ASCHK$ - Check for Duplicate Assignments

A subtle problem with the Extended Unit Assignment Tables can occur if applications linked with the pre-V8.1 ASSIG$ routine are used with a GSM V8.1 that has been configured with 1, or more, extra ASSIG$ tables. If the Primary ASSIG$ Table is full then when an application linked with the V8.1 ASSIG$ routine (e.g. the standard Menu Handler) adds a new Unit Assignment (e.g. NLD) that entry will be added into the Extended Table(s). If one, or more units, are then removed from the Primary Table thus creating spare slots in the Primary Table when an application linked with the pre-V8.1 ASSIG$ attempts to add a new Unit Assignment (e.g. NLD) the situation can arise where a Unit Assignment appears in both the Primary Assignment Table and an Extended Assignment Table.

The ASCHK$ routine can be used to check if a Logical Unit Assignment appears in both the Primary Assignment Table and an Extended Assignment Table(s).

1. Invocation

To check if a Logical Unit Assignment appears in both the Primary Assignment Table and an Extended Assignment Table(s) code:

        CALL ASCHK$ USING unit

where unit is a PIC X(3) containing the Logical Unit to be tested (e.g. "NLD"). If an exception is returned (see below) the Physical Unit of the first occurrence in an Extended Table will be returned in unit, so this parameter should not be a 3-character literal.

2. STOP Codes and Exception Conditions

No STOP codes are generated by ASCHK$.

The following EXIT codes may be returned by ASCHK$:

EXIT code	$$COND	Description
10807	7	The Logical Unit Assignment appears in both the Primary Assignment Table and an Extended Assignment Table(s). The Physical Unit of the first occurrence in an Extended Table is returned in the unit parameter.

3. Programming Notes

ASCHK$ is only available for use with GSM SP-14, or later. Programming Notes. It is reserved for use by the 32-bit Menu Handler.

4. Examples

[EXAMPLE REQUIRED]

5. Copy-Books

None.

6. See Also

None.