

Some Conventions for Unshared Subroutines Capable
of Operating in the SUPERVISED MODE

1. When control enters and leaves the subroutine, the interrupt system is enabled. Any disabling which occurs within the routine stems from macro-ops which also provide for re-enabling.
2. Index register XI is used for depositing the IC on entry and for indexing the return branch.
3. All other index registers (including X0), which are used by the subroutine, are restored by it.
4. The arithmetic registers (8-13) are used freely with no attempt at preservation or restoration.
5. No matter what changes the subroutine may make in the contents of the indicator register, the original contents at time of entry need not be preserved or restored.
6. If the subroutine is capable of altering the contents of the mask register in any way, a copy of the mask, which is effective upon entry, is preserved in memory and is restored just prior to exit.
7. If the subroutine uses indicator k on an interrupt basis and requires an interrupt procedure other than the one in use by the higher level program, it copies entry k from the PP interrupt table, substitutes its own entry and later restores the preserved entry.

Conventions Dropped Because of Burden on Object Program

1. Indicators LC through RU, mask bits LC through RU, and indicator NM from the higher level program are copied by the subroutine upon entry into memory.
2. Let indicator k be any indicator in the set LC through RU plus NM. Immediately prior to executing a "normal return" to the higher level program, the subroutine restores indicator k and mask bit k UNLESS indicator k is output from the subroutine to the higher level program. (Note: in this case the mask bit is zero).

Jv

EMcd

EFC

September 30, 1958

[Note: k is one of LC-RU]

SUBROUTINE ACTIVITY

CONDITION k

Guaranteed
NOT to generate
condition k

Generates k
under
certain conditions

Generation of k by subroutine
is information required by:

MAIN PROG.	SUBROUTINE
No	No
No	Yes
Yes	No
Yes	Yes

AND

Main Prog. requires INITIAL status
of condition k to be saved

INDICATOR	MARK
No	No
No	Yes
Yes	No
Yes	Yes

Macro-Ops for Restoring Indicators

A problem programmer may need to restore any or all of indicators LC through RU and also NM to some former status. If he is writing a program to operate in the supervised mode, there is no direct way in which he can program this restoration because he is not allowed to use the BRANCH DISABLE operation and, as long as the interrupt system is enabled, the indicator register is store-protected.

One solution to this problem is to make the PP call in the SP to do the restoring. This amounts to using an interpretive macro-op. However, a much more satisfactory solution is to use a compiled macro-op. Two such macro-ops are defined below: one for restoring a few specified indicator positions belonging to the set LC through RU plus NM, the other for restoring all indicator positions belonging to this set except those specified.

REINDICATE

Example: REINDICATE ALPHA XPH , XPN, NM

ALPHA is the name of a full word memory location into which a copy of the indicator register contents has been placed. The macro-op causes the XPH, XPN, NM configuration in this copy to be restored into the indicator register. The compiler executes the following steps:

1. A 64-bit word W is manufactured which consists of zero bits except for the XPH, XPN, and NM positions which are occupied by ones. A check is made to see that no attempt is being made to restore any indicators other than LC through RU plus NM.
2. A sequence of instructions is developed as part of the object program:

1) CONNECT	LOAD PARTIAL	Places W in accumulator
2) CONNECT TO MEMORY	AND	Replaces ALPHA by W.
3) CONNECT	AND NOT	Forms I.W in accumulator
4) BRANCH DISABLE		Makes indicator-register accessible
5) CONNECT TO MEMORY	OR	Places W + I.W in indicator reg.
6) BRANCH ENABLE		Re-enables interrupt system

W is 64-bit word manufactured by compiler

I is contents of indicator register

α is contents of location ALPHA.

REINDICATE EXCEPT - operates in a similar way except that the 64-bit word W contains zeros in positions CTK through IF and TF through H, zeros in all specified positions, and ones in all remaining positions.

EMcD
EFC
September 30, 1958

jv