Identification

Macro Expander
B. Wolman

(Note that the following are Abstracts, which should be replaced by a full description at a later time.)

EXPMAC

Function of Entry:

> Temporary version of macro expander for output code generator of the PL/I compiler. This segment of use only to the PL/I compiler.

Calling Sequence for Entry:

```
call   expmac$many (n, arg_pt, arg_cnt)
call   expmac$one (n, arg)
call   expmac$abs (blk, blk_size)
call   expmac$oneabs (word)
call   expmac$origin (org)
call   expmac$cur_loc (loc)
call   expmac$skip (n)
```

Declaration of Arguments:

```
dcl   (arg_cnt, blk_size, n) fixed bin(15),
      (org, loc) fixed bin(18),
       word        fixed bin(36),
      (arg_pt, arg, blk) ptr;
```

Description of Arguments:

expmac$many expands macro n with arg_cnt arguments stored in the array pointed at by arg_pt. Each element of the array is a pointer to an argument node.

expmac$one expands macro n with a single argument.

expmac$abs copies a block of blk_size words pointed at by blk into the output.

expmac$oneabs appends a single word.

expmac$origin sets the loading origin to org.

expmac$cur_loc sets loc to the current loading origin.

expmac$skip causes n words to be filled with zeros.

PUT_OUTPUT

Function of Entry:

> Temporary output processor for PL/I output code
> macro expander (expmac).

> Each call of put_output causes current output
> buffer to be printed in the error file.

> This segment of use only to PL/I compiler.

Calling Sequence for Entry:

> call put_output

> call put_output$initialize

Declaration of Arguments:

> none

Description of Arguments:

> a single call of put_output$initialize must be made
> before any calls of put_output.