INTERNAL USE ONLY

INTERNATIONAL COMPUTERS AND TABULATORS LIMITED

Scientific Programming Department
I.C.T. 1900 Series

FORTRAN NOTE 4
12.4.65.

## FØRTRAN IV (ICT) Input and Output Operations

In FØRTRAN IV the following types of statement are available to the programmer

```
READ (i,n) list
READ (i) list
WRITE (i,n) list
WRITE (i) list
BACKSPACE i
REWIND i
ENDFILE i
```

where i is the "Peripheral Value" (P.V.) and may be either a small integer (<4096) or an integer variable reference. "n" is either an integer which is the label of a FØRMAT statement or is an Array name in which case the Array itself is taken as being a FØRMAT specification.

By association with a peripheral type statement in the Program Description the Peripheral Value at the time the I/Ø statement is initialized determines the particular peripheral device which is to be used for that operation.

Thus ifiin the Program Description there is a statement

```
INPUT 10 = TR0
```

and in the body of the program there is a statement

```
READ (10,105) A,B,I
```

The input operation specified would use the primary tape reader. The following two statements would also have the same effect.

```
I = 10
READ (I,105) A,B,I
```

Input and Output Operations - to the Compiler

On recognizing any I/Ø statement, the Compiler generates cues to the private library routine %FINØUT which performs the actual peripheral operations.

For all entries to %FINØUT, at the time the call to %FINØUT is obeyed, X3 and X6 will give the parameters required. The actual entries (and entry requirements) are as follows.

| Entry | Operation | X3 | X6 |
|---|---|---|---|
| %FINØUT + 2 | ENDFILE | | P.V. |
| %FINØUT + 4 | Initiate WRITE | address of FØRMAT (0 if no FØRMAT) | P.V. |
| %FINØUT + 6 | Initiate READ | address of FORMAT (0 if no FØRMAT) | P.V. |
| %FINØUT + 8 | Read or Write Integer Item | address of Item | - |
| %FINØUT + 10 | Read or Write Real Item | address of Item | - |
| %FINØUT + 12 | Read or Write double Precision Item | address of Item | - |
| %FINØUT + 14 | Read or Write Complex Item | address of Item | - |
| %FINØUT + 16 | Read or Write Logical Item | address of Item | - |
| %FINØUT + 18 | Terminate READ or WRITE | - | - |
| %FINØUT + 20 | BACKSPACE | - | P.V. |
| %FINØUT + 22 | REWIND | - | P.V. |

In addition to those entries to %FINOUT, there are also the following four entry points which are used in associated PLAN compiled subroutines for the operations "Runout" (paper tape), "Disengage", "Release" and "Allocate". In each case X6 = P.V.

| | |
|---|---|
| %FINØUT + 0 | Runout |
| %FINØUT + 24 | Disengage |
| %FINØUT + 26 | Release |
| %FINØUT + 28 | Allocate |

The list of a READ or WRITE statement may include an Array name, in which case it is required to transfer all the elements of the array. When this situation occurs, the compiler generates a call to a private library routine %FIØTA which processes the whole array.

%FIØTA has the following entries.

| | |
|---|---|
| %FIØTA + 0 | Process Integer Array |
| %FIØTA + 2 | Process Real Array |
| %FIØTA + 4 | Process Double Precision Array |
| %FIØTA + 6 | Process Complex Array |
| %FIØTA + 8 | Process Logical Array |

On entry to the routine %FIØTA, X3 contains the address of the "Array Header" for the particular array.

As an example of the use of %FINØUT and %FIØTA note the results of compiling the following statement.

```
READ (10,105) A,B,I
```

where B is an array name. A and B are both "Real", I is "Integer".

```
LDX   3       a            a contains address of Format labelled 105
LDN   6       10           X6 = P.V.
CALL  1    %FINØUT + 6     Initiate "Read" operation
LDN   3       A            X3 = Address of A
CALL  1    %FINØUT + 10    Process "Real" Item
LDN   3       b            b = 1st word of Array Header for B
CALL  1    %FIØTA + 2      Process the entire array B
LDN   3       I            X3 = Address of I
CALL  1    %FINØUT + 8     Process "Integer" Item
CALL  1    %FINØUT + 18    Terminate  "Read" operation
```

For the statement ENDFILE J, the following is compiled

```
LDX   6       J            X6 = P.V.
CALL  1    %FINØUT + 2     Do "Endfile" operation
```

V.K. Taylor