

## B\$FDIO – Call Access Method Entry Point via FDPTR

The B\$FDIO routine calls the entry-point of an Access Method via the FDPTR pointer field within the FD.

### 1. Invocation

To call the entry-point of an Access Method code:

```
$PUSH fd
$LOAD opcode
CALL B$FDIO
```

or:

```
$PUSH fd
$LOAD opcode
$PUSH mem_area
CALL B$FDIO
```

where *fd* is a File Definition control block (FD), *opcode* is the operation code to be executed by the Access Method and *mem\_area* is an optional memory area required for some Access Method operations (e.g. the read/write buffer for read and write operations).

### 2. STOP Codes and Exception Conditions

No STOP codes are generated by B\$FDIO.

The following exception conditions are returned by B\$FDIO:

EXIT code	\$\$COND	Description
xxxxyy	yy	B\$FDIO merely returns the exception code returned by the Access Method

### 3. Programming Notes

B\$FDIO is reserved for internal use by GSM and Speedbase only. This routine is never coded explicitly but is generated by the compiler for the various file i/o verbs (e.g. OPEN, READ, WRITE, CLOSE etc.).

Although B\$FDIO can be called with either a single parameter, or two parameters pushed on the Parameter Stack it always calls the appropriate Access Method, via FDPTR, with **exactly two** parameters pushed on the Parameter Stack. Routing **all** file i/o via B\$FDIO avoids the need for each individual Access Method to cope with a variable number of pushed parameters.

### 4. Examples

None.

### 5. Copy-Books

None.

### 6. See Also

None.

