations and operation comments.
9. Under the phases tab, specify if phases and phase comments are to be printed.
10. Specify how many spaces to indent the printout of phases and phase comments.
11. Under the arguments tab, specify if argument default values, recipe values, and types are to be printed.
12. Specify how many spaces to indent the printout of arguments, default values, recipe values, and type.
13. Under the print profiles tab, specify if the print profile just configured is to be saved or an existing print profile to be loaded. When saving a new print profile, enter a name print profile name. When loading an existing print profile, select the print profile name from the list.
14. Click on *OK*.
15. Specify the printer, print range, and copies settings in the standard Windows print dialog box.

16. Click on *OK*.

17. In the print stats dialog box, click on *OK* to confirm the number of bytes sent to the printer and total page count.

To print all or part of a master recipe from the ***overview window*** of the master recipe editor:

1. Verify the ***overview window*** is the active window.
2. Right click on the master recipe name.
3. Select *Print...* from the pop-up menu.

The master recipe print options dialog box will appear.

4. Under the segments tab, specify if segments are to be printed.
5. Specify how many spaces to indent the printout of segments.
6. Under the unit procedures tab, specify if the unit procedure, comments, and unit parameters should be printed. When printing unit parameters, specify if default values, recipe values, and types are to be printed.
7. Specify how many spaces to indent the printout of unit procedures, comments, and unit parameters.
8. Under the operations tab, specify if the operations and operation comments are to be printed.
9. Specify how many spaces to indent the printout of operations and operation comments.
10. Under the phases tab, specify if phases and phase comments are to be printed.
11. Specify how many spaces to indent the printout of phases and phase comments.
12. Under the arguments tab, specify if argument default values, recipe values, and types are to be printed.
13. Specify how many spaces to indent the printout of arguments, default values, recipe values, and type.
14. Under the print profiles tab, specify if the print profile just configured is to be saved or an existing print profile to be loaded. When saving a new print profile, enter a print profile name. When loading an existing print profile, select the print profile name from the list.

15. Click on *OK*.
16. Specify the printer, print range, and copies settings in the standard Windows print dialog box.
17. Click on *OK*.
18. In the print stats dialog box, click on *OK* to confirm the number of bytes sent to the printer and total page count.

Delete

To delete a master recipe from a project:

1. In the **master recipes window**, right click on the master recipe name.
2. Select *Delete* from the pop-up menu.
3. In the delete dialog box, specify if all revisions of the master recipe should be deleted or just this revision.
4. Click on *OK*.

CONTROL RECIPES

After the master recipe is edited, a control recipe should be created by compiling the master recipe and train. It is possible to edit, copy, rename, and delete a control recipe also.

Add

To generate a control recipe from the menu bar of the master recipe editor:

1. Verify the **master recipe window** is the active window.
2. Select *Add Control Recipe* from the File menu.
3. Enter a name (for new control recipes) or select one from the list provided (for overwriting existing control recipes) under the control recipe tab of the add control recipe dialog box.

Control recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric control recipe names are one through 65,536.

4. Select the train to be used when compiling.
5. Enter any comments under the comment tab.

6. Enter description information under the description tab.
7. Click on *OK*.

To generate a control recipe from the **master recipe window** of the master recipe editor:

1. In the **master recipe window**, right click on the name of the master recipe that will be used to create the control recipe.
2. Select *Add Control Recipe...* from the pop-up menu.
3. Enter a name (for new control recipes) or select one from the list provided (for overwriting existing control recipes) under the control recipe tab of the add control recipe dialog box.

Control recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric control recipe names are one through 65,536.

4. Select the train to be used when compiling.
5. Enter any comments under the comment tab.
6. Enter description information under the description tab.
7. Click on *OK*.

To generate a control recipe and add it to the master recipe from the **overview window** of the master recipe editor:

1. In the **overview window**, right click on the name of the train or master recipe that will be used to create the control recipe.
2. Select *Add Control Recipe...* from the pop-up menu.
3. Enter a name (for new control recipes) or select one from the list provided (for overwriting existing control recipes) under the control recipe tab of the add control recipe dialog box.

Control recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric control recipe names are one through 65,536.

4. Select the train to be used when compiling.
5. Enter any comments under the comment tab.

6. Enter description information under the description tab.
7. Click on *OK*.

Edit

To edit a control recipe from the **master recipes window**:

1. Right click on the control recipe name.
2. Select *Edit...* from the pop-up menu.

The master recipe editor window containing the control recipe, overview, and output windows will appear. The control recipe window shows the master recipe structure on top and a listing of the arguments associated with selected control recipe on the bottom. The overview window displays the structure of the master recipe after it has been compiled to create a control recipe. The output window displays general output messages.

3. Edit the control recipe as required.
4. Select *Save* from the File menu.
5. Select *Exit* from the File menu.

To edit a control recipe from the master recipe editor:

1. Verify that the **overview window** is the active menu.
2. Right click on the control recipe name.
3. Select *Edit...* from the pop-up menu.

The master recipe editor window containing the control recipe, overview, and output windows will appear.

4. Edit the control recipe as required.
5. Select *Save* from the File menu.
6. Select *Exit* from the File menu.

The only editing function available in the control recipe is the editing of unit procedure and phase arguments. To edit arguments:

1. In the **control recipe window** of the master recipe editor, click the unit procedure or phase containing the argument.
2. Double click on the argument.
3. Edit the argument as necessary in the edit dialog box.

4. Click on *OK*.

Copy

To make a copy of a control recipe in the same project directory and give it a new name from the **master recipes window**:

1. Right click on the control recipe name.
2. Select *Copy...* from the pop-up menu.
3. Enter the new control recipe name in the copy control recipe dialog box.

Control recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric control recipe names are one through 65,536.

4. Click on *OK*.

To make a copy of a control recipe in the same project directory and give it a new name from the master recipe editor:

1. Verify the **overview window** is the active window.
2. Right click on the control recipe name.
3. Select *Copy...* from the pop-up menu.
4. Enter the new control recipe name in the copy control recipe dialog box.

Control recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric control recipe names are one through 65,536.

5. Click on *OK*.

Rename

To rename a control recipe from the **master recipes window**:

1. Verify the control recipe to be renamed is not being edited.
2. Right click on the control recipe name.

3. Select *Rename...* from the pop-up menu.
4. Enter the new control recipe name in the rename control recipe dialog box.

Control recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric control recipe names are one through 65,536.

5. Click on *OK*.

To rename a control recipe from the master recipe editor:

1. Verify the control recipe to be renamed is not being edited.
2. Verify the **overview window** is the active window.
3. Right click on the control recipe name.
4. Select *Rename...* from the pop-up menu.
5. Enter the new control recipe name in the rename control recipe dialog box.

Control recipe names can be alphabetic or numeric. Alphabetic unit names are forced into upper case, not case sensitive, and can not begin with an underscore. A maximum of 16 characters can be used. Valid characters include letters A through Z, numbers one through nine, and underscores. Valid numbers for numeric control recipe names are one through 65,536.

6. Click on *OK*.

Delete

To delete a control recipe from the **master recipes window**:

1. Verify the control recipe to be deleted is not being edited.
2. Right click on the control recipe name.
3. Select *Delete...* from the pop-up menu.
4. Click on *OK* in the delete control recipe dialog box.

To delete a control recipe from the master recipe editor:

1. Verify the **overview window** is the active window.
2. Right click on the control recipe name.

3. Select *Delete...* from the pop-up menu.
4. Click on *OK* in the delete control recipe dialog box.

To delete one or more control recipes associated with a train from the master recipe editor:

1. Verify the control recipe to be deleted is not being edited.
2. Verify the **overview window** is the active window.
3. Right click on the train name.
4. Select *Delete Selected...* from the pop-up menu.
5. Select the control recipes to be deleted in the delete control recipe dialog box.
6. Click on *OK*.

PROPERTIES

The master recipe, segment, operation, phase, master recipe editor, control recipe, and train property display are available from the master recipes and master recipe editor windows.

Master Recipe

The properties of a master recipe include:

- Type.
- Location and name.
- Date the master recipe was created.
- Size (in bytes).
- Number of trains and control recipes associated with this master recipe.
- Date the master recipe was last compiled.
- Type of master recipe ID used.
- Write protection status.
- Compiled status.
- Notification if the master recipe is currently opened.

- Comment.
- Description.

To view or edit the properties of a master recipe from the **master recipes window**:

1. Right click on the master recipe name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit the properties of a master recipe from the **overview window** of the master recipe editor:

1. Right click on the master recipe name.
2. Select *Properties...* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit the properties of a master recipe from the **master recipe window** of the master recipe editor:

1. Right click on the master recipe name.
2. Select *Properties...* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Segment

The properties of a segment include:

- Type.
- Location and name.
- Date the segment was created.
- Size (in bytes).
- Date the segment was last compiled.
- Type of segment ID used.
- Notification if the segment is currently opened.

To view or edit the properties of a segment from the **master recipe window** of the master recipe editor:

1. Right click on the segment name.
2. Select *Properties...* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Unit Procedure

The properties of a unit procedure include:

- Type.
- Location and name.
- Date the unit procedure was created.
- Size (in bytes).
- Date the unit procedure was last compiled.
- Type of unit procedure ID used.
- Notification if the unit procedure is currently opened.
- Name of the class associated with this unit procedure.
- Date the batch program for the class was last compiled.
- Name of the class library containing the class used by this unit procedure.
- Unit relative status.
- Firmware revision number of the MFP module containing the unit recipe.

To view or edit the properties of a unit procedure from the **master recipe window** of the master recipe editor:

1. Right click on the unit procedure name.
2. Select *Properties...* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Operation

The properties of an operation include:

- Type.
- Location.
- Number of phases contained in this operation
- Comment.

To view or edit the properties of an operation from the **master recipe window** of the master recipe editor:

1. Right click on the operation name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Phase

The properties of a phase include:

- Type.
- Location.
- Number of arguments used.
- Comment.

To view or edit the properties of a phase from the **master recipe window** of the master recipe editor:

1. Right click on the phase name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Master Recipe Editor

The properties of the master recipe editor include:

- Color of all objects used.
- Choice of displaying property name, number, or both.
- Option to change font size when object size changes.
- Width and height of objects.

To view or edit the properties of the master recipe editor:

1. Select *Properties* from the Edit menu or right click on any empty space in the **master recipe window**.
2. View or edit the desired properties in the properties dialog box.
3. Click on *OK*.

Control Recipe

The properties of a control recipe include:

- Type.
- Name of the master recipe and train that contain this control recipe.
- Date first compiled.
- Size (in bytes).
- Number of segments and unit recipes contained in this control recipe.
- ID type (numeric or alphanumeric).
- Comment.
- Description.

To view or edit the properties of a control recipe from the **master recipes window**:

1. Right click on the control recipe name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit the properties of a control recipe from the master recipe editor:

1. In the **overview window**, right click on the control recipe name.
2. Select *Properties* from the pop-up menu.

3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

Train

The properties of a train include:

- Type.
- Name of the master recipe that contains this train.
- Number of control recipes contained.
- Validity.
- Comment.
- Description.

To view or edit the properties of a train from the **master recipes window**:

1. Right click on the train name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view or edit the properties of a train from the master recipe editor:

1. In the **overview window**, right click on the train name.
2. Select *Properties* from the pop-up menu.
3. View or edit (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

SECTION 9 - INFORMATION DOWNLOAD

INTRODUCTION

After a creating the necessary master recipes, unit procedures, or UDF programs, they must be downloaded into the MFP module. This section describes how to download the required files to and delete them from an MFP module. How to compare the files in a module to the same files on the hard disk drive of a work station is also described in this section.

FORMAT

The MFP module must be formatted before any files are downloaded. To format the module from the **batch data manager window**:

1. Select *Downloader* from the Applications menu.

The batch download utility window will appear.

2. Select *Format* from the Module menu.

NOTE: Formatting an MFP module erases the stored function block configuration. Replace the configuration after the format operation is complete.

The MFP format dialog box will appear.

3. Under the format tab, select the type of format to be performed. If mini format (only deletes existing programs and function code logic) is selected, click on *Format* to complete the format operation. If standard or expanded format is selected, proceed to the next step.

4. Specify the type of MFP module being formatted, number of batch files desired, address of the first batch file (base batch file), and the amount of NVRAM memory assigned to function code logic. If standard format was selected, click on *Format* to complete the format operation. If expanded format was selected, proceed to the next step.

5. Specify the number of other (non-batch) files desired and the number and size of the MBFIO and check point buffers desired.

6. Click on *Format*.

Refer to **Communication Settings** for more information about setting communication standards under the communications tab. Refer to **Change Module Mode** for more information about

selecting MFP modules and changing MFP module modes under the MFP tab.

DOWNLOAD

The files that must be downloaded to the MFP module for a master recipe to execute are:

- Batch 90 program file for every class used by the unit procedures that comprise the master recipe.
- Units (unit definition files) used by the unit procedures that comprise the master recipe.
- Control recipe generated by compiling unit procedures and units.

The files that must be downloaded to the MFP module for a unit procedure to execute are:

- Batch 90 program file for the class used by the unit procedure.
- Units (unit definition file) used by the unit procedure.
- Unit recipes generated by compiling the unit procedure and units.

Only one file, the program file, must be downloaded to the MFP module for a UDF program to execute.

Batch 90 Program Files

To download a batch program that is associated with a class to an MFP module:

1. In the **class library window**, right click on the class associated with the batch program.
2. Select *Download...* from the pop-up menu.
3. Select the unit or the address (loop, PCU, and module) of the MFP module that will receive the batch program from the B90 program download dialog box.
4. Click on *OK*.

UDF Program Files

To download a UDF program that is associated with a project to an MFP module:

1. In the **UDF window**, right click on the program.

2. Select *Download* from the pop-up menu.
3. Specify the address (loop, PCU, and module) of the module that will receive the UDF program from the download UDF program dialog box.
4. Click on *OK*.

Units

To download one or more units associated with a class to an MFP module:

1. In the **class library window**, right click on the class name.
2. Select *Download Selected...* from the pop-up menu.
3. Select the unit or units to be downloaded from the download units dialog box.
4. Click on *OK*.

To download a unit that is associated with a class to an MFP module:

1. In the **class library window**, right click on the unit name.
2. Select *Download* from the pop-up menu.

Unit Recipes

To download a unit recipe to an MFP module:

1. In the **unit procedures window**, right click on the unit recipe name.
2. Select *Download* from the pop-up menu.

To download one or more unit recipes to an MFP module:

1. In the **unit procedures window**, right click on the unit name.
2. Select *Download Selected...* from the pop-up menu.
3. Select the unit recipe or unit recipes in the download unit recipes dialog box.
4. Click on *OK*.

Control Recipes

To download a control recipe to an MFP module:

1. In the **master recipes window**, right click on the control recipe name.
2. Select *Download...* from the pop-up menu.
3. Specify the parts of the control recipe (entire control recipe, segment, unit, unit recipe) to download in the download control recipe to MFP dialog box.
4. Click on *OK*.

VERIFY

It is possible to compare files residing on a work station hard disk drive to files of the same name residing in an MFP module. This capability can be used to insure the module contains the most recent files.

Batch 90 Program Files

To verify a Batch 90 program in an MFP module is identical to the associated Batch 90 program identified by the class definition on the work station:

1. In the **class library window**, right click on the class associated with the batch program.
2. Select *Verify In MFP* from the pop-up menu.
3. Select the unit or address (loop, PCU, and module) associated with the batch program from the B90 program verify dialog box.
4. Click on *OK*.
5. Click on *OK* in the BDU dialog box.

UDF Program Files

To verify the UDF program that is associated with a project that resides in an MFP module is identical to the UDF program that is associated with the same project that resides in the work station:

1. In the **UDF window**, right click on the program name.
2. Select *Verify In MFP* from the pop-up menu.

3. Enter or select from a list the MFP module address from the verify UDF program in MFP verify dialog box.
4. Click on *OK*.
5. Click on *OK* in the BDU dialog box.

Units

To verify the unit that resides in an MFP module is identical to the unit that resides in the work station:

1. In the **class library window**, right click on the unit name.
2. Select *Verify In MFP* from the pop-up menu.
3. Click on *OK* in the BDU dialog box.

Unit Recipes

To verify the unit recipe that resides in an MFP module is identical to the unit recipe that resides in the work station:

1. In the **unit procedures window**, right click on the unit recipe name.
2. Select *Verify In MFP* from the pop-up menu.
3. Click on *OK* in the BDU dialog box.

Control Recipes

To verify the control recipe that resides in an MFP module is identical to the control recipe that resides in the work station:

1. In the **master recipes window**, right click on the control recipe name.
2. Select *Verify In MFP* from the pop-up menu.
3. Specify the parts of the control recipe (entire control recipe, segment, unit, unit recipe) to verify in the verify control recipe dialog box.
4. Click on *OK*.

DELETE

At times it may be necessary to delete certain files from an MFP module. The batch data manager provides the capability of deleting individual program, unit, unit recipe, and control recipe files.

Batch 90 Program Files

To delete the class batch program associated with a unit from an MFP module:

1. In the **class library window**, right click on the class associated with the batch program.
2. Select *Delete from MFP* from the pop-up menu.
3. In the B90 program delete dialog box, select the unit or address (loop, PCU, and module) associated with class batch program that will be deleted.
4. Click on *OK*.

UDF Program Files

To delete a UDF program that is associated with a project from an MFP module:

1. In the **UDF window**, right click on the program name.
2. Select *Delete from MFP* from the pop-up menu.
3. Enter or select from a list the MFP module address from the delete UDF program from MFP dialog box.
4. Click on *OK*.

Units

To delete a unit that is associated with a class from an MFP module:

1. In the **class library window**, right click on the unit name.
2. Select *Delete from MFP* from the pop-up menu.

Unit Recipes

To delete a unit recipe from an MFP module:

1. In the **unit procedures window**, right click on the unit recipe name.
2. Select *Delete from MFP* from the pop-up menu.

Control Recipes

To delete a control recipe from an MFP module:

1. In the **master recipes window**, right click on the control recipe name.
2. Select *Delete from MFP* from the pop-up menu.
3. Specify the parts of the control recipe (entire control recipe, segment, unit, and unit recipe) to delete from the MFP module in the delete control recipe from MFP dialog box.
4. Click on *OK*.

DOWNLOAD UTILITY

The batch download utility allows the formatting, changing modes of, and reading file directories from MFP modules. Searching for MFP modules and the changing of MFP communication parameters are also supported. The utility also lists (in the window) any messages generated by the utility. This list or parts of it can be printed, sorted, and deleted. Figure 9-1 shows an example batch download utility window.

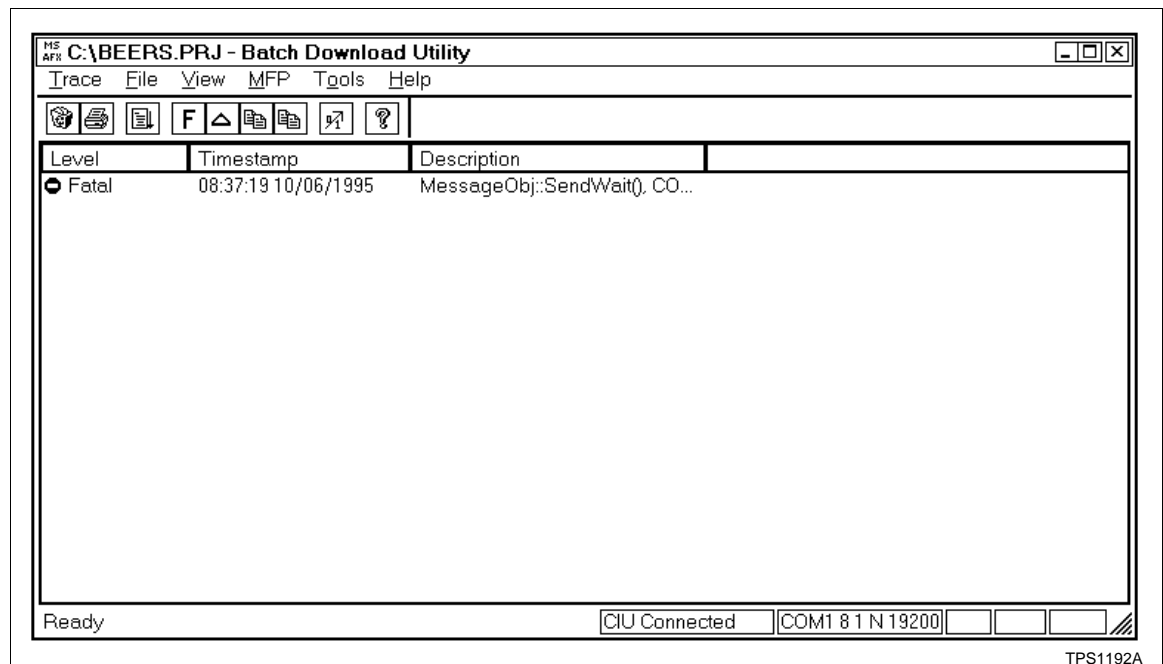


Figure 9-1. Batch Downloader Utility Window Example

Change Module Mode

To change the mode of a module from the **batch download utility window**:

1. Select *Change Mode* from the Module menu.

The MFP mode change dialog box will appear.

2. Under the MFP tab, enter the loop, PCU, and module address of the MFP module whose mode is to be changed.
3. Confirm the current MFP module mode by clicking on *Inspect* and reading the current mode field of the tab.
4. Click on *Configure*, or *Execute* to specify the new mode or click on *Reset* to initialize and restart the module.

Refer to **Communication Settings** for more information about setting communication parameters under the communications tab.

Read Module Directories

To view a directory of all files in an MFP module from the **batch download utility window**:

1. Select *File Directory* from the Module menu.

The MFP file directory dialog box will appear. The MFP address, type, and mode of the last MFP module downloaded to are displayed on the MFP directory tab. Also displayed are the number of used files, total memory space available to programs, and the amount of unused NVRAM memory. The files are displayed on the bottom of the tab and can be sorted by name, size, and attributes. To sort the files click on *Name*, *Size*, or *Attributes*.

2. Click on *Read*.

Refer to **Communication Settings** for more information about setting the communication parameters under the communications tab. Refer to **Change Module Mode** for more information about selecting MFP modules and changing MFP module modes under the MFP tab.

To view a directory of the batch and UDF program files in an MFP module from the **batch download utility window**:

1. Select *Batch/UDF Directory* from the Module menu.

The batch/UDF directory dialog box will appear. The MFP address, type, and firmware revision of the last MFP module

downloaded to are displayed on the directory tab. Also displayed are the number of used and free files. The files are displayed on the bottom of the directory tab. These files can be sorted by type, program name, program ID, function block address for the file, file name, and (if expanded program option is selected) description, length, and time stamp. Sort the files by clicking on *Type*, *File*, *ID*, *Block*, *Name*, *Description*, *Length*, or *Timestamp*.

2. Click on *Read*.

Refer to **Communication Settings** for more information about setting the communication parameters under the communications tab. Refer to **Change Module Mode** for more information about selecting MFP modules and changing MFP module modes under the MFP tab. Refer to **Read Module Directories** for more information about viewing a directory under the MFP directory tab.

Search For Modules

To search for an MFP module from the **batch download utility window**:

1. Select *Search for MFPs* from the Tools menu.
2. Specify the loop, PCU, and module address of the module to be searched for in the search for MFP modules dialog box. A wild card character (?) can be used.
3. Click on *Search Now*.

The loop, PCU, and module address of the searched for module or modules will be displayed on the bottom of the dialog box. Also displayed will be the type, revision, and current mode of the module. The displayed module information can be sorted by clicking on *Loop*, *PCU*, *Module*, *Type*, *Revision*, and *Mode*.

Communication Settings

To view or edit the communication settings used between the network interface unit and the work station:

1. In the **batch download utility window**, select *Communications* from the Tools menu.
2. Click on *Auto Detect* to allow the batch data manager software to configure itself to be able to communicate with the network interface modules.

The communication port being used, baud rate, parity, stop bits, and data bits used by the interface unit are displayed on the NIU communications settings dialog box.

3. View or edit the parameters as required.
4. Click on *Connect* to communicate using the new parameters.
5. If satisfied with the new parameters, click on *Restart NIU* to save the new parameters and restart the interface unit.

Size

To set the maximum number of messages the download utility file will hold from the **batch download utility window**:

1. Select *Size* from the Trace menu.
2. Enter the maximum number of messages to save in the trace size dialog box.
3. Click on *OK*.

Delete

To delete all messages in the download utility from the **batch download utility window**, select *Delete all* from the Trace menu.

To delete all messages in the download utility from the **batch download utility window**:

1. Right click anywhere (except on the level of a message) in the window.
2. Select *Delete All* from the pop-up menu.

To delete one message from the **batch download utility window**:

1. Right click on the message to be deleted.
2. Select *Delete* from the pop-up menu.

To delete multiple messages from the **batch download utility window**:

1. Select the messages to be deleted.
2. Right click in the level column of one of the selected messages.
3. Select *Delete Selected* from the pop-up menu.

Print

To print all the messages saved by the download utility from the **batch download utility window**:

1. Select *Print* from the Trace menu or right click anywhere (except on the level of a message) in the window and select *Print All* from the pop-up menu.
2. Select the print options as necessary in the standard Windows print dialog box.
3. Click on *Open*.

To print one message from the **batch download utility window**:

1. Right click on the message to be printed.
2. Select *Print* from the pop-up menu.
3. Select the print options as necessary in the standard Windows print dialog box.
4. Click on *Open*.

To print multiple messages from the **batch download utility window**:

1. Select the messages to be printed.
2. Right click in the level column and on one of the selected messages.
3. Select *Print Selected* from the pop-up menu.
4. Select the print options as necessary in the standard Windows print dialog box.
5. Click on *Open*.

Properties

The properties of a message include:

- Date logged.
- Time logged.
- Message level (type).
- User and computer name.
- Description of the message.

To view the properties of a message from the **batch download utility window**:

1. Right click on the level (type) of the message.
2. Select *Properties* from the pop-up menu.
3. View the properties in the event detail dialog box.
4. Click on *Next* (view details of next newer message), *Previous* (view details of previous older message), or *Close*.

SECTION 10 - DYNAMIC DEBUGGER UTILITY

INTRODUCTION

The next step after creating and downloading a unit procedure, master recipe, or UDF program is to verify that it executes correctly. Use the control system to execute the logic and watch for any logic errors. If any errors are generated, debug the appropriate unit recipe, control recipe, or program.

The batch debugger window consists of two windows and a status bar across the bottom. Refer to Figure 10-1 for an example batch debugger window. One window displays the program code for the class and a watch box at the bottom. The watch box displays the values of variables put on watch. The second window (not present when debugging UDF programs) displays a graphical representation of the unit or control recipe and a box showing the parameters associated with the selected phase or unit procedure.

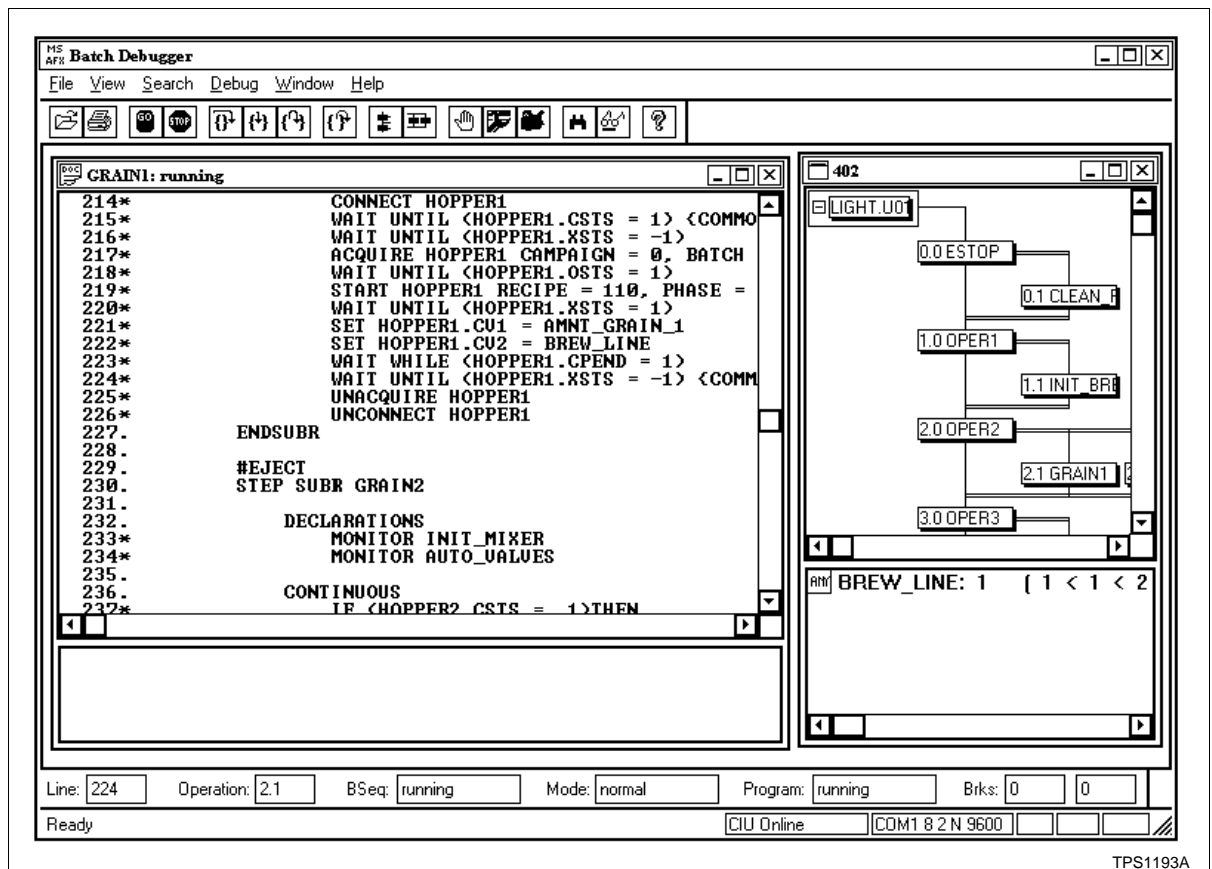


Figure 10-1. Batch Debugger Window Example

To debug a unit recipe:

1. In the **unit procedures window**, right click on the unit recipe name.
2. Select *Debug* from the pop-up menu.

The batch debugger window appears.

3. Find any runtime fault codes reported by the debugger (refer to Table 12-2) using the debugger capabilities discussed later in this section.
4. Select *Exit* from the File menu.
5. Edit the unit procedure to correct the errors.
6. Generate a unit recipe of the same name.
7. Download the unit recipe to the MFP module.
8. Execute the unit recipe and watch for any logic errors.
9. Repeat Steps 1 through 8 until no errors are found.

To debug a control recipe:

1. In the **master recipes window**, right click on the control recipe name.
2. Select *Debug...* from the pop-up menu.
3. Select the segment and unit procedure name associated with the control recipe from the debug dialog box.

The batch debugger window appears.

4. Find any errors using the debugger capabilities discussed later in this section.
5. Select *Exit* from the File menu.
6. Edit the necessary unit procedures to correct the errors.
7. Generate a control recipe of the same name.
8. Download the control recipe to the MFP module.
9. Execute the control recipe and watch for any logic errors.
10. Repeat Steps 1 through 9 until no errors are found.

To debug a UDF program:

1. In the **UDF window**, right click on the UDF program name.
2. Select *Debug...* from the pop-up menu.
3. Enter the loop, PCU, module, and UDF block address for the UDF program in the dialog box.

The batch debugger window appears.

4. Find any errors using the debugger capabilities discussed later in this section.
5. Edit the UDF program to correct the errors.
6. Download the UDF program to the MFP module.
7. Execute the UDF program and watch for any logic errors.
8. Repeat Steps 1 through 7 until no errors are found.
9. Select *Exit* from the File menu.

DEBUGGER CAPABILITIES

The batch debugger allows the execution of the program to be controlled. By executing the program in pieces or even one step at a time, progress can be closely monitored allowing the causes of errors to be more easily found. The value of program and built-in variables can be viewed. Program variables can be put on watch and monitored while the program executes. Program variables can be changed when program execution is stopped.

Stop Execution

To stop the execution of a program in an MFP module from the batch debugger:

1. Verify the **program window** is the active window.
2. Select *Stop* from the Debug menu.

Start Execution

To start the execution of a program in an MFP module from the batch debugger:

1. Verify the **program window** is the active window.

2. Select *Go* from the Debug menu.

Step Execution

It is possible to execute programs one statement at a time (after stopping the program) using the *Step Section*, *Step Level*, and *Step Same* functions under the Debug menu.

Step Section To execute the current statement and stop from the ***batch debugger window***, select *Step Section* from the Debug menu.

NOTE: If the statement is a monitor or function subroutine call, the entire subroutine is executed normally.

The next executable statement (regardless of section boundaries) will be highlighted.

Step Level To execute the first statement of a monitor or function subroutine from the ***batch debugger window***, select *Step Level* from the Debug menu.

If the next statement to be executed is a function or monitor subroutine, the step level function will cause the debugger to execute the next executable statement ***within*** the monitor or function subroutine. The step level function will not execute out of the subroutine. Execution of statements will stop after the last statement. From within a subroutine, the step level function executes exactly like the step same function.

Step Same To execute the current statement and stop at the next executable statement within the current program section (function calls, monitor calls, continuous logic, sequential logic, etc.), select *Step Same* from the Debug menu of the ***batch debugger window***.

The next executable statement ***within*** the program section will be highlighted. The step same function will not execute out of program section. Execution of statements will stop after the last statement.

Go To Specific Line

To go to the first, last, or a specific line of the source code:

1. Verify the ***program window*** is the active window.
2. Select *Go To Line...* from the Search menu.
3. In the go to line dialog box, enter the desired line number and click on *OK* (specified line), *Home* (first line), or *End* (last line).

To go to the first, last, or a specific line of the source code and continue execution from the **program window** of the batch debugger:

1. Right click on a variable name appearing in the program code.
2. Select *Go To Line...* from the pop-up menu.
3. In the go to line dialog box, enter the desired line number and click on *OK* (specified line), *Home* (first line), or *End* (last line).

To go to the line of code that defines a variable from the **program window** of the batch debugger:

1. Right click on a variable name appearing in the program code.
2. Select *Go To Definition...* from the pop-up menu.

View and Watch Variables

It is possible to view or edit the value of program and built-in variables. It is also possible to put program variables (except constant and constant string variables) in a watch box so they can be monitored during program execution.

To view the value of a program variable from the batch debugger:

1. Verify the **program window** is the active window.
2. Right click on the variable name in the program code.
3. Select *Inspect...* from the pop-up menu.
4. View the current value.
5. Click on *OK*.

To edit the value of a program variable from the batch debugger:

1. Verify the **program window** is the active window.
2. Select *Stop* from the Debug menu.
3. Right click on the variable name in the program code.
4. Select *Inspect...* from the pop-up menu.

5. Enter a new value and click on *Change* in the dialog box.

To put a program variable into the watch box from the batch debugger:

1. Verify the **program window** is the active window.
2. Select *Stop* from the Debug menu.
3. Right click on the variable name in the program code.
4. Select *Inspect...* from the pop-up menu.
5. Click on *Add Watch* in the dialog box.
6. Click on *Close*.

To view the value of a built-in variable from the batch debugger:

1. Verify the **program window** is the active window.
2. Select *Stop* from the Debug menu.
3. Select *Builtin Variables* from the Debug menu.
4. Select the desired variable from the list provided in the built-in variable dialog box.
5. View the current value or enter (depending on variable type) a new value and click on *Change* in the dialog box.
6. Click on *OK*.
7. Click on *Close*.

Change Operation

To change the operation being executed from the batch debugger:

1. Verify the **program window** is the active window.
2. Select *Change Operation...* from the Debug menu.
3. Click on *Next*, *Previous*, or enter the number of the desired operation in the change operation dialog box.
4. Click on *OK*.

Change Phase Debugged

To change the phase whose source code is being debugged from the batch debugger:

1. Verify the ***program window*** is the active window.
2. Select *View Phase...* from the Debug menu.
3. Click on *Next*, *Previous*, or select the phase to be viewed from the list of phases provided in the view phase dialog box.
4. Click on *OK*.

Breakpoints

Breakpoints are pointers that stop program execution. They are useful when debugging because they allow programs to be executed in user defined pieces. The dynamic debugger allows two individual breakpoints to be active at one time.

To set or edit breakpoints from the batch debugger:

1. Verify the ***program window*** is the active window.
2. Select *Breakpoint...* from the Debug menu.
3. Enter a new first breakpoint (source code line number) in the breakpoint dialog box.
4. If desired, enter a new second breakpoint.
5. Click on *OK*.

To set or edit breakpoints from the ***program window*** of the batch debugger:

1. Verify the is the ***program window*** active window.
2. Right click on a variable name appearing in the program code.
3. Select *Breakpoints...* from the pop-up menu.
4. Enter a new first breakpoint (source code line number) in the breakpoint dialog box.
5. If desired, enter a new second breakpoint.
6. Click on *OK*.

To clear both breakpoints at the same time, click on *Clear All*.
To jump to a line of source code containing the breakpoint, click on *Go To* below the line number field.

Remove Variables From Watch

To remove a variable from the watch box of the ***batch debugger window***:

1. Click on the name of the variable in the watch box.
2. Select *Remove Watch* from the Debug menu or press the delete key.

Find Specific Text

To find specific text in the batch debugger:

1. Verify the ***program window*** is the active window.
2. Select *Find...* from the Search menu.
3. Enter the text to be searched for in the find dialog box.
4. Select the case matching option and the search direction.
5. Click on *Find Next*.

The cursor will indicate the line of code containing the specified text.

Escape Loop

Use this functionality to exit a loop when it is not practical to wait for the loop to process or if an infinite loop is encountered. To escape a loop, select *Escape Loop* from the Debug menu of the ***batch debugger window***.

Properties

The properties of the debug session include:

- Program name, creation date and time, and size.
- BSEQ function block status.
- MFP module mode.
- Program status.
- Indication of the disable program fault or hold logic state.

- Fault information.
- Recipe name, creation data and time, size, and ID.
- The operation and phase being executed.
- Unit name, creation date and time, and size.
- Loop, PCU, module, and function block address of the program.

To view or edit the properties of a debug session from the batch debugger:

1. Verify the **program window** is the active window.
2. Select *Properties...* from the Debug menu.
3. Edit or view (if the current user has access) the desired properties in the properties dialog box.
4. Click on *OK*.

To view the properties of the graphical display window, refer to [Section 7](#) or [Section 8](#).

SECTION 11 - ARCHIVE LOG

INTRODUCTION

The archive log saves the information messages generated by all INFI-BATCH tools into a log file. The types of messages that can be saved consist of fatal, warning, information, and diagnostic messages generated by the:

- Batch data manager.
- Unit procedure editor.
- Batch and UDF compilers.
- Master recipe editor.
- Dynamic debugger.
- Batch downloader.

The SYSTEM user for the project configures the write filter of the log to archive the desired information. Only the SYSTEM user can set the log write filter, delete a log, delete a log entry, or change number of messages to be stored. However, any user can set the log display filter, print a log or log entry, view and sort the log contents displayed. Changes made by any user other than the SYSTEM user will only effect the display of the log contents and not the actual condition of the archive log file. Figure 11-1 shows an example batch archive log window.

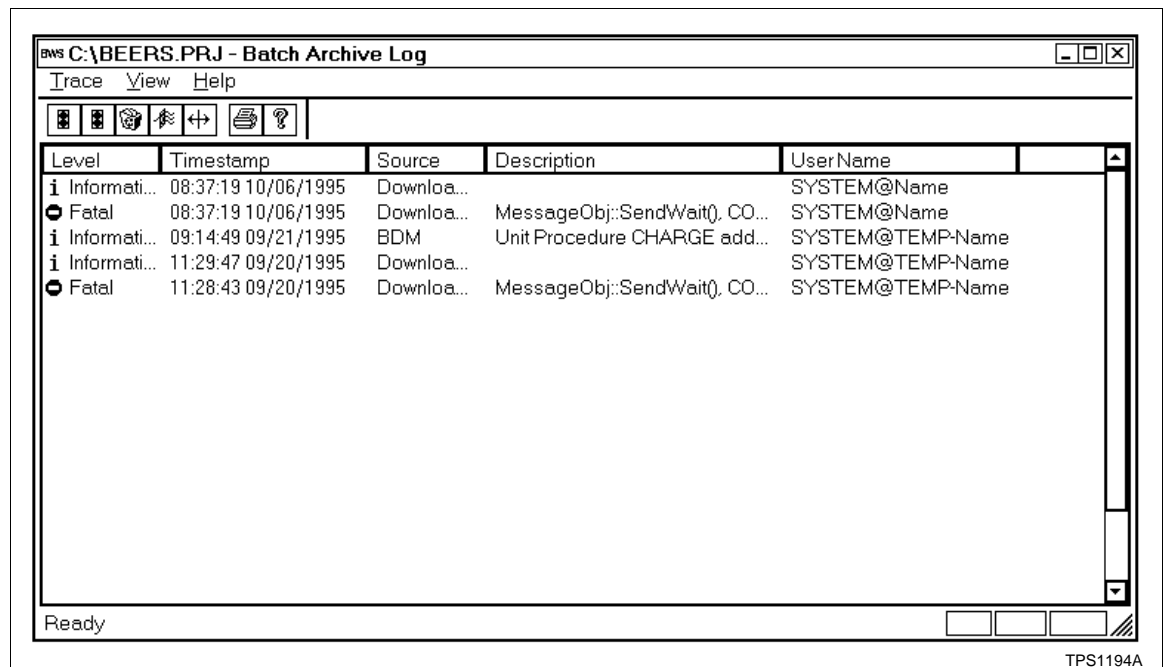


Figure 11-1. Batch Archive Log Window Example

VIEW

To view the contents of the archive log, select *Archive Log* from the Applications menu of the **batch data manager window**. The batch archive log window will appear.

NOTE: The batch archive log window is opened when the project is opened and not closed until the project is closed.

PAUSE

To pause the input of messages into the displayed log contents, select *Pause* from the Trace menu of the **batch archive log window**.

RESUME

To resume the input of messages into the displayed log contents, select *Resume* from the Trace menu of the **batch archive log window**.

SORT

The displayed log contents can be sorted according to message type (level), date and time the message was generated (timestamp), utility that generated the message (source), first letter of the message (description), and the name of the computer and user that caused the message to be generated (user name). To sort the displayed log contents, click on *Level*, *Timestamp*, *Source*, *Description*, or *User Name* in the **batch archive log window**.

PRINT

To print a complete log from the **batch archive log window**:

1. Select *Print...* from the Trace menu or right click anywhere (except on the level of a message) in the window and select *Print All* from the pop-up menu.
2. Select the print options as necessary in the standard Windows print dialog box.
3. Click on *OK*.

To print one displayed log entry from the **batch archive log window**:

1. Right click on the log entry to be printed.
2. Select *Print* from the pop-up menu.

3. Select the print options as necessary in the standard Windows print dialog box.
4. Click on *OK*.

To print multiple displayed log entries from the **batch archive log window**:

1. Select the log entries to be printed.
2. Right click in the level column of one of the selected log entries.
3. Select *Print Selected* from the pop-up menu.
4. Select the print options as necessary in the standard Windows print dialog box.
5. Click on *OK*.

FILTERS

There are two filters in the batch archive log. The display filter controls what log entries are to be displayed and can be manipulated by any user. The write filter controls which messages from which utility are logged into the log file and is configurable by the SYSTEM user only.

Display Filter

To configure the batch archive log display filter from the **batch archive log window**:

1. Select *Collection Filter...* from the Trace menu.

The event trace filter dialog box will appear.

2. On the display filter tab, select which previously logged messages (fatal, warning, information, and diagnostic) from each of the batch data manager, unit procedure editor, batch and UDF compiler, master recipe editor, batch debugger, and batch downloader utilities are to be displayed.

3. Click on *OK*.

Write Filter

To configure the batch archive log write filter from the **batch archive log window**:

1. Select *Collection Filter...* from the Trace menu.

The event trace filter dialog box will appear.

2. On the write filter tab, select which messages (fatal, warning, information, and diagnostic) are to be logged from each of the batch data manager, unit procedure editor, batch and UDF compiler, master recipe editor, batch debugger, and batch downloader utilities.
3. Click on *OK*.

LOG SIZE

The maximum number of log entries the log file will hold is configurable by the SYSTEM user only. To set the maximum number of log entries the log file will hold from the **batch archive log window**:

1. Select *Size...* from the Trace menu.
2. Enter the number of log entries to save in the number of events dialog box.
3. Click on *OK*.

The maximum number of log entries to be saved directly effects the maximum log file size.

DELETE

The ability to delete some or all log file entries from a log file is reserved for the SYSTEM user only.

To delete all contents of a log from the **batch archive log window**:

1. Select *Delete all* from the Trace menu.
2. Click on *OK* in the clear dialog box.

To delete all contents of a log from the **batch archive log window**:

1. Right click anywhere (except on the level of a message) in the window.
2. Select *Delete All* from the pop-up menu.
3. Click on *OK* in the delete dialog box.

To delete one displayed log entry from the **batch archive log window**:

1. Right click on the log entry to be deleted.
2. Select *Delete* from the pop-up menu.

3. Click on *OK* in the delete dialog box.

To delete multiple displayed log entries from the **batch archive log window**:

1. Select the log entries to be deleted.
2. Right click in the level column of one of the selected log entries.
3. Select *Delete Selected* from the pop-up menu.
4. Click on *OK* in the delete dialog box.

PROPERTIES

The properties of a message include:

- Date logged.
- Time logged.
- User and computer name.
- Name of the utility that generated the message.
- Message level (type).
- Description of the message.

To view the properties of a message from the **batch archive log window**:

1. Right click on the level (type) of the message.
2. Select *Properties* from the pop-up menu.
3. View the properties in the event detail dialog box.
4. Click on *Next* (view details of next newer log entry), *Previous* (view details of previous older log entry), *Delete*, or *Close*.

SECTION 12 - ERROR MESSAGES AND RECOVERY

COMPILER ERROR CODES

Table 12-1 provides a complete list of the compiler error numbers and the corresponding error messages.

Table 12-1. Compiler Error Messages

Number	Message
0	Operator not yet implemented
1	Invalid statement
2	Operator not allowed below base level
3	Duplicate definition
4	Invalid definition
5	Syntax error
6	Premature EOF
7	Undefined name
8	Undefined structure member
9	Constant left of equals sign
10	Constant expected
11	Invalid data type
12	Invalid compiler directive
13	Return not allowed outside function
14	Invalid time units
15	Unmatched END WHILE statement
16	Missing END WHILE statement
17	Unmatched UNTIL statement
18	Missing UNTIL statement
19	Declaration statements not allowed
20	Unmatched THEN statement
21	Unmatched ELSE statement
22	Missing ENDIF statement
23	Not allowed in fault logic
24	Invalid in Continuous Mode
25	Only one batch data area allowed
26	Not allowed outside of loops
27	Not allowed inside monitor
28	Invalid return for function type
29	Invalid call of nonfunction
30	Invalid function routine call
31	Incorrect number of arguments
32	Invalid argument type

Table 12-1. Compiler Error Messages (continued)

Number	Message
33	Invalid nonfunctional routine call
34	Invalid recursive function call
35	Invalid operator
36	Operator not allowed at base level
37	Not allowed outside step subroutine
38	Invalid structure reference
39	Invalid restart point
40	Not allowed outside of normal logic
41	Not allowed outside restart logic
42	Undefined label
43	Call by value item left of equals
44	Invalid input type
45	Batch data must precede step subroutines
46	Expression expected
47	Too many arguments
48	Invalid LC
49	Invalid block type
50	Invalid block number
51	Must have at least 1 case
52	Missing END CASE
53	Unexpected declaration statement
54	Illegal use of reserved word
55	Step subroutine name too long
56	Subscript out of range
57	Invalid usage
58	One and only one programmable block required
59	Block type not allocated
60	Not allowed outside of fault logic
61	Overlapping case range
62	Missing END CASE
63	List elements must be unique
64	Too many list elements
65	Not used
66	
67	
68	Value out of range
69	Function not referenced
70	Monitor not referenced
71	Invalid lower bound
72	Invalid upper bound
73	Array too large

Table 12-1. Compiler Error Messages (continued)

Number	Message
74	Name mismatch on END SUBR line
75	Array subscript out of bounds
76	Array dimensions incompatible
77	Too many dimensions
78	Missing subscript
79	Too many subscripts
80	Unmatched END FOR statement
81	Missing END FOR statement
82	Source line too long
83	Missing name
84	Only one descriptor allowed
85	Duplicate block number
86	Unterminated comment exists
87	Include files nested too deep
88	Missing terminator
89	Duplicate include file
90	Missing END SUBR statement
91	Missing <
92	Missing >
93	Missing ,
94	Missing =
95	Missing :
96	Missing .
97	Program too complex - Internal stack overflow
98	Function contains WAIT statements
99	Value too large
100	Undefined constant operation
101	Missing normal logic section
102	Local data value not used
103	Argument not used
104	Batch data value not used
105	Data value not initialized
106	Program too large for available space
107	Too many recipe arguments for one step subroutine
108	System constant file B90CONST.INC not used
109	Value too small
110	FOR LOOP index already in use in outer loop
111	Name mismatch on END FOR line
112	Complete address must be specified

RUNTIME FAULT CODES

The batch sequence function block executes a series of diagnostic tests that detect errors that cannot be detected by the compiler. The fault codes can be seen using the batch debugger or by viewing the batch sequence block output (N+7). The errors are only detectable while the module is in execution and are, therefore, called runtime errors. Table 12-2 lists the possible error codes and an explanation of each.

Table 12-2. Runtime Fault Codes

Fault Codes	Explanation
Any Positive Number	This number (any positive number) is assigned by the user in the batch language program and is used to indicate what type of fault has occurred. There is no limit to the number of fault codes the user may assign.
-1.0 Hold command	The batch sequence is in hold, through either the batch sequence block or a command in the batch language. Going to hold suspends normal logic and starts execution of hold logic.
-2.0 Debug command	The debug pushbutton has been depressed. This forces the language to the next statement unconditionally. This feature is largely unused since the dynamic debugger has the ability to do this.
-3.0 Stack overflow error	Contact Bailey Controls Technical Support Center.
-4.0 Error reading object file	No batch program exists in the NVRAM memory that matches the number indicated in specification 9 of the batch sequence block. Normally this means that the batch object file has not been downloaded to the module or specification 9 of the batch sequence block references an undefined program number.
-5.0 Object file exceeds memory allocation	The batch program size exceeds the amount of MFP volatile memory specified by specification 11 of the batch sequence function block. Correct this problem by increasing specification 11.
-7.0 Phase data size exceeds memory allocation	The stack size exceeds the amount of memory specified by specification 12 of the batch sequence function block. The stack contains certain data structures of the batch language (batch data declarations, global and local timers, integrators, etc.). To correct, increase the value of S12 in the batch sequence block.
-8.0 Recipe refers to undefined phase subroutine	The unit recipe contains a phase subroutine name that is not contained within the batch program. This situation can happen when a batch program is edited so that a phase subroutine is removed, but the corresponding recipes are not changed. To correct, add the undefined phase subroutine or remove the called (undefined) phase subroutine from the unit recipe.
-9.0 Batch directory error	Format module and reload necessary programs, recipes, and data files.
-10.0 Recipe file error	The user attempted to run a unit recipe that does not exist within the NVRAM memory of the MFP module. To correct, create or download a unit recipe to the module or input a valid program number and then restart the sequence.
-12.0 Recipe does not match program	There is an argument data type conflict between the unit recipe and the batch program. To correct, recompile the batch program and the recipe. Then, download both the recompiled batch program and unit recipe.
-13.0 Emergency shutdown	The emergency input to the batch sequence block (specification 5) is on. This drives the batch program unconditionally to operation 0 of the current unit recipe. To correct, find out why the emergency shut down input is being set to on and correct it.

Table 12-2. Runtime Fault Codes (continued)

Fault Codes	Explanation
-15.0 Invalid operation number	The user attempted to start a batch sequence at an operation number not defined within the unit recipe being run. Create a unit recipe with an operation number that matches the one the user is attempting to run or change the operation number.
-16.0 Bad function block reference	In the batch data declaration sections of the batch language, the program is trying to reference a function block that does not exist or one whose type does not match the function code type in the declaration. The batch debugger will give provide the function block number within the batch data section that is making the reference. To correct, change the function block number to a valid one, erase the reference from the program, or correct the type to match the function block in the MFP module.
-17.0 Array error	An array subscript is out of bounds. Normal logic is suspended and execution of fault logic begins. Note that it is possible for the user to inspect the value of the fault code to detect when this fault has occurred.
-18.0 BCODE revision mismatch	The batch program was compiled using firmware that does not match the firmware in the MFP module. Recompile the batch program with the compiler that matches the firmware within the MFP module.
-19.0 Recipe requires too many parallel phases	The user attempted to run a recipe that contains more parallel phase subroutines than are allowed in the target program. To correct, edit and recompile the unit recipe to contain less parallel phase subroutines or edit the #MAXPARALLEL statement.
-20.0 Recipe or phase subroutine parameter error	The unit recipe contains either the wrong number of recipe parameters or the type of parameters in the recipe does not agree with the target program. To correct, fix the recipe or the program (normally the recipe).
-21.0 Invalid on-line program change	The user attempted to run a new program that differs from the previous one in that there has been a change in the batch data area or the local declaration section of the active phase subroutine. Such on-line changes are not permitted. Refer to Section 2 for more information.
-22.0 Batch descriptor not unique	The batch and lot number in the BHIST function block are not unique to the Batch Historian. Change the batch and/or lot number and restart the program.
-23.0 Wait for batch historian	The batch historian is busy and the program may not proceed until it is available. No corrective action is required.
-24.0 Batch historian off-line	The Batch Historian is off-line and the program may not proceed until the batch historian is on-line and the program is restarted.
-25.0 Block reference error	The unit recipe used contains a reference to an incorrect or nonexistent block. Correct the block number in the unit recipe.
-26.0 Bad data reference in recipe	A data entry in the unit recipe does not match the program. This most commonly happens when a unit recipe argument value was selected from a selection lists and the program was changed to no longer include that selection. Recompile the unit recipe with the batch program and resolve any discrepancies.
-27.0 Bad block reference in UNIT DATA	A function block declaration in the unit data file does not match the MFP module configuration (either the function block address or the function code type are in error). Recompile the unit data file against the batch program and resolve any discrepancies.
-28.0 Bad data reference in UNIT DATA	A data entry in the unit data file does not match the program. Recompile the unit data file against the batch program and resolve any discrepancies.

Table 12-2. Runtime Fault Codes (continued)

Fault Codes	Explanation
-29.0 UNIT DATA does not match the B90 program	The unit data file does not match the batch program structurally (the number or type declarations do not match). Recompile the unit data file against the batch program and resolve any discrepancies.
-30.0 Error reading UNIT DEF OBJ file	No unit data file exists in the NVRAM memory that matches the number indicated by specification 9 of the BSEQ function block. Normally this means that the unit data object file has not been downloaded to the module.

Index

A

Abbreviations.....	1-4
Add	
Class.....	5-3
Class library.....	5-2
Control recipe.....	8-20
Master recipe.....	8-1
Operation.....	7-5, 8-8
Phase.....	7-7, 8-10
Project.....	4-1
Segment.....	8-12
Source file.....	6-3, 6-8
Train.....	8-13
UDF program.....	6-4
Unit.....	6-9
Unit procedure.....	7-2, 8-5
Unit recipe.....	7-12
User name.....	4-5
Archive log	
Delete entries.....	11-4
Filters.....	11-3
Pause.....	11-2
Print.....	11-2
Resume.....	11-2
Size.....	11-4
Sort.....	11-2
View.....	11-2

B

Batch 90 support description.....	2-8
Batch data manager description.....	2-2
Batch process description.....	2-1
Batch program	
Add source file.....	6-3
Compile.....	6-1
Delete.....	9-6
Download.....	9-2
Edit.....	6-1
Print.....	6-3
Verify.....	9-4
Batch project description.....	2-2
Batch recipe description.....	2-1

C

Change class program.....	5-6
Class	
Add.....	5-3
Change program.....	5-6
Copy.....	5-5
Delete.....	5-6

Properties.....	5-7
Rename.....	5-6
Class library	
Add.....	5-2
Copy.....	5-3
Delete.....	5-1
Properties.....	5-7
Refresh.....	7-9, 8-16
Remove reference.....	5-1
Rename.....	5-2
Compile	
Batch program.....	6-1
UDF program.....	6-6
Unit.....	6-9
Compiler errors.....	12-1
Control recipe	
Add.....	8-20
Copy.....	8-23
Debug.....	10-2
Delete.....	8-24, 9-7
Download.....	9-4
Edit.....	8-22
Properties.....	8-29
Rename.....	8-23
Verify.....	9-5
Copy	
Class.....	5-5
Class library.....	5-3
Control recipe.....	8-23
Master recipe.....	8-16
Project.....	4-3
Train.....	8-14
UDF program.....	6-5
Unit procedure.....	7-9
Unit recipe.....	7-14

D

Debug.....	10-1
Delete	
Batch program.....	9-6
Class.....	5-6
Class library.....	5-1
Control recipe.....	8-24, 9-7
Master recipe.....	8-20
Operation.....	7-8, 8-16
Phase.....	7-8, 8-16
Project.....	4-3
Segment.....	8-16
Train.....	8-15
UDF program.....	6-7, 9-6
Unit.....	6-11, 9-6

Index (continued)

- Unit procedure.....7-11, 8-16
- Unit recipe7-15, 9-6
- User name.....4-5, 4-6
- Description
 - Batch 90 support2-8
 - Batch data manager2-2
 - Batch process.....2-1
 - Batch Project.....2-2
 - Batch recipe2-1
 - Directory structure3-2
 - Files.....3-2
 - Online program changes2-10
 - UDF program.....2-10
 - UDF project2-7
- Directory structure.....3-2
- Document conventions.....1-3
- Download
 - Batch program.....9-2
 - Control recipe9-4
 - Information9-2
 - UDF program.....9-2
 - Unit.....9-3
 - Unit recipe9-3
- Download utility
 - Delete messages.....9-10
 - Print messages.....9-11
 - Size9-10
- Drive properties.....4-7

E

- Edit
 - Batch program.....6-1
 - Control recipe8-22
 - Master recipe.....8-2
 - Train8-14
 - UDF program.....6-4
 - Unit procedure.....7-2
 - Unit recipe7-13
 - Users list.....4-4
- Errors12-1
- Export
 - Operation.....7-6, 8-9
 - Phase7-8, 8-11
 - Unit procedure.....7-4, 8-6

F

- Fault codes12-4
- Files3-2
- Format MFP module9-1

G

- Glossary1-4

H

- Hardware requirements1-1
- How to use this instruction.....1-2

I

- Import
 - Operation7-6, 8-9
 - Phase.....7-8, 8-11
 - Unit procedure7-5, 8-7
- Installation3-1
- Instruction content1-2
- Intended user.....1-1

M

- Master recipe
 - Add8-1
 - Copy8-16
 - Delete8-20
 - Edit.....8-2
 - Print8-18
 - Properties8-25
 - Rename8-17
- Master recipe editor properties.....8-28
- Modify user privileges.....4-6
- Module directories9-8
- Module mode.....9-8

O

- On-line program changes2-10
- Operation
 - Add7-5, 8-8
 - Delete7-8, 8-16
 - Export7-6, 8-9
 - Import.....7-6, 8-9
 - Properties7-18, 8-28
 - Rename7-6, 8-10

P

- Phase
 - Add7-7, 8-10
 - Delete7-8, 8-16
 - Export7-8, 8-11
 - Import.....7-8, 8-11
 - Properties7-18, 8-28

Index (continued)

Terminate.....	7-8, 8-12
Print	
Batch program	6-3
Master recipe.....	8-18
UDF program	6-8
Unit	6-11
Unit procedure	7-10, 8-7
Project	
Add	4-1
Close.....	4-2
Copy	4-3
Delete	4-3
Open	4-2
Properties	4-7
Rename	4-4
Security.....	4-8
Properties	
Class.....	5-7
Class library	5-7
Control recipe	8-29
Debug session	10-8
Drive	4-7
Master recipe	8-25
Master recipe editor	8-28
Message	9-11
Operation	7-18, 8-28
Phase.....	7-18, 8-28
Project.....	4-7
Segment	8-26
Train.....	8-30
Unit	6-11
Unit procedure	7-16, 8-27
Unit procedure editor	7-19
Unit recipe.....	7-19
Users list.....	4-8
R	
Rebuild	6-12
Reference documents	1-4
Refresh	
Class library	7-9, 8-16
Projects window	4-8
Users list.....	4-5
Remove class library reference.....	5-1
Rename	
Class.....	5-6
Class library	5-2
Control recipe	8-23
Master recipe.....	8-17
Operation.....	7-6, 8-10
Project.....	4-4
Segment.....	8-13
Train	8-15
UDF program	6-7
Unit procedure.....	7-10
Unit recipe	7-15
Requirements	
Hardware.....	1-1
Software	1-2
Reset user password	4-6
S	
Search.....	7-9, 8-16, 9-9
Security	4-8
Segment	
Add.....	8-12
Delete	8-16
Properties.....	8-26
Rename.....	8-13
Software requirements.....	1-2
Start-up	3-1
T	
Terminate phase	7-8, 8-12
Terms.....	1-4
Train	
Add.....	8-13
Copy	8-14
Delete	8-15
Edit	8-14
Properties.....	8-30
Rename.....	8-15
U	
UDF program	
Add.....	6-4
Add source file	6-8
Compile	6-6
Copy	6-5
Debug.....	10-3
Delete	6-7, 9-6
Description	2-10
Download	9-2
Edit	6-4
Print.....	6-8
Rename.....	6-7
Verify	9-4
UDF project description	2-7
Unit	
Add.....	6-9
Compile	6-9

For prompt, personal attention to your instrumentation and control needs or a full listing of Bailey representatives in principal cities around the world, contact the Bailey location nearest you.

Australia

Elsag Bailey Pty. Limited
Regents Park, NSW
Phone: 61-2-645-3322
Telefax: 61-2-645-2212

Japan

Bailey Japan Company, Ltd.
Tagata-Gun, Shizuoka-Ken
Phone: 81-559-49-3311
Telefax: 81-559-49-1114

United Kingdom

Bailey Automation plc
Telford, Shropshire
Phone: 44-1952-670-477
Telefax: 44-1952-670-455

Brazil

Bailey do Brasil
São Paulo
Phone: 55-11-548-4122
Telefax: 55-11-547-0315

Jordan

Bailey Controls Jordan
Amman
Phone: 962-6-788-116
Telefax: 962-6-756-908

United States

Bailey Controls Company
Wickliffe, Ohio
Phone: 1-216-585-8500
Telefax: 1-216-585-8756

Canada

Elsag Bailey (Canada), Inc.
Burlington, Ontario
Phone: 1-905-639-8840
Telefax: 1-905-639-8639

Mexico

Bailey Mexico S.A. de C.V.
Naucalpan
Phone: 52-5-557-6100
Telefax: 52-5-557-7022

Venezuela

Bailey de Venezuela SA
Valencia
Phone: 58-41-329-196
Telefax: 58-41-327-632

France

Elsag Bailey S.A.
Massy
Phone: 33-1-64-47-2000
Telefax: 33-1-64-47-2016

Norway

Bailey Norge A.S.
Bergen
Phone: 47-55-222-000
Telefax: 47-55-222-010

Germany

Bailey-F & P Automation GmbH
Overath
Phone: 49-220-473-90
Telefax: 49-220-473-979

People's Republic of China

Bailey Beijing Controls
Beijing
Phone: 86-10-401-0651
Telefax: 86-10-401-1643

Italy

Elsag Bailey
Genoa
Phone: 39-10-658-1
Telefax: 39-10-658-2941

Singapore

Elsag Bailey Pte. Ltd.
Singapore
Phone: 65-442-3200
Telefax: 65-442-2700