To: MSPM Distribution.

From: J. F. Ossanna.

Subj: BF.1.02.

Date: April 21, 1967.

The attached BF.1.02 uses some new I/O System terminology. Attachment graph "terminals" are now called "nodes". Each node is either an indirect frame node (previously called stream terminals) or a direct frame node (previously called frame terminals). What was a "streamname" is now an "indirect-framename". BF.1.03, now in preparation, will explain the attachment graph and the associated terminology.

Published: 4/21/67

Identification

I/O System Modes and Device-Mode Relationships.
J. F. Ossanna.

Purpose

This document enumerates, defines, and discusses the standard I/O System modes and their properties. These modes are defined in a way which permits uniform interpretation within the I/O System. The applicability of these modes to various devices is described. Device-dependent hardware modes are discussed briefly.

Modes: General

I/O System (IOS) modes are provided to permit user selection of alternative courses of IOS operation, to permit, for example, user-controlled restriction of IOS usage. Any particular mode is specified or requested in one and only one of three ways: (1) by inclusion in the mode argument of an attach or changemode call (see Section BF.1.01); (2) by means of a specially-tailored call, such as the readsync mode in the readsync call; or (3) as the request argument in an order call (see Section BF.1.07). This document is primarily concerned with modes controlled by method (1). Modes controlled by method (2) are enumerated but not described herein; references to appropriate subsections of Section BF are included. Modes controlled by method (3) include hardware device modes used when doing physical input/output, various single-device-oriented special modes, and certain internal modes communicated between IOS modules; these modes are enumerated in an appendix of Section BF.1.07, and are described in appropriate subsections of Section BF.

The major mode classifications described later in this document are the Access, Data, Use, Code, Synchronization, and Attachment modes. The mode properties to be described include: the default mode in a mode class; whether or not the mode is changeable; with which attachment graph component the mode is associated (see Section BF.1.03); and the mutual exclusivity of submodes with respect to applicibility.

The mode Argument

The mode argument in the attach and changemode calls is a character string of arbitrary length. The individual modes are represented by character strings composed by concatenating characters taken from the set consisting of the 52 alphabetic letters plus the "not" character (ASCII circumflex). In this document the "not" character will be represented within text by the mnemonic string "not". The mode argument is composed by concatenating the individual mode strings separated by a mode delimiter. The latter may be any convenient special character except "not". Typical mode strings are more than one character long and are mnemonic. Many of the commonly-used modes are additionally represented by a one-character mnemonic. For example, the two permissible mnemonics for the Readable mode are "READ" and "R".

General Properties of Modes

The various IOS modes described herein can conveniently be viewed as being arranged in mode tree-structures. The major mode classifications can be regarded as the root modes leading to submode branches. In turn these submodes may have submodes. At each level the submodes may or may not be mutually exclusive. The submodes at a final level are always mutually exclusive.

An important property at every level where the submodes are mutually exclusive is that one of these submodes is defined as the default mode, in the absence of a user attempt to specify a choice. In all cases, applicable modes are set to valid user-requested submodes or to default submodes at attachment time.

When a device attachment is made (see Section BF.1.01) the Access, Data, Use, Code (if applicable to the device), and Synchronization modes are set. Except for the Use mode and the Workspace synchronization mode, these modes have the property of being device-oriented. Specifically, they are associated with the device node described in Section BF.1.01, and more specifically they are associated with the primary device node defined in Section BF.1.03.

When an indirect-frame - ioname attachment is made (see Section BF.1.01) only the Use mode and the Workspace synchronization mode are set. These modes have the property of being ioname-oriented. Specifically, they are associated with a node on the attachment graph (see Section BF.1.03). The Workspace synchronization mode has the additional property of propagating down the interior portion of iopath leading from the node whenever it is changed (see Sections BF.1.03 and BF.1.04).

Another important property of any IOS mode is whether or not it is changeable, once set. The Data mode cannot be changed once set, and the Access mode cannot be changed once actual input/output has begun. The Use, Code, and Synchronization modes can be changed provided the newly-requested submode is valid.

The Access Mode

This mode is related to the positioning properties of input/output devices. An essential characteristic of the positionability of a device is its dimensionality. Further, each positioning coordinate may or may not be bidirectional. However, for all current and foreseeable multidimensional input/output devices, the positioning along each coordinate is bidirectional or periodic, and such devices are hereby characterized as Random.

Examples of random devices include disks and drums. Devices with one-dimensional positioning are characterized as Sequential. Examples are ordinary magnetic tape and unit record equipment. Sequential devices are either Forward-only or Backspaceable. Examples of the former are printer, card, typewriter, and most paper tape equipment. Examples of the latter include magnetic tape and some paper tape equipment.

In summary, the Access mode tree is:

```
Access
  Random
  Sequential
    Forward-only
    Backspaceable
```

All submode levels contain mutually exclusive submodes.

The Access mode is set at device attachment time, and is associated with the device node. The default Access submode is Sequential, and the default Sequential submode is Forward-only. Any Access mode within the capability of the attached device can be requested. If the requested Access mode is more restrictive than the device capability, the mode is so set and subsequent access so restricted. Sequential is regarded as more restrictive than Random, and Forward-only more than Backspaceable. The Access mode can be changed only if no actual input/output has begun as a result of a read, write, seek, tell, or order call; after input/output is in progress, the Access mode cannot be changed. If the device is detached and reattached, a different valid Access can be set.

The Access mode is summarized in Table 1. Device access capabilities are listed in Table 3.

The Data Mode

The Data mode affects the relationship between the data transmitted to or from the user's workspace and the physical data transmitted to or from an actual device. Under almost all ordinary circumstances the IOS processes the data flowing through it. Examples of such processing include code conversion, standard physical data formatting and blocking, and interpolation of function-related time delays during output. The Data submodes are Physical and Logical. The default Data submode is Logical.

When the Data submode is Physical, the user has complete control over data transmission to and from the physical device. That is, the IOS behaves as a transparent transporter of the data. Further, the user can issue order calls that affect hardware operation modes and effect hardware functions. Examples are setting magnetic tape density and writing a physical end-of-file on magnetic tape, respectively. The Physical Data submode is not ordinarily used; it is provided for users who must interchange demountable media with the outside world. The Physical Data submode is implemented for only a few devices; see Table 3.

When the Data submode is Logical, the IOS processes the data flowing through it in some standard way. The extent of this processing is device-dependent. Certain aspects of this processing are under a degree of user control. The Logical submodes are Linear and Sectional; the default submode is Linear. Linear and Sectional frames are described in detail in BF.1.08.

A Linear frame is a linear array of elements; an element is a linear array of bits. The degree of processing of linear data by the IOS varies considerably among devices. In the case of a File System file, the File System Interface Module (FSIM) does no processing whatsoever of linear data; that is, the data flowing between the user and the FSIM is the same data that the FSIM transmits to or from the actual file. In the case of linear data on Magnetic Tape, the Tape Device Interface Module (DIM) imposes a standard tape format (see Section BB.3.01); the physical data on the tape is recorded in standard-length records containing header and trailer information as well as the user's linear data. Such standard formatting is necessary for reliable tape operation with a uniform tape strategy; in addition, the standard format provides safe, unique tape reel identification. Similarly, the Data Disc DIM imposes a standard data disc format. In the case of typewriter input/output, the IOS not only accomplishes code conversion, but interpolates the control data transmission necessary to operate the typewriter. The foregoing typifies the range of IOS data processing; see the appropriate overview sections of Section BF for the discussion of the standard IOS processing for any particular device type.

A Sectional frame is a linear array of items; each item is composed of a Linear subframe and/or a Sectional subframe. This recursive structure interpretation is implemented by the Sectional Formatting Module (SFM), which is spliced between the user and the actual primary device node. Data transmission between the user and the SFM is conducted in terms of linear and/or sectional subframes, while the data transmission between the SFM and the primary device node is in terms of a single, ordinary Linear frame.

The Data mode tree is:

```
Data
  Physical
  Logical
    Linear
    Sectional
```

All submode levels contain mutually exclusive submodes.

The Data mode is set at device attachment time, and is associated with the device node. The default Data submode is Logical, and the default Logical submode is Linear. The Data mode cannot be changed after attachment time.

The Data mode is summarized in Table 1. Device - Data-mode relationships are listed in Table 3.

The Use Mode

The Use mode relates both to a device's ability to be read or written and to a user's desire to read or write a particular frame. At device attachment time, the requested Use mode is matched against the device's capability, and any request within that capability is honored. A Use mode request that is more restrictive than a device's capability (for example, an Unreadable request on a readable device) is regarded as a request to subsequently so restrict the device's use. If the Use mode is not explicitly provided in the mode argument of the attach call, the requested Use mode is implicitly determined by including appropriate default submodes. When an indirect-frame - ioname attachment is attempted, the requested Use mode is compared to a "capability" which is the current Use mode at the ioname node to which the indirect frame is to be associated.

The Use submodes are Read, Rewrite, and Append, and are not mutually exclusive. The Read mode relates to reading a frame, and the Rewrite and Append modes relate to writing a frame.

The mutually exclusive Read submodes are Readable and Unreadable. The former is the default submode if the device being attached is intrinsically readable or if the ioname being attached (in an indirect-frame - ioname attachment) has its Read mode set to Readable. The default submode is Unreadable if the device being attached is unreadable or if the ioname being attached has its Read mode set to Unreadable. A read call reaching a frame node whose Read mode is set to Unreadable will be rejected.

The Rewrite mode relates to rewriting (overwriting) previously-written data. The Append mode relates to writing onto an unwritten portion of a frame. On a Linear frame, the unwritten portion is that which follows the last item written. On a Sectional frame, unwritten items can also occur between written items.

The mutually exclusive Rewrite submodes are Rewritable and Unrewritable. The former is the default submode if the device being attached is intrinsically rewritable or truncatable or if the ioname being attached has its Rewrite mode set to Rewritable. The default submode is Unrewritable if the device being attached is not rewritable or if the ioname being attached has its Rewrite mode set to Unrewritable. A sequential, backspaceable device which is intrinsically unrewritable may be rewritable in the sense of being truncatable and appendable; for such devices Rewritable implies truncatable. An example is magnetic tape. The mutually exclusive Append submodes are Appendable and

Unappendable. The former is the default submode if the device being attached is intrinsically appendable or if the ioname being attached has its Append mode set to Appendable. The default submode is Unappendable if the device being attached is not intrinsically unappendable or if the ioname being attached has its Append mode set to Unappendable.

A write call reaching an indirect frame node is accepted and passed on if either the Rewritable or Appendable modes are set on. A write call reaching a direct frame node where the Unrewritable or the Unappendable mode is set is accepted only if it does not correspond to rewriting or appending, respectively.

The Use mode tree is:

```
Use
  Read
    Readable
    Unreadable
  Rewrite
    Rewritable
    Unrewritable
  Append
    Appendable
    Unappendable
```

The final submode levels are mutually exclusive.

The Use mode is set at attachment time and is associated with the frame node created. The Use mode can be changed by a changemode call, provided the newly-requested Use mode is within the device or ioname capability.

The Use mode is summarized in Table 1. Device - Use-mode capability relationships are indicated in Table 3.

The Code Mode

This mode concerns the element-by-element mapping performed by the IOS for character-oriented devices. Such mapping is necessary when the binary representation of a character on an input/output device differs from the internal representation in Multics. The mapping is not always one-to-one. In addition, character strings within Multics are kept in a standard canonical form (see Section BC.2.02). Further, certain devices are deficient with respect to the Multic standard (ASCII) character set, and the missing graphics (and sometimes other characters) need to be somehow input and output; such input/output is accomplished by using conventions using sequences of available characters (see Section BC.2.04). The processing required by these circumstances is called code conversion.

This code conversion is implemented by a Code Conversion Module (CCM) usually spliced between the user and the primary device node. One exception to this rule occurs when using the pseudoprinter pseudodevice to accomplish delayed printing; in this case code conversion is delayed until the actual printing is accomplished by the Line Printer Driver (see Section BF.5). Code conversion during delayed card punching is similarly delayed. Conversion during delayed input/output is generally delayed.

The Code submodes are Input and Output and are not mutually exclusive. They refer to input and output code conversion respectively. The mutually exclusive Input submodes are Raw and Canonical; the latter is the default Input submode. Canonical conversion includes: (1) binary element representation conversion; (2) canonical print position allignment; (3) interpretation of escape sequences; and (4) print position erase and kill processing. See Section BF.1.05 and Sections of BC.2 and BF.10. Raw conversion implies no conversion; the data delivered to the user is the same data originally collected by the IOS from the device. This mode is provided for the binary input case and as an escape hatch for unforeseen user applications.

The mutually exclusive Output submodes are Straight, Edited, and Normal; the last is the default Output submode. Straight conversion implies no conversion; it is provided for the same reason that Raw is provided, as well as for the binary output case. Edited and Normal are similar modes and the following discussion applies to both. The character data provided to the IOS by the user must be in ASCII form. The IOS classifies the characters provided by the user into four categories: (1) the character is a graphic which can and is to be printed on the device or is a control whose function the device can perform; (2) the character is to be deleted prior to output; (3) the character is to be printed as an appropriate escape sequence; or (4) the character is a graphic which is to be replaced by a blank prior to printing. In the Normal mode, no characters are placed in categories (2) or (4); in the Edited mode, no characters are placed in category (3). Thus the Normal mode is more unambiguous, but the Edited mode is more appropriate to documentation where escape sequences may not be desired. The classification for each mode is determined by device-dependent tables which drive the CCM. The driving tables for the Normal mode are fixed. Default driving tables are supplied for the Edit mode; however, the Edit mode tables can be altered by an appropriate order call (see Section BF.1.07). Again see Section BF.1.05 and Sections of BC.2 and BF.10.

The Code mode tree is:

```
Code
  Input
    Raw
    Canonical
  Output
```

Straight
Edited
Normal

Only the lowest levels are mutually exclusive.

The Code mode is set at device attachment time and is associated with the device node. Any code mode within the implementation capability for a particular device can be requested. The default submodes are Canonical and Normal. The Code mode can be changed by a *changemode* call. When this is done, the only read-ahead data lost is that unavoidably read-ahead by the CCM (see Section BF.1.04).

The Code mode is summarized in Table 1. The devices for which the Code mode is ordinarily meaningful are indicated in Table 3; the IOS automatically invokes a CCM for these devices. The user can, of course, overtly splice a CCM into any iopath. Whether or not this is meaningful in any particular case on any particular device depends primarily on the data contained on the device and/or on the user's intent. For example, one could splice a typewriter-oriented CCM in front of the FSIM and thereby write that CCM's output into a file.

The Attachment Mode

The Attachment submodes are *Regular* or *Emergency*. The former is the default Attachment mode. The Regular mode relates to the conventional attachment ordinarily used. The Emergency mode is recognized only at primary device nodes. Emergency attachment essentially requests that current input/output on the device be suspended so that a new, independent series of input/output transactions can be made; the independent input/output is ended and the previous input/output restarted by the advent of an emergency detachment. See Section BF.1.04 for a detailed explanation. An *attach* call requesting emergency attachment which reaches an indirect frame node is rejected.

The Synchronization Modes

See Section BF.1.04 for a complete discussion of these modes. They are summarized in Table 2. The read and write synchronization modes apply to *all* devices, and the workspace synchronization mode is associated with an ioname and the corresponding interior iopath (see BF.1.03). The synchronization modes are set by special calls; see Table 2 and Section BF.1.04.

General Examples

The call

```
call attach("x","tape","Reel_4231","",status)
```

will attach a tape reel 4231 as a Sequential, Forward-only, Linear frame, with the use mode set to Readable, Unrewritable, and Appendable. This corresponds to a mode = "Q/F/L/R/notW/A" (where "/" is used as the mode delimiter). As another example consider:

```
call attach("x","tape","Reel_4231","B/S",status1)

call attach("y",linear,"x","D",status2)
```

The first call will attach the same reel as a Sequential, Backspaceable, Sectional frame named "x", with the same Use modes as above. The second call attaches the linear component of the current item of "x" as a Random, Linear frame named "y", with Use modes equal to those of "x".

Special Device Modes

Certain special device-oriented modes may be set and/or changed by the use of the attach and/or changemode calls. These are listed in Table 4. The references included in Table 4 should be consulted for a detailed explanation of these modes.

Summary Tables

Several summary tables are included in this Section. Table 1 summarizes all but the synchronization and special device modes which are summarized in Tables 2 and 4 respectively. Tables 1 and 4 also list the mode mnemonics. Table 3 lists standard devices and pseudodevices along with their mode relationships.

Table 1

I/O System Modes Settable by attach and/or changemode Calls.

Modes	Mnemonics		(D)efault/ (C)hangeable	Attachment Graph Association
Access	-	-	-	(with device node)
Random	D	RAND	-	
Sequential	Q	SEQ	D	
Forward only	F	FOR	D	
Backspaceable	B	BACK	-	
Data	-	-	-	(with device node)
Physical	P	PHY	-	
Logical	G	LOG	D	
Linear	L	LIN	D	
Sectional	S	SECT	-	
Use*	-	-	*	(with any ioname)
Read	-	-	C	
Readable	R	READ	D**	
Unreadable	¬R	¬READ	-	
Rewrite	-	-	C	
Rewritable	W	WRITE	D***	
Unrewritable	¬W	¬WRITE	-	
Append	-	-	C	
Appendable	A	APP	D****	
Unappendable	¬A	¬APP	-	
Code*	-	-	*	(with device node)
Input	-	-	C	
Raw	-	RAW	-	
Canonical	C	CANON	D	
Output	-	-	C	
Straight	-	STR	-	
Edit	-	EDIT	-	
Normal	N	NORM	D	
Attachment	-	-	-	
Regular	-	-	D	(with primary device node)
Emergency	E	EMERG	-	

Asynchrony Prevention: see footnote Table 2.

*Immediate submodes not mutually exclusive.
**Unless directed to unreadable device.
***Unless directed to unwritable device.
****Unless directed to unappendable device.
"¬" = "not"; "-" indicates none or not relevent.

Table 2

Synchronization Modes (see Section BF.1.04).

Mode	Call to set mode	(D)efault/ (C)hangeable	Attachment Graph Association
Read Sync.	readsync	C*	(with device node)
Asynchronous		D	
Synchronous		-	
Write Sync.	writesync	C*	(with device node)
Asynchronous		D	
Synchronous		-	
Workspace Sync.	worksync	C	(with ioname and interior iopath)
Asynchronous		-	
Synchronous		D	

*These modes may be forced synchronous by including the asynchrony-prevention mnemonic "Y" in the mode argument of an attach and/or changemode call as a prefix to the appropriate "use" mode mnemonics. The inclusion of "notY" in the same context restores control of these modes. (See Section BF.1.04). For example, the inclusion of "YR" forces the read synchronization mode synchronous.

Table 3

Device-Mode Relationships.
Legend: D=Default; "X"=possible alternative;
".''=impossible alternative; "¬"="not";
"*"=dependent on actual physical device
attached as the primary Sectional frame.

	Modes:																
Devices or	Access			Data			Use						Code				
Pseudodevices	D	F	B	P	L	S	R	¬R	W	¬W	A	¬A	RAW	C	STR	EDIT	N
tapes	.	D	X	X	D	X	D	X	D	X	D	X	.	.	.	.	.
data disc	X	D	X	X	D	X	D	X	D	X	D	X	.	..	.	.	.
pseudoprinter	.	D	.	.	D	.	.	D	.	D	D	X	.	.	X	X	D
pseudocard-punch	.	D	.	.	D	.	.	D	.	D	D	X	.	.	X	X	D
pseudocard-reader	.	D	.	.	D	.	D	X	.	D	.	D	X	D	.	.	.
pseudopaper-tape-punch	.	D	.	X	D	.	.	D	.	D	D	X	.	.	X	X	D
pseudopaper-tape-' reader	.	D	.	X	D	.	D	X	.	D	.	D	X	D	.	.	.
File System file	X	D	X	.	D	X	D	X	D	X	D	X	.	.	.	.	.
typewriter	.	D	.	.	D	.	D	X	.	D	D	X	X	D	X	X	D
typewriter keyboard	.	D	.	.	D	.	D	X	.	D	.	D	X	D	.	.	.
typewriter printer	.	D	.	.	D	.	.	D	.	D	D	X	.	.	X	X	D
Linear subframe	X	D	X	.	D	X	*	*	*	*	*	*	.	.	.	.	.
Sectional subframe	X	D	X	.	.	D	*	*	*	*	*	*	.	.	.	.	.
PRT 202	.	D	.	.	D	.	.	D	.	D	D	X	.	.	.	X	D
CPZ 200	.	D	.	.	D	.	.	D	.	D	D	X	.	.	X	X	D
CRZ 200	.	D	.	.	D	.	D	X	.	D	.	D	X	D	.	.	.
BELL 103	.	D	.	.	D	.	D	X	.	D	D	X	.	.	.	.	.
BELL 201	.	D	.	.	D	.	D	X	.	D	D	X	.	.	.	.	.
BELL 202	.	D	.	.	D	.	D	X	.	D	D	X	.	.	.	.	.
BELL 301	.	D	.	.	D	.	D	X	.	D	D	X	.	.	.	.	.
BELL 801	.	D	.	.	D	.	.	D	.	D	D	X	.	.	.	.	.

Table 4

Special Device Modes Settable by *attach* and/or *changemode* Calls.

Special Device Modes	Mnemonics		(D)efault/ (C)hangeable	Device Associated	Reference
Duplex	-	-	C on M37	Typewriter	BF.11
Half-duplex	H	HDX	D on 1050/2741		
Full-duplex	X	FDX	D on M37		
Echoplex	-	-	C on M37	(Typewriter channel)	BF.11
Echoplex-on	-	ECHO	D with FDX		
Echoplex-off	-	NOECHO	D on 1050/2741		