## Topics


| | | |
|---|---|---|
| **9** | **Absolute Lister Description** | **9-3** |
| 9.1 | Producing an Absolute Listing | 9-4 |
| 9.2 | Invoking the Absolute Lister | 9-5 |
| 9.3 | Absolute Lister Example | 9-6 |


## Figures


| Fig. | Title | Page |
|---|---|---|
| 9.1 | Absolute Lister Development Flow | 9-4 |

# 9 Absolute Lister Description

The MSP430 absolute lister is a debugging tool. This utility accepts linked object files as input and creates .abs files as output. These .abs files can be assembled to produce a listing that shows the absolute addresses of object code. Normally, this is a tedious process requiring many manual operations; the absolute lister utility, however, performs these operations automatically.

### 9.1 Producing an Absolute Listing

The figure illustrates the steps required to produce an absolute listing.

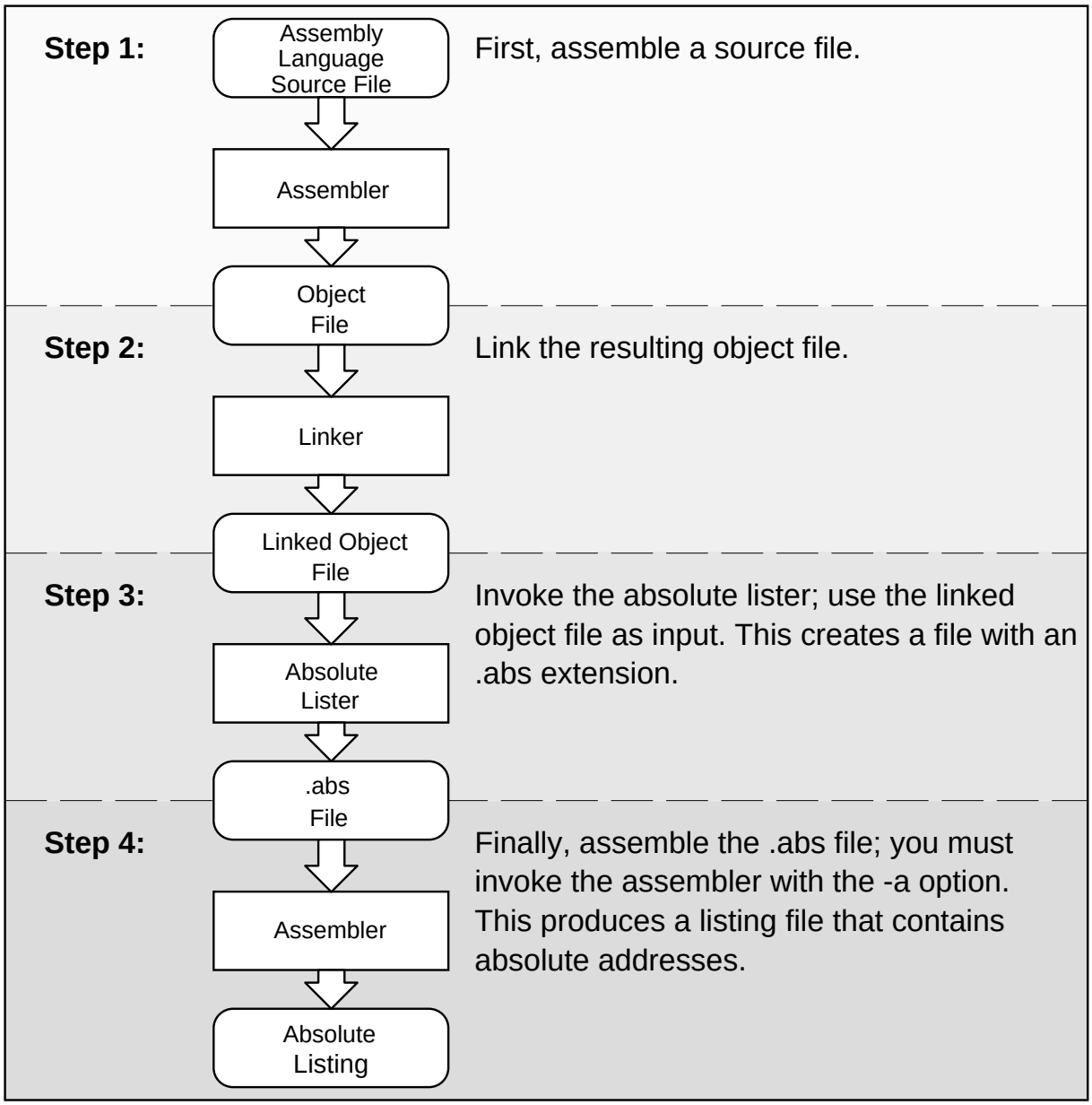


**Figure 9.1:** Absolute Lister Development Flow

### 9.2 Invoking the Absolute Lister

The syntax for invoking the absolute lister is:

**abs430** *filename*

where *filename* must be a linked object file. The absolute lister assumes that this file has an extension of .out. (This is the extension that the linker produces for output files).

If you omit the filename when you invoke the absolute lister, the utility prompts you for a filename.

The absolute lister produces an output file for each file that was linked to create *filename*.out. These files are named with the individual filenames and an extension of **.abs**.

Assemble this file and use the -a assembler option to create the absolute listing:

**asm430** *filename*.abs -a

### 9.3 Absolute Lister Example

This example uses three source files. Note that module1.asm and module2.asm both include the file globals.def.

**module1.asm**

```
.bss    xflags,2
.bss    flags
.copy   "globals.def"
.text
bis     #Gflag,&flags
```

**module2.asm**

```
.copy   "globals.def"
.text
bic     #Gflag,&flags
```

**globals.def**

```
        .global  flags
Gflag   .set     2
```

The following steps create absolute listings for the files module1.asm and module2.asm:

**Step 1:** First, assemble module1.asm and module2.asm:

```
asm430 module1
asm430 module2
```

This creates two object files called module1.obj and module2.obj.

**Step 2:** Next, link module1.obj and module2.obj. using the following linker command file, called abstest.cmd:

```
/*****************************************************/
/* File abstest.cmd -- COFF linker control file      */
/* for linking MSP430 modules                        */
/*****************************************************/
-o ABSTEST.OUT         /* executable output file     */
-m ABSTEST.MAP         /* output map file            */

/* input files                                       */
MODULE1.OBJ
MODULE2.OBJ

/* define MSP430 memory map                          */
MEMORY
{
       RAM:          origin=00200h  length=0100h
       ROM:          origin=0F000h  length=1000h
}

/* define the output sections                         */
SECTIONS
{
       .bss:           >RAM
       .text:          >ROM
}
```

Invoke the linker:

**lnk430** abstest.cmd

This creates an executable object file called abstest.out; use this new file as input for the absolute lister.

**Step 3:** Now, invoke the absolute lister:

**abs430** abstest.out

This creates two files called module1.abs and module2.abs:

**module1.abs:**

```
                        .nolist
flags                   .setsym     0202h
.bss                    .setsym     0200h
end                     .setsym     0203h
.text                   .setsym     0f000h
etext                   .setsym     0f008h
.data                   .setsym     00h
edata                   .setsym     00h
                        .setsect    ".text",0f000h
                        .setsect    ".data",00h
                        .setsect    ".bss",0200h
                        .list
                        .text
                        .copy       "MODULE1.ASM"
```

**module2.abs:**

```
                        .nolist
flags                   .setsym     0202h
.bss                    .setsym     0200h
end                     .setsym     0203h
.text                   .setsym     0f000h
etext                   .setsym     0f008h
.data                   .setsym     00h
edata                   .setsym     00h
                        .setsect    ".text",0f004h
                        .setsect    ".data",00h
                        .setsect    ".bss",0203h
                        .list
                        .text
                        .copy       "MODULE2.ASM"
```

These files have information that the assembler needs when you invoke it in step 4:

- They contain .setsym directives, which equate values to global symbols. Both files contain global equates for the symbol flags. The symbol flags was defined in the file globals.def, which was included in module1.asm and module2.asm.
- They contain .setsect directives, which define the absolute addresses for sections.
- They contain .copy directives, which tell the assembler which assembly language source file to include.

Note that the .setsym and .setsect directives are not useful in normal assembly; they are useful only for creating absolute listings.

**Step 4:** Finally, assemble the .abs files created by the absolute lister (remember that you must use the -a option when you invoke the assembler):

```
asm430  -a  module1.abs
asm430  -a  module2.abs
```

This creates two listing files called module1.lst and module2.lst; no object code is produced. These listing files are similar to normal listing files; however, the addresses shown are absolute addresses. The absolute listing files created are:

**module1.lst:**

```
MSP430 Macro Assembler  Version 1.00 [04/94]     Wed May 25 14:12:55 1994
 Copyright (c) 1994    Texas Instruments Incorporated

MODULE1.ABS                                                        PAGE    1

       13 f000                              .text
       14                                   .copy       "MODULE1.ASM"
 A      1 0200                             .bss    xflags,2
 A      2 0202                             .bss    flags
 A      3                                  .copy   "globals.def"
 B      1                                  .global flags
 B      2        02               Gflag    .set    2
 B      3
 B      4
 A      4 f000                             .text
 A      5 f000  -d3a20202                 bis     #Gflag, &flags
 A      6

 No Errors,  No Warnings
```

**module2.lst:**

```
MSP430 Macro Assembler  Version 1.00 [04/94]     Wed May 25 14:42:20 1994
 Copyright (c) 1994    Texas Instruments Incorporated

MODULE2.ABS                                                      PAGE    1

       13 f004                              .text
       14                                   .copy        "MODULE2.ASM"
 A      1                                   .copy   "globals.def"
 B      1                                   .global flags
 B      2        02               Gflag     .set    2
 B      3
 B      4
 A      2 f004                              .text
 A      3 f004  !c3a20202                   bic     #Gflag, &flags
 A      4

 No Errors,  No Warnings
```