



BSLBR14

User Manual

E-mail: Support@abbydalesystems.com

COPYRIGHT

This computer programming material remains the exclusive property of Abbydale Systems LLC..
Permission for its use may be obtained by contacting:

Abbydale Systems LLC.
2925 Gulf Freeway South
Suite B #229
LEAGUE CITY
Texas USA
77573

ATTN: K.E.Ferguson
Kevin@abbydalesystems.com

Disclaimer

Kevin E. Ferguson of **Abbydale Systems LLC.**, has developed this computer program and associated materials.

This material has been used successfully by **Abbydale Systems LLC.** and to the best of our knowledge this material and any system(s) of which it is a part are operational as of the service level or date stated in the body of this material (if so stated). However, ***no warranty*** is given or implied as to the accuracy of this material or any related material or systems, and ***no responsibility*** is assumed for any effect or modification directly or indirectly caused by the use of this material.

It is the responsibility of any user of this material to evaluate its usefulness to the user's environment.

Abbydale Systems LLC. does not guarantee to keep this nor any related material current, nor does it guarantee to provide any corrections or extensions described by any users of this material or any corrections or extensions made in the future by **Abbydale Systems LLC.** itself.

Acknowledgements

This document refers to number of software products that are produced by other companies. In most cases the names of these products are trademarks and/or copyright of those companies. It is not our intention to claim either the name of the trademark, nor the product itself, these remain solely the right of the owning companies.

CONTENTS

COPYRIGHT 2

Disclaimer..... 2

Acknowledgements..... 2

1 Overview..... 6

1.1 Warning6

1.2 Caution.....6

2 Installation Procedure..... 7

2.1 From XMI File7

2.2 By Assembling the Source Code7

 2.2.1 Sample Assembly Job.....7

3 Using BSLBR14 8

3.1 Available Parameters.....8

 WAIT=nnn8

 RC=nn | CODE=nn | COND=nn8

 ABEND=cccc8

3.2 Issuing Messages From BSLBR14.....9

3.3 Sample JCL for BSLBR1410

 Example 110

 Example 2.....10

 Example 3.....10

 Example 4.....10

 Example 5.....11

4 Messages 12

BSLBR01E.....12

BSLBR02E.....12

BSLBR03I.....13

BSLBR04E.....13

BSLBR05E.....14

BSLBR06E.....14

BSLBR07E.....15

5 Summary of Amendments..... 16

Obtaining Support..... 17

1 Overview

BSLBR14 is a flexible alternative to the IBM supplied program, IEFBR14. It functions in much the same way as IEFBR14 with the following notable exceptions.

- **BSLBR14** will wait before ending
- **BSLBR14** can be made to issue any message
- **BSLBR14** can return any condition code or any ABEND code

Having the ability to request how **BSLBR14** should terminate, coupled with its ability to issue any message (or sequence of messages) gives users an ideal tool to prepare jobstreams to test error handling/recovery.

You can make **BSLBR14** terminate with **any** system or user ABEND or with **any** condition code.

1.1 Warning

BSLBR14 is currently *not* re-entrant and therefore should *not* be considered as a total replacement for IEFBR14.

IEFBR14 is often used as a dummy routine for various system type modules; **BSLBR14** is not a dummy routine and should **not** be used to replace IEFBR14.

1.2 Caution

As **BSLBR14** can also issue any message that you ask it to (including its own messages), please check **very** carefully that someone isn't pulling your leg.

2 Installation Procedure

2.1 From XMI File

The XMI (or XMIT) file is in IBM TSO TRANSMIT format and **must** be transferred to z/OS™ as a fixed blocked 80 byte BINARY file. The disk space requirement for the file is 2 tracks of 3390 disk.

The FTP process (if performed in a 3270 emulator) must be performed in TSO READY mode or in option 6 of ISPF™.

The dataset name used as input for the TSO TRANSMIT was ABBYDALE.BSLBR14.LOADLIB. Unless this is changed by the TSO RECEIVE command it will be the name of the dataset created by the RECEIVE command.

2.2 By Assembling the Source Code

To assemble **BSLBR14** the source code must be saved in a partitioned dataset and assembled using High Level Assembler. (ASMA90).

It should be linked using LIST, LET, NCAL, MAP LOADER options.

2.2.1 Sample Assembly Job

The following is a sample assemble and link JCL the can be used to assemble **BSLBR14**

```
//STEP1 EXEC PGM=ASMA90, PARM='OBJECT,NODECK', REGION=1024K
//SYSPRINT DD SYSOUT=*
//SYSLIB DD DSN=SYS1.MACLIB, DISP=SHR
// DD DSN=SYS1.AMODGEN, DISP=SHR
//SYSUT1 DD SPACE=(CYL,(2,1)), UNIT=SYSDA, DSN=&&WRK1
//SYSLIN DD DSN=&&WRK5, UNIT=SYSDA, SPACE=(CYL,(2,1)),
// DCB=(RECFM=FB, BLKSIZE=3200, LRECL=80), DISP=(, PASS)
//SYSIN DD DSN=your.input.pds (BSLBR14), DISP=SHR
//LKED1 EXEC PGM=IEWL, PARM='LIST,LET,NCAL,MAP', COND=(4,LT)
//SYSLMOD DD DSN=your.output.load.library (BSLBR14), DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSLIN DD DSN=&&WRK5, DISP=(OLD,DELETE)
```

3 Using BSLBR14

The program is mostly controlled by means of parameters passed to it on the JCL EXEC card. The only exception to this is for passing messages to **BSLBR14** for it to issue. These are coded on the EXEC card. There are 3 keywords for the passed parameters, they can appear in any order and are explained below.

3.1 Available Parameters

WAIT=nnn

This keyword will cause **BSLBR14** to wait for the specified number of SECONDS before it terminates with either the desired condition code or the desired ABEND code. This parameter **must** be numeric and may be up to 3 digits in length only. Leading zeros may be omitted.

The default wait time is 20 seconds.

RC=nn | CODE=nn | COND=nn

Any of these keywords determine the return code produced by this execution of **BSLBR14**; the value of these parameter **must** be numeric and may be up to 2 digits in length only.

The default return code is 00

ABEND=cccc

This keyword determines the ABEND code that this execution of **BSLBR14** terminates with. The ABEND code can be either a **U** for a user ABEND or a **S** for a system ABEND.

i.e. U0099 will cause **BSLBR14** to terminate with a user 99 abend.

SB37 will cause **BSLBR14** to terminate with a SB37 abend.

The first character of this parameter **must** be either U or S.

The value following the ABEND type may be up to a maximum of 4 digits. These digits **must** be either numeric or the letters A through to F.

If the ABEND code supplied is syntactically invalid a default abend of U0001 will be used.

Note:- The code specified need **not** be a valid IBM ABEND code. No checking is performed for the validity of the ABEND code.

By default **BSLBR14** will **not** ABEND.

3.2 Issuing Messages From BSLBR14

Any messages may be issued from **BSLBR14** via the optional SYSIN DD card.

The maximum length of message that will be issued is 80 bytes, however, multiple messages can be issued.

Messages are written, via a Write To Operator (WTO) with routing codes of:

Route Code 2 - Write message to Master Console.
Route Code 11 - Write message to the JOB log.

There is currently no facility written to allow these to be changed dynamically from within the job stream. This facility is under review for the next release of the program.

There is no checking for the validity of any messages issued or for their 'usual' routing and descriptor codes.

Any customer message suppression currently in use will effect **BSLBR14**, as will any automation based upon messages, including Message Processing Facility (MPF) exits.

BSLBR14 will issue its messages **before** processing its wait.

Coding a WAIT=nnnn (See **Available Parameters** section in this manual) in the SYSIN stream will cause **BSLBR14** to wait nnnn seconds between issuing messages. Any number of WAIT= statements may be coded in this way thereby allowing differing wait intervals between messages to be achieved.

The WAIT= SYSIN card has exactly the same format as that of the WAIT= parameter. It should be noted that these waits are in addition to the wait specified (if any) on the EXEC card. All SYSIN specified waits are executed before that specified on the EXEC card.

The WAIT= statements in a SYSIN stream only effect the WAIT interval between messages, **no messages, no wait**.

By default there is no wait time between messages.

An invalid WAIT= statement will result in the default of 20 seconds being used.

All information on the WAIT= cards after the wait time is ignored.

No messages can be issued that start with the characters 'WAIT=' as the program treats these as potentially valid WAIT= cards.

As **BSLBR14** can issue any message, it is recommended that access to the program be carefully controlled to prevent misuse and abuse.

3.3 Sample JCL for BSLBR14

The following JCL samples are examples of how to run **BSLBR14**

Example 1

```
//stepname EXEC PGM=BSLBR14
```

This JCL will assume all of the **BSLBR14** defaults so it will take 20 seconds to run and will return COND CODE 0

Example 2

```
//stepname EXEC PGM=BSLBR14,PARM='RC=16,WAIT=45'
```

This JCL will take 45 seconds to run and will return COND CODE 16.

Example 3

```
//stepname EXEC PGM=BSLBR14,PARM='ABEND=S806,WAIT=30'
```

This JCL will take 30 seconds to run and will crash with a S806 ABEND.

Example 4

```
//stepname EXEC PGM=BSLBR14,PARM='WAIT=5,RC=4'  
//SYSIN      DD *  
IEF009Z Please ignore this message
```

This JCL will issue the message 'IEF009Z PLEASE IGNORE THIS MESSAGE' then wait 5 seconds before ending with a COND CODE 4.

Obviously as BSLBR14 can issue **any** messages you could issue messages that **BSLBR14** may itself issue so a careful check of the job stream is recommended.

Example 5

```
//stepname EXEC PGM=BSLBR14,PARM='ABEND=U0001,WAIT=2'  
//SYSIN      DD *  
ABC message  
WAIT=20     wait for 20 seconds  
XYZ message
```

This JCL will issue the message 'ABC message'. It will then wait 20 wall clock seconds and issue the message 'XYZ message'. It will then wait a further 2 wall clock seconds and then abend with a U0001 abend.

4 Messages

BSLBR01E

BSLBR01E - INVALID PARMS SPECIFIED - WAIT 20 USED

And

BSLBR01E- INVALID PARMS SPECIFIED - RETURN CODE 0000 USED

Meaning

Due to a detected error in the PARM field the default values for WAIT (20 seconds) and for return code (0000) was used.

This is an informational message but it may mean that the execution desired has not taken place as requested.

Corrective Action

Correct the error and rerun if required.

BSLBR02E

BSLBR02E - INVALID WAIT TIME PASSED - WAIT 20 USED

Meaning

Due to a detected error in the WAIT parameter that was passed to the program the default value (20 seconds) will be used.

This is an informational message but it may mean that the execution desired has not taken place as requested.

Corrective Action

Correct the error and rerun if required.

BSLBR03I**BSLBR03I - NO RETURN CODE SPECIFIED 0000 USED****Meaning**

RC= was specified with no value **BSLBR14** has therefore has given you the default value of 0000.

This is an informational message but it may mean that execution you were expecting has not taken place.

Corrective Action

Correct the error and rerun if required.

BSLBR04E**BSLBR04E - INVALID NUMERICS IN THE RC= PARAMETER. PARM IGNORED****Meaning**

An invalid value was found in the RC= parm **BSLBR14** has therefore used the default value of 0000.

This is an informational message but it may mean that execution you were expecting has not taken place.

Corrective Action

Correct the error and rerun if required.

BSLBR05E**BSLBR05E - NO ABEND CODE SPECIFIED - IGNORED****Meaning**

ABEND= was coded with no value. **BSLBR14** will ignore the parameter.

This is an error message. It will probably mean that execution you were expecting has not taken place.

Corrective Action

Correct the error and rerun if required.

BSLBR06E**BSLBR06E - INVALID ABEND CODE SPECIFIED U0001 USED****Meaning**

Invalid characters were detected in the ABEND= parameter. This could be that the User code specified is not numeric, or that you have used letters other than A, B, C, D, E, and F in a system ABEND code.

BSLBR14 will assume the default and terminate with a U0001 Abend.

This is an informational message but it may mean that execution you were expecting has not taken place.

Corrective Action

Correct the error and rerun if required.

BSLBR07E**BSLBR07E - INVALID NUMERICS IN PARAMETER - PARM IGNORED****Meaning**

The abend specified is a value other than U or S. **BSLBR14** will use the default abend of U0001 as the ABEND code for this execution.

This is an error message that will mean that the execution requested may not have taken place.

Corrective Action

Correct the error and rerun if required.

5 Summary of Amendments

Date	Version	Fix Id.	Comment
30 th May 2017	6.0	n/a	Made program 31 bit addressable.
28 th May 2017	5.1	n/a	Transferred to Abbydale Systems LLC.
20 th February 2012	5.1	JOB02	Added Replace_Jobname subroutine
19 th February 2012	5.0	JOB01	Commented out redundant code

Obtaining Support

Support for, comments about and suggestions for enhancements for this product can be obtained from our website :

www.abbydalesystems.com

or by emailing us at

support@abbydalesystems.com

In order to assist us in filtering support emails please specify in the heading of the email the name of the product that you require support on.

Spam will not be tolerated at this email address.

Where source code is provided for the product, support will be on a 'best efforts' basis. Where the user site has modified the source code, support may entail requesting copies of that sites source code and may result in support being withdrawn if this is not provided.

Abbydale Systems LLC. reserves the right to any code modifications that may have been undertaken at the user site.

Any alteration of the copyright information contained in the original source code is an infringement of the copyright of this and any other Abbydale Systems product and may result in legal action being taken against the perpetrator.

