APPENDIX D : STANDARD ASSEMBLER SUBROUTINES

A number of Standard Assembler subroutines are held in the system library and may be called from CREDIT programs.

EMPTY *Empty Test* **EMPTY**

Syntax: [statement-identifier] ␣ CALL ␣ EMPTYT, data-item-identifier

Description: The data-item-referenced by data-item-identifier is tested for an empty value. Data item types binary, decimal and string are allowed.
The following values are considered as empty:
a) Binary-data-item, zero (0)
b) Decimal-data-item, all spaces (X'F')
c) String-data-item, all null characters (X'00')

Condition register: = 0 not empty
≠ 0 empty

Example: CALL EMPTYT, DEC2

Intermediate code format:

Byte 1	0	0	1	1	0	0	0	0
Byte 2	external reference							
operand-1	data-item-identifier							

Bytes 1 and 2 are filled by the system
Byte 2 contains a reference to an external system routine.
Operand-1 is a reference to a binary, decimal or string data item.

GETCW *Get control word* **GETCW**

Syntax : [statement-identifier] ␣ CALL GETCW, data-set-identifier, data-item-identifier

Description : The control word of a data set indicated by data-set-identifier is stored in a binary data item indicated by data-item-identifier.

Condition register : unchanged

Example : CALL GETCW, DSDK, CONTRW

Intermediate code format :

Byte 1	0	0	1	1	0	0	0	0
Byte 2	external reference							
operand-1	data-set-identifier							
operand-2	data-item-identifier							

Byte 1 contains the operation code (X'30')
Byte 2 is a reference to an external system routine
Operand-1 is a reference to a data set
Operand-2 is a reference to a binary data item

FMOVE *Format Move* **FMOVE**

Syntax: [statement-identifier] CALL FMOVE.
data-item-identifier, format-list-identifier

Description: The format-list referenced by format-list-identifier is copied into the string-data-item referenced by data-item-identifier. If the string-data-item is longer than the format list, remaining bytes are filled with the TEXT format-list-item.

Condition register: Unchanged.

Example: CALL FMOVE, STR1, FORM5

Intermediate object code:

Byte 1	0	0	1	1	0	0	0	0
Byte 2	external reference							
operand-1	data-item-identifier							
operand-2	format-list-identifier							

Bytes 1 and 2 are filled by the system
operand-1 is a reference to a data set.
operand-2 is a reference to a format list.

ICLEAR *Clear data item* **ICLEAR**

Syntax: [statement-identifier] ␣ CALL ␣ ICLEAR, data-item-identifier

Description: The data item referenced by data-item-identifier will be cleared. Data-item types binary, decimal or string are allowed. Clearing of a data item will result in:

a) Binary-data-item is set to zero.
b) Decimal-data-item is set to all spaces (X'F')
c) String-data-item is set to null characters (X'00')

Condition register: Unchanged

Example: CALL ICLEAR, DEC5

Intermediate code format:

Byte 1	0	0	1	1	0	0	0	0
Byte 2	external reference							
operand-1	data-item-identifier							

Bytes 1 and 2 are filled by the system
Byte 2 contains a reference to an external system routine.
Operand-1 is a reference to a binary, decimal or string data item.

MASK *Mask function* **MASK**

Syntax : [statement-identifier] ⊔ CALL MASK, data-item-identifier-1, data-item-identifier-2.

Description : From the two binary data items, indicated by data-item-identifier-1 and data-item-identifier-2, the logical product is taken and compared to zero. The result is stored in the condition register. This function is useful after a XSTAT instruction to examine the device dependent status.
The contents of the binary data items are not changed.

Condition register :

= 0 if result is zero
= 1 if result is positive
= 2 if result is negative

Condition mask :

0	1	2	3	4	5	6	7
= 0	> 0	< 0	—	≠ 0	⩽ 0	⩾ 0	Unconditional

Example : CALL MASK, MK1, STATUS

Intermediate code format :

Byte 1	0 0 1 1	0 0 0 0
Byte 2	external reference	
operand-1	data-item-identifier-1	
operand-2	data-item-identifier-2	

Byte 1 contains the operation code (X'30')
Byte 2 is a reference to an external system routine
Operands-1,2 are references to binary data items

TYPET *Type Test* **TYPET**

Syntax: [statement-identifier] CALL TYPET, data-item-identifier-1, data-item-identifier-2

Description: Data-item-type of the data-item referenced by data-item-identifier-2 is tested and type is returned in the binary data-item referenced by data-item-identifier-1.

Following values are returned:

a) 1 for binary-data-item
b) 2 for decimal-data-item
c) 3 for string-data-item.

Condition register: Unchanged.

Example: CALL TYPET, BIN1, BIN2

Intermediate code format:

Byte 1	0	0	1	1	0	0	0	0
Byte 2	external reference							
operand-1	data-item-identifier-1							
operand-2	data-item-identifier-2							

Bytes 1 and 2 are filled by the system
Byte 2 contains a reference to a system routine
Operand-1 is a reference to a binary data item
Operand-2 is a reference to a binary, decimal or string data item.