RC PROCEDURES FOR GENT, PSBLK, PSWD, SUBTRAN (THROUGH 1E5 GENERIC PROGRAM) 2-WIRE NO. 1 ELECTRONIC SWITCHING SYSTEM

CONTENTS PAGE

1. GENERAL

1.01 This section covers recent change (RC) messages PSWD, GENT, PSBLK, and

SUBTRAN, through the 1E5 generic program, for a 2-wire No. 1 Electronic Switching System (ESS).

1.02 This section is reissued for the following reasons:

(a) To add information on YES/NO keywords and keyword combinations, in change messages.

(b) To delete information on generics prior to 1E3.

(c) To add to the procedures for linking a subtranslator to a head table, for subtranslators other than a directory number subtranslator.

(d) To make changes to the RC:GENT flowchart.

(e) To make changes to Table C.

(f) To make minor changes as required.

(g) Add list of abbreviations (Part 9).

Since this reissue covers a general revision, arrows ordinarily used to indicate changes have been omitted.

1.03 In change message flowcharts, keywords without a variable shown are YES/NO keywords. When a YES/NO feature is added, enter the keyword; when a YES/NO feature is removed, enter the keyword followed by NO or N.

1.04 When using a change message flowchart, refer to the associated new message flowchart for valid combinations of keywords.

1.05 Abbreviations used in this section are listed in Part 9.

2. MISCELLANEOUS RC INFORMATION

REFERENCES

2.01 The following applies to all RC messages in this section.

- Refer to Section 231-118-321 for information on RC message formats and the interpretation of message flowcharts.
- Refer to the information accompanying the message flowcharts for definitions of keywords used in the messages.
- The order number can be entered as an option in any message by typing in the following with the RC message:

ORD m̸nnnnnn̸n

m̸ = Letter. This variable is shown slashed because it need not be present.

nnnnnn̸n = Decimal number. Leading zeros can be omitted.

For example, the following are valid order numbers.

ORD B123456
ORD F6
ORD 23

- Refer to Section 231-118-320 for RC message program listings, system acknowledgment, and RC18 and RC16 output messages.
- Refer to Translation Guide TG-1A for documentation of translation data and associated ESS Forms.
- Refer to PA-591003—Translation Output Configuration to relate the translation input forms (ESS 1100, 1200, 1300, 1400, 1500 and 1600 series of forms) and the ESS translation memory (translators). It describes the format and content of each translator.

FLOWCHART SYMBOLS

2.02 The following flowchart symbols are used in RC message flowcharts.

◯ ***OPTION Symbol:*** The OPTION symbol is used to indicate that all flowlines leaving the symbol are optional. None, one, some, or all such flowlines may be selected.

⊗ ***EXCLUSIVE OR Symbol:*** The EXCLUSIVE OR symbol is used to indicate that exactly one of two or more flowlines leaving the symbol must be selected.

⊕ ***NON-EXCLUSIVE OR Symbol:*** The NON-EXCLUSIVE OR symbol is used to indicate that one or more of the flowlines leaving the symbol must be selected (no less than one, but more than one may be selected).

● ***AND Symbol:*** The AND symbol is used to indicate that all flowlines leaving the symbol must be used.

┌─┐
│ %
└─┘ ***Repeatable Segment:*** The repeatable segment symbol is used to indicate that the keyword units within the segment bracket can be repeated within an RC message without reentering previous keyword units. Each segment is terminated by the percent sign (%).

DETERMINING STARTING ADDRESS OF THE BASE TRANSLATION BLOCK (MASTER HEAD TABLE)

2.03 The master head table is the first 30 words in the base translation block. The base translation block is 530 words long. The address may be determined as follows by reading an address in parameters.

T-READ 1105615 01.

System response TW02 message contains the octal starting address of the master head table. Record this address for future reference.

DETERMINING STARTING ADDRESS OF MASTER HEAD TABLE ANNEX

2.04 The master head table annex is not located at a fixed address. The address may be determined as follows by typing in

T-READ aaaaaaa 01.

aaaaaaa = Starting address of master head (see paragraph 2.03) plus octal 33.

System response TW02 message contains the octal starting address of the master head table annex.

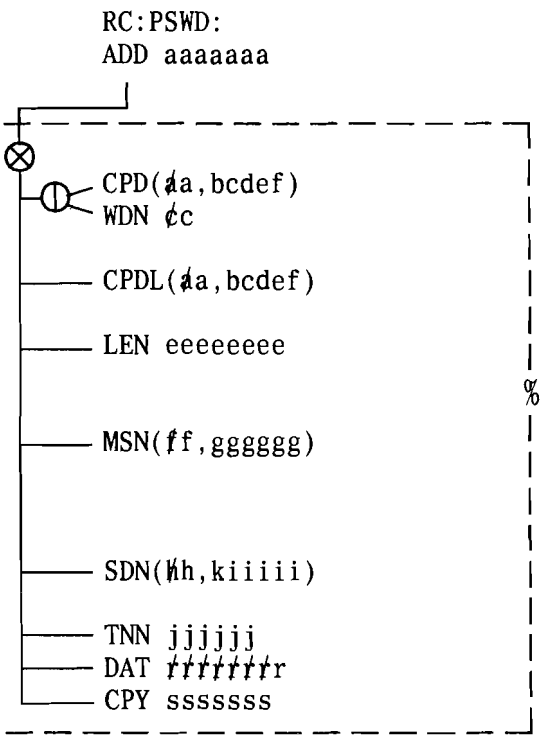
3. CHANGING A PROGRAM STORE WORD RC:PSWD (PR-1A344)

Caution: Because of the generality of this message, few program checks can be made. The user should be sure that the address entered is the one intended. A specific RC message should be used if available. (See Section 231-118-321.)

Writing One or More Consecutive Right-Half 23-Bit Words

INITIAL CONDITION: See ***Caution.***

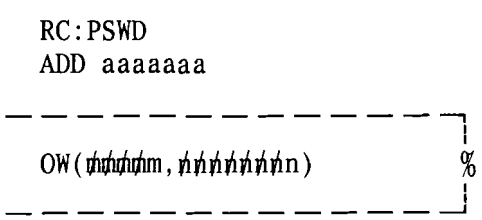
RESULTS OF MESSAGE: The data word specified by the keyword in the first segment replaces the contents of the right-half word of the address specified by ADD. For each succeeding segment, the system adds 1 to the last address used and replaces the contents of that data with the word specified in that segment.



Writing One or More Consecutive Full 37-Bit Words

INITIAL CONDITION: See ***Caution***, Page 3.

RESULTS OF MESSAGE: The data words (left 14 bits, right 23 bits) specified by keyword OW in the segment replaces the contents of the left and right parts of the full 37-bit word at the address (right half) specified by ADD (using two primary RCs). For each succeeding segment, the system adds 1 to the last address used and replaces the contents of that full word with the left and right words specified in that segment.

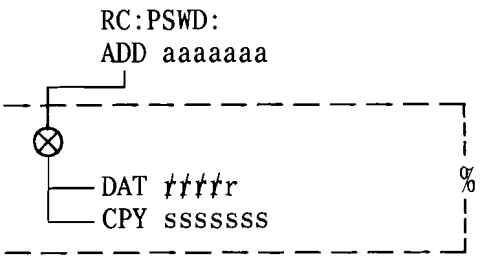


Writing One or More Consecutive Left-Half 14-Bit Words

INITIAL CONDITION: See ***Caution***, Page 3.

RESULTS OF MESSAGE: The data word specified in the first segment (rrrrr or contents of sssssss) replaces the contents of the left half word of the address specified by ADD. For each succeeding segment, the system adds 1 to the last address used and replaces the contents of that left-half word with the data specified in that segment.

Implementation note: Each even address causes two RC registers to be assigned. The first, on the specified even address, contains the specified data. The second on the following odd address, contains the ***existing*** contents of the word. This is necessary because the cardwriting program assumes the unusual case of a 23-bit pair of left-half words when an even left-half address is given and would overwrite the following word if there were no recent change on it specifying, in effect, "no change". A subsequent segment or message changing the odd-address word would overwrite the second RC register, of course.

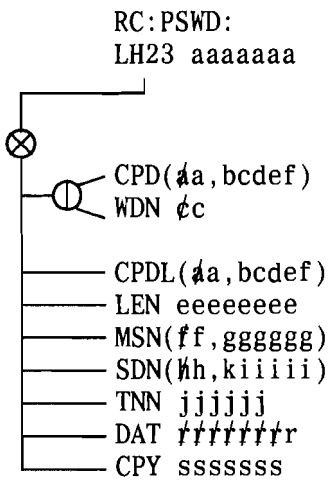


Writing a 23-Bit Left-Half Word Pair to 0 or a Value Whose 5 Most Significant Bits Are Not All Zero

INITIAL CONDITION: See ***Caution***, Page 3.

RESULTS OF MESSAGE: The data specified replaces the contents of two consecutive left-half words at the even address specified by LH23 and the following odd address.

Implementation note: Only one RC register is assigned. The cardwriting program breaks the data into two pieces for the two addresses.



Caution: Note that this form of the message should not be segmented. Presently, no program check is made, but translations will be built incorrectly if % is used.

Note: To write a 23-bit left-half word pair to a value not all 0 but with the 5 most significant bits 0, the two left-half words must be specified separately. Use the ADD keyword with left-half addresses covered previously in ***writing one or more consecutive left-half 14-bit codes.***

KEYWORD UNIT	DEFINITION
ADD aaaaaaa	Octal address of first PS word to be written. This address is used for the first segment. For each succeeding segment (if more than one is used), the system adds 1 to the address last used. The right-half address (bit 20 equal to 0) is used for a right-half 23-bit word and for a full 37-bit word.
CPD(¢a,bcdef)	Central Pulse Distributor Point and Quantity in auxiliary block word 0 ¢a = Quantity (decimal) in bits 17-14 bcdef = CPD Point in bits 13-0 (b = Pair, c = Half, d = Group, e = Row, f = Column)
CPDL(¢a,bcdef)	Central Pulse Distributor Point and left adjusted quantity in auxiliary block words other than 0 ¢a = Quantity (decimal) in bit 22-18 bcdef = CPD Point (decimal) in bits 13-0 (b = Pair, c = Half, d = Group, e = Row, f = Column
CPY sssssss	Copy. Contents of the word at this address is copied into the word being written. If CPY sssssss is specified, the contents of the word at address sssssss is used to write (copy).
DAT ¢¢¢¢¢¢	Octal Data. *Note:* If the value to be written is too large for the location into which it is to be written, the leftover upper bits are lost. If the value to be written is smaller than the location which it is to be written into, the remaining bits are made zeros.
LEN eeeeeee	Line Equipment Number
LH23 aaaaaaa	Left-Half Even Address for Left-Half 23-Bit Word-Pairs (must include a 1 in bit 20)
MSN (ff,gggggg)	Master Scanner Number and Quantity. ff = Quantity or Word Number (decimal); gggggg = Master Scanner Number.
OW(¢¢¢¢¢¢m,¢¢¢¢¢¢¢n)	Overwrite Data (See Note listed in Keyword DAT) ¢¢¢¢¢¢m = Octal Value of Bits 36 through 23 ¢¢¢¢¢¢¢n = Octal Value of Bits 22 through 0
SDN(¢h,kiiiii)	Signal Distributor Number and Quantity ¢h = Quantity (decimal) k = Frame Type (U,M,J, or S) iiiii = Signal Distributor Number
TNN jjjjjj	Trunk Network Number
WDN ¢c	Word Number (decimal)

Verification and Hunting Information

If the word being changed is RC hunted, the change is effective immediately and may be verified. Others are effective only after PS memory cards have been updated.

4. CHANGING A PRIMARY TRANSLATION WORD OR AUXILIARY BLOCK (1E4 AND LATER GENERICS) RC:GENT (PR-1A378)

4.01 The general translator (RC:GENT) message is available in 1E4 and later generics. Since the RC:GENT message is not tailored to a particular translator, normal checks and safeguards against improper use cannot be made by the program. Thus, input values should be checked carefully and, unless the message is used as a part of standard procedures specified for a particular application, it must be used with the same caution and restraint as the RC:PSWD message.

4.02 An RC:GENT message can be used to change a primary translation word (PTW), build a new auxiliary block, or change an existing auxiliary block (including its size). The NEW form of the message (RC:GENT;NEW: or, by default, simply RC:GENT:) should be used when the existing PTW is known to be all zeros, and the change (CHG) form (RC:GENT;CHG:) if it is not all zeros. To zero a PTW (and release any auxiliary block it may point to), use the OUT form of the message. A NEW message is rejected if the existing PTW is not all zeros, and a CHG or OUT message is rejected if it is all zeros.

4.03 In all cases (NEW, CHG, or OUT form), when an existing auxiliary block must be returned to the idle link list, it is automatically done by the message. This includes a change of auxiliary block size, since that is accomplished by returning the old auxiliary block and seizing a new auxiliary block. Examples of zeroing a primary translation word and building or changing an auxiliary block are given in Figures 1 and 2 respectively.

RC:GENT Definitions:

4.04 The ***segment-specified data*** for any word referred to in an RC:GENT message is the data obtained from the one or more data keywords used in a single segment of a message (the whole message if it is not segmented). It is formed as follows: the data for each keyword is located in its proper bit range, with zeros outside the range of the data, and then these words are all added together. For 1E5 and later generics, overlapping of item ranges within one segment for any keywords except DD and OD will cause rejection; keywords DD and OD may overlap each other and/or any other keyword. For earlier generics, no overlap checks are made.

4.05 The ***newly specified word*** is the result of insertion of the ***segment-specified data*** into the ***base word.*** Because of the unique way that segmenting words in the RC:GENT message, the ***newly specified word*** becomes an actual word only when and if the message is completed by an exclamation point and accepted. Therefore, unlike other segmented messages, if a segment of the RC:GENT message is rejected, no earlier segments of the same message become effective.

For 1E5 and Later Generics:

4.06 The ***base word*** is the effective word on which a segment of a message (or an entire message if not segmented) operates. At the end of any segment, the ***base word*** for any following segment (that refers in the same message to the same translation word) is the ***newly specified word.*** At the beginning of a message, the ***base word*** for a NEW-type message (RC:GENT:) is zero, and for a CHG-type message (RC:GENT;CHG:) is the existing translation word (whether in program stores or the recent change area).

For Generics Earlier than 1E5:

4.07 The ***base word*** is always zero for both NEW and CHG forms of the message. Thus, only one segment can be used per message to build or modify a single word. If more than one segment refers to the same word, only the last has any effect.

For All Generics:

Insertion into the base word means that those bits of the base word not covered by any of the ***ranges*** of the specified data are preserved. For example, if the base word was octal 5031, and OI (2, 5-3) and DI (4, 2-0) are inputted, the resulting word is octal 5024. Then, if OI (7, 10-6) is inputted via another segment or message, the resulting word becomes 4724. Note that bits 10 and 9 of the octal 5 are zeroed by the range of the item, changing octal 5 to octal 4.

Observe that keywords DD or OD cover all 23 bits, so no part of the ***base word*** is preserved when either of these keywords are used. For example, to write 7 into bits 10-8 and zero the remainder of a word with unknown (nonzero) contents, the keyword units OI (7, 10-8) and OD 0 can be used. An equivalent pair of keyword units are OI (7, 22-8) and DI (0, 7-0).

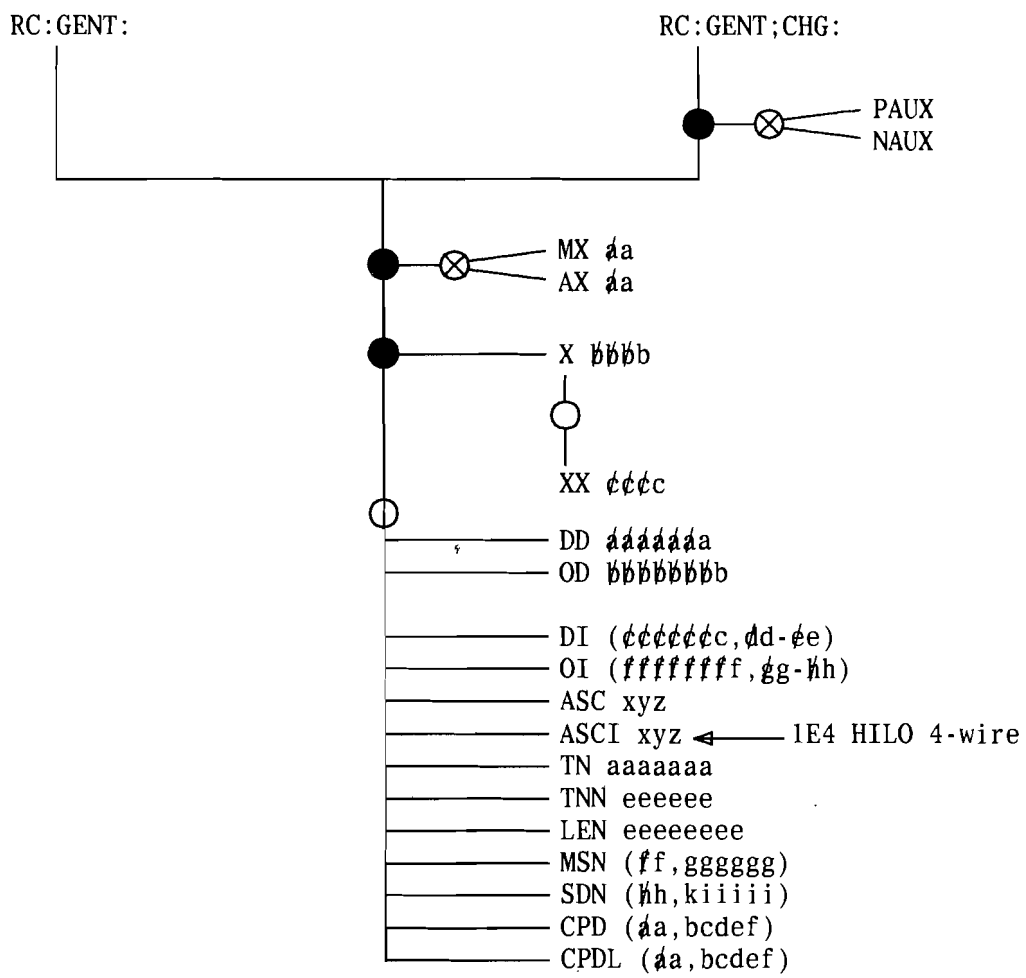
The ***range*** of the data is given with each keyword definition and does not vary with the value of the data specified. For example, the ***range*** for keyword DD is 23 bits (the entire word), even though the maximum value of its data uses only 20 bits. This means that if DD data is "inserted" into the base word, no previous bits in the ***base word*** are preserved.

Building, Changing or Changing To a Data Primary Translation Word

INITIAL CONDITIONS: The PTW is in a table linked to the master head table or its annex, either directly or via one intervening table. If the NEW form of the message (RC:GENT:) is used, the PTW must be zero; if the CHG form is used, the PTW cannot be zero.

RESULTS OF MESSAGE: If PAUX is entered (CHG message) and bits 22 through 20 of the PTW are initially 000, the remainder of the word is interpreted as an auxiliary address and the auxiliary block is released. The one or more keyword units of data specified are added together and the result is inserted into the existing PTW.

Note: Refer to paragraphs 1.03 and 1.04.



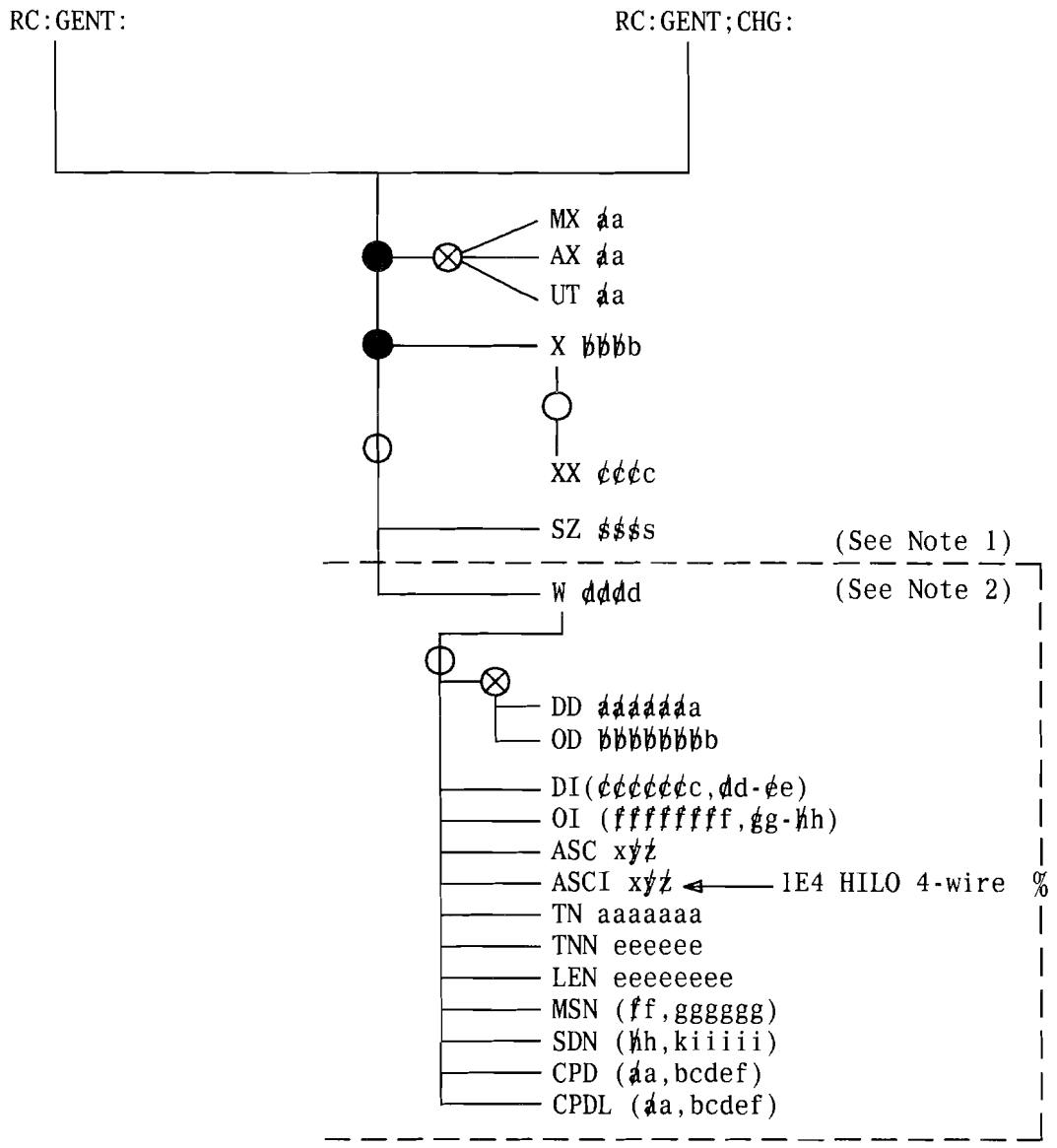
Building or Changing an Auxiliary Block

INITIAL CONDITIONS: The PTW is in a table linked directly to the unit type head table, or linked to the master head table or its annex either directly or via one intervening table. If the NEW form of the message (RC:GENT:) is used, the PTW must be zero; if the CHG form is used, the PTW cannot be zero.

RESULTS OF MESSAGE: If there is an existing auxiliary block of the required size, it is reused; otherwise, a new auxiliary block is seized and the old block (if any) is released. If there is an old block, its contents are copied into the new block, starting at the top and continuing until the end of either the new or the old block is reached. The remaining words (if any) in the new block, or all words if there was no old block, are initially zero. Each word specified (by W) in the message is then modified as follows: the one or more keyword units of data specified in a single segment are added together to form the segment specified data. This result is inserted into the base word to form the newly specified word. If the entire message is accepted, all newly specified words become effective in the RC area. ***Note:*** The meaning

of base word in the 1E5 generic program differs from that in earlier generics. (See paragraph 4.03 RC:GENT definitions.)

Note: Refer to paragraphs 1.03 and 1.04.



Note 1: A one-word auxiliary block size cannot be built. The auxiliary block size must be 2 to 2047. If a CHG form is used, the SZ keyword is needed only if the auxiliary block is new or of changed size. See definition of keyword SZ.

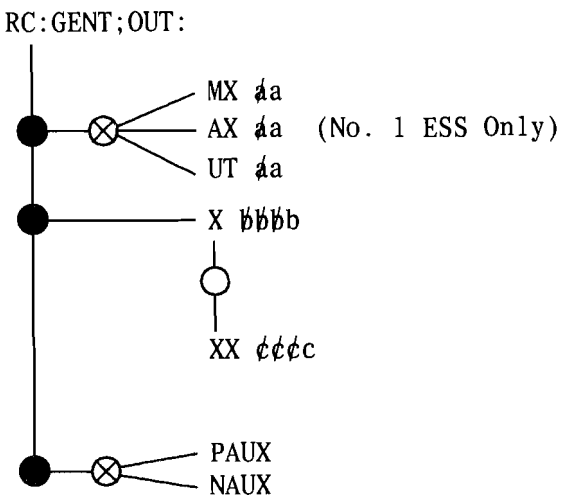
Note 2: In generics prior to 1E4, if there are any changes to word (W) 0, it is the responsibility of the user to preserve the auxiliary block size (WRDN item in bits 22 through 18). See definition of keyword W.

Note 3: An individual word (DD, OD, DI, or OI) can be entered in more than one message segment. After insertion into the first segment, ***only*** keywords DI and OI should be used in subsequent segments.

Zeroing a Primary Translation Word

INITIAL CONDITIONS: The nonzero PTW is in a table linked directly to the unit type head table, or linked to the master head table or its annex, either directly or via one intervening table.

RESULTS OF MESSAGE: If PAUX is entered and bits 22 through 20 of the PTW are 000, the remainder of the word is interpreted as an auxiliary address and this auxiliary block is released. In any case, the PTW is made zero.



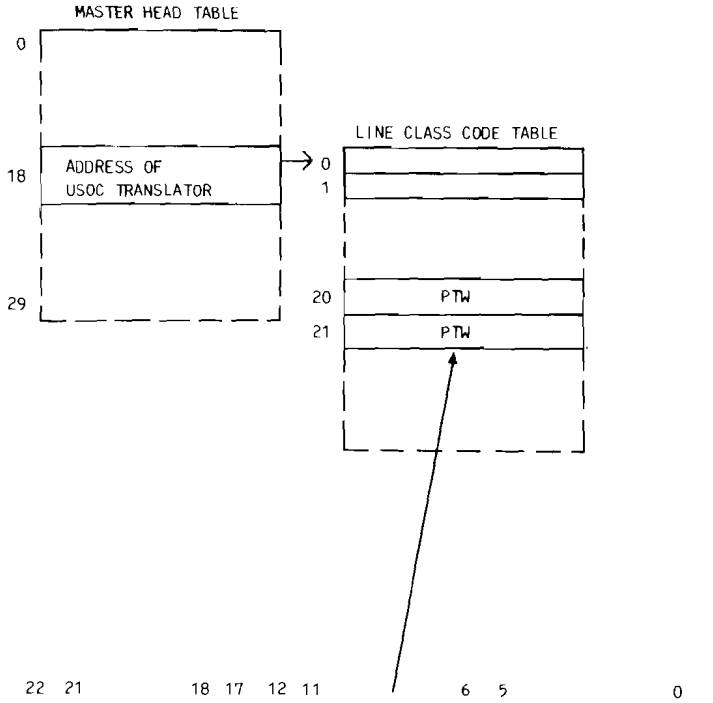
System response should be a TACK of OK and an OM of RC18 59 0 ACPT. An example of building or changing an auxiliary block and zeroing a primary translation word is given in Fig. 1 and 2.

KEYWORD UNIT	DEFINITION
ASC xy¢	Alphanumeric characters to be stored in 7-bit ASCII code in bits 20-0. If one character is inputted, it is stored in bits 20-14 with the ASII code for "blank" in the other two postions (bits 13-0). If two characters are inputed, a "blank" is stored in bits 6-0. Regardless of the number of characters inputted, the range of this data is bits 20-0. If 3 blank characters are required, use OD 2010040.
ASCI xy¢	Alphanumeric characters to be stored in 6-bit ASCII code in bits 17-0. If one character is inputted, it is stored in bits 17-12 with the ASCII code for "blank" in the other two positions (bits 11-0). If two characters are inputted a "blank" is stored in bits 5-0. Regardless of the number of characters inputted, the range of this data is bits 17-0. If 3 blank characters are required, use OD 404040. ASCI is used with the 1E4 generic for HILO 4-wire applications.
AX ¢a	Master Head Table Annex 0 to 39. Starting with 1E4 and later generics, Master Head Table Annex is 0 to 79.
CPD (¢a,bcdef)	Central Pulse Distributor Point (bcdef) is given in bits 13-0 (b = Pair, c = Half, d = Group, e = Row, f = Column) and quantity of points is given (¢a in decimal) in bits 17-14. The total range of this data is bits 17-0.
CPDL (¢a,bcdef)	Central Pulse Distributor Point (bcdef) is given in bits 13-0 (b = Pair, c = Half, d = Group, e = Row, f = Column) and left-adjusted quantity of points is given (¢a in decimal) in bits 22-18. The total range of this data is all bits except 17-14.
DD ¢¢¢¢¢¢a	Decimal Data (no greater than 1048575 which is 20 bits) right adjusted in the word. The range of the data is 23 bits.
DI (¢¢¢¢¢¢c, ¢d-¢e)	Decimal Item: the decimal value ¢¢¢¢¢¢c in decimal bits ¢d-¢e inclusive. The range of the data is the specified bit range, but the value can be no greater than 20 bits (which is 1048575). The range notation must be used even for single binary bits; for example, bit 8 must be inputted as the range 8-8.
LEN eeeeeeee	Line Equipment Number. The range of this data is bits 16-0.
MSN (¢f,gggggg)	Master Scanner Number (gggggg) in bits 15-0 and Quantity ¢f in decimal in bits 22-18. The total range of this data is all bits except 17-16.
MX ¢a	Master Head Table index 0 to 26, 28 and 29.
NAUX	No Auxiliary Block Possible. The PTW located at the final indexing is assumed to be a data word, regardless of its contents.
OD ¢¢¢¢¢¢¢¢b	Octal data (up to 37777777 which is 23 bits) right adjusted in the word. The range of the data is 23 bits. NOTE: The maximum value is 3777777 in 1E4 and earlier generics.
OI (¢¢¢¢¢¢¢f,¢g-¢h)	Octal item(up to 37777777): the octal value ¢¢¢¢¢¢¢f in decimal bits ¢g-¢h inclusive. The range notation must be used even for single bits; for example, bit 8 must be inputted as the range 8-8. NOTE: The maximum value is 3777777 in 1E4 and earlier generics.

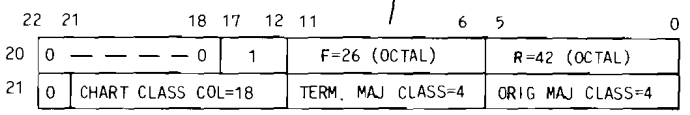
KEYWORD UNIT	DEFINITION
PAUX	Possible Auxiliary Block. The PTW is located at the final indexing (by keyword XX, or by keyword X if XX is not entered). If the top three bits (22-20) of the PTW are 000, the remainder of the word is assumed to be the address of an auxiliary block.
SDN (ℌh,kiiiii)	Signal Distributor Number (iiiii) and frame type (k = U,M,J, or S) in bits 17-0, and Quantity (ℌh in decimal) in bits 22-18. The total range of this data is 22-0.
SZ ♯♯♯a	Decimal size 2 to 2047 of the auxiliary block. This number does not include the 1 word that is automatically provided immediately preceding the 0 word to hold the size when greater than 31.
TN aaaaaaaa	Telephone (Directory) Number, input in 7-digit form and stored in 17-bit binary form. The range of this data is bits 16-0.
TNN eeeeee	Trunk Network Number. The range of this data is bits 14-0.
UT ♯a	Unit Type 1 to 62 (decimal). (Only RC:GENT;CHG: message can be used with UT12, 17, 19 and member number 2 of UT 50. NEW for of RC: GENT message cannot be used with the above UTs since they are a one-word auxiliary block).
W ¢¢¢d	Word number (decimal 0 to 2047) in the auxiliary block. This index identifies the word to be initialized or changed. The minus 1 word required when the auxiliary block has more than 31 words need not be referenced, since it is created automatically by the program. The WRDN item in word 0, bits 22-18 (giving the auxiliary block size when 31 words or less), is in generic 1E4 and later, also set automatically and protected against being overwritten by data specified for W0. *CAUTION:* With generics before 1E4, care must be taken to preserve or restore the WRDN item. The correct WRDN is preserved in a NEW message only if no data is entered for W0, and in a CHG message only if the auxiliary block size is not changed. For example, in a CHG message a segment containing only W0 and OD 3 would wipe out the WRDN item. It would have to be reinserted in a following segment. This could be avoided by using OI (3, 17-0) instead of OD 3. In a NEW message, even OI (3, 17-0) in W0 would wipe out WRDN: Entering both OI (3, 17-0) and DI (5, 22-18) in the same segment (assuming WRDN = 5) would solve the problem.
X ♭♭♭♭b	The decimal index 0 to 2047 into the table located by MX, AX, or UT keyword unit. This is often called a head table.
XX ¢¢¢c	The decimal index 0 to 2047 into the table located by the X keyword unit. This table is usually called a subtranslator.

Verification and Hunting Information

If the word being changed is RC hunted, the change is effective immediately and may be verified. Others are effective only after PS memory cards have been updated.

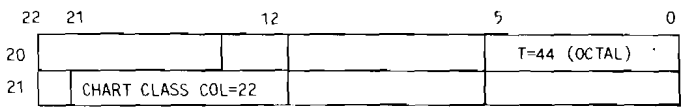


EXAMPLE OF BUILDING NEW PTW'S THAT WERE ZERO INITIALLY



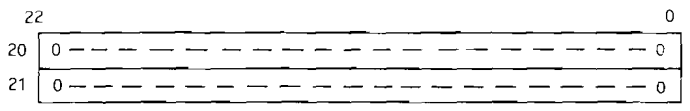

```
RC:GENT:        RC:GENT:
MX 18           MX 18
X 20            X 21
OD 42           OD 4
OI(26,11-6)     OI(4,11-6)      }*
DI(1,17-12)!    DI(18,21-12)!   }
```

EXAMPLE OF CHANGING PTW'S THAT WERE NON-ZERO INITIALLY




```
RC:GENT;CHG:    RC GENT;CHG:
NAUX            NAUX
MX 18           MX 18
X 20            X 21
OI(44,5-0)!     DI(22,21-12)!
```

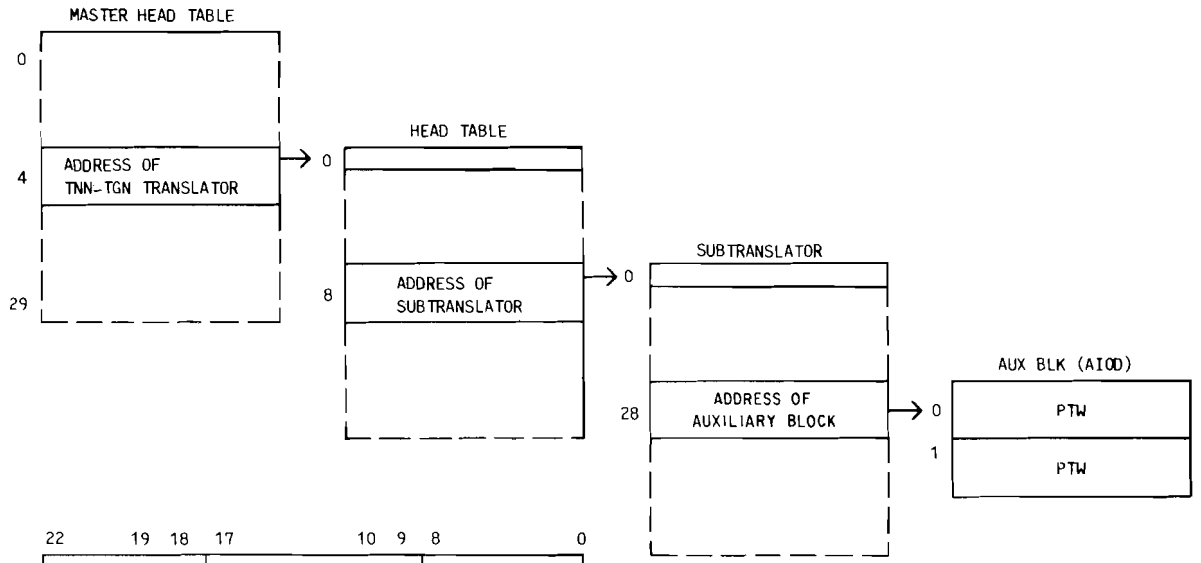
EXAMPLE OF ZEROING PTW'S THAT WERE NON-ZERO INITIALLY



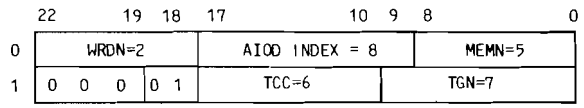

```
RC:GENT;OUT:    RC:GENT;OUT:
MX 18           MX 18
X 20            X 21
NAUX!           NAUX!
```

* ORIGINATING MAJOR CLASS AND TERMINATING MAJOR CLASS INFORMATION IS NORMALLY EXPRESSED IN DECIMAL VALUES, BUT SINCE THE EXAMPLE CLASS SHOWN (INDIVIDUAL LINE) IS LESS THAN 8, THE VALUES ARE THE SAME IN OCTAL OR DECIMAL. IN THIS EXAMPLE THE CHART CLASS COLUMN CAN BE EXPRESSED IN DECIMAL VALUE WITHOUT GENERATING ANOTHER RC MESSAGE. SINCE DI HAS BEEN USED, DI CANNOT BE REPEATED IN THE SAME SEGMENT.

Fig. 1—Examples of Building, Changing, or Zeroing a Primary Translation Word (RC:GENT)

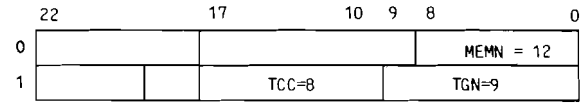


EXAMPLE OF AN AUXILIARY BLOCK WHEN THE SUBTRANSLATOR PTW WAS ZERO INITIALLY



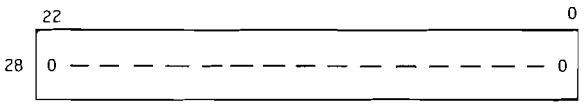

```
RC:GENT:
MX 4
X 8
XX 28
SZ 2
W 0
OI(5,8-0)
DI(8,17-9)%
W 1
OD 7
OI(6,17-10)
DI(1,22-18)!
```

EXAMPLE OF CHANGING AN AUXILIARY BLOCK




```
RC:GENT;CHG:
MX 4
X 8
XX 28
W 0
OI(12,8-0)%
W 1
DI(9,9-0)%
W 1
DI(8,17-10)!
```

EXAMPLE OF RELEASING AN AUXILIARY BLOCK



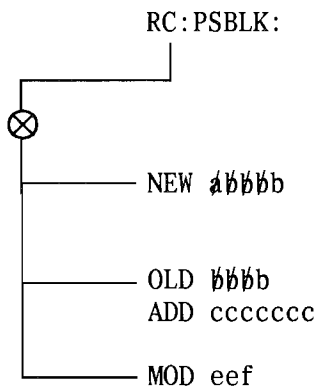

```
RC:GENT;OUT:
MX 4
X 8
XX 28
PAUX!
```

Fig. 2—Examples of Building, Changing, or Releasing an Auxiliary Block (RC:GENT)

5. SEIZING OR RELEASING PROGRAM STORE BLOCK RC:PSBLK (PR-1A342)

INITIAL CONDITIONS: To link a block using keywords OLD and ADD, the block that is identified must contain no test words (see Note 2). To link an entire module using keywords MOD and ADD, no part of the module can already be on the link list. Neither of these checks are made by the program.

RESULTS OF MESSAGE: If seizing a block of memory, a block of the size specified is removed from the link list and its address is indicated as shown below in Note 3(a). If returning a block of memory or a module, the block(s) is linked to the appropriate list(s). ***A block of PS memory is not initialized or zeroed when seized from or returned to the idle link list.***



(Use to seize and reserve a block of PS memory from the idle link list.)

(Use when a block of PS memory is to be returned to the idle link list; see Note 2.)

(Use when an entire PS module is to be placed on the idle link lists. The break-up forced when test words exist is handled automatically by the program.)

Note 1: For further information regarding linkages and verification of translations, refer to Section 231-119-320.

Note 2: When placing on the link lists a block of memory that was not seized as a single block, care should be taken to ensure that the block does not contain any test words (see Table A).

Note 3: When attempting to seize a PS block, the resulting output message(s) will be one of the following:

(a) If successful

RC18 8 0 INFO
ADD aaaaaaaa OCT bbbbbbbb

where 8 is the message index identifying the RC:PSBLK message and the second line gives the octal PS address and octal length of the block seized.

TABLE A

PROGRAM STORE TEST WORDS—RIGHT-HALF AND LEFT-HALF

PS BLOCK	LENGTH (WORDS)	RIGHT-HALF WORDS		LEFT-HALF WORDS	
		STARTING ADDRESS	ENDING ADDRESS	STARTING ADDRESS	ENDING ADDRESS
2	7	1437674	1437702	5437674	5437702
	6	1440075	1440102	5440075	5440102
	6	1537675	1537702	5537675	5537702
	6	1540075	1540102	5540075	5540102
3	7	1637674	1637702	5637674	5637702
	6	1640075	1640102	5640075	5640102
	6	1737675	1737702	5737675	5737702
	6	1740075	1740102	5740075	5740102
4	7	2037674	2037702	6037674	6037702
	6	2040075	2040102	6040075	6040102
	6	2137675	2137702	6137675	6137702
	6	2140075	2140102	6140075	6140102
5	7	2237674	2237702	6237674	6237702
	6	2240075	2240102	6240075	6240102
	6	2337675	2337702	6337675	6337702
	6	2340075	2340102	6340075	6340102
6	7	2437674	2437702	6437674	6437702
	6	2440075	2440102	6440075	6440102
	6	2537675	2537702	6537675	6537702
	6	2540075	2540102	6540075	6540102
7	7	2637674	2637702	6637674	6637702
	6	2640075	2640102	6640075	6640102
	6	2737675	2737702	6737675	6737702
	6	2740075	2740102	6740075	6740102
8	7	3037674	3037702	7037674	7037702
	6	3040075	3040102	7040075	7040102
	6	3137675	3137702	7137675	7137702
	6	3140075	3140102	7140075	7140102
9	7	3237674	3237702	7237674	7237702
	6	3240075	3240102	7240075	7240102
	6	3337675	3337702	7337675	7337702
	6	3340075	3340102	7340075	7340102
10	7	3437674	3437702	7437674	7437702
	6	3440075	3440102	7440075	7440102
	6	3537675	3537702	7537675	7537702
	6	3540075	3540102	7540075	7540102
11	7	3637674	3637702	7637674	7637702
	6	3640075	3640102	7640075	7640102
	6	3737675	3737702	7737675	7737702
	6	3740075	3740102	7740075	7740102

This message will then be followed by the standard RC18 ACPT message.

(b) If unsuccessful because a block of the requested type (left half or right half) and size is not available

RC18 8 0 NOPS

where NOPS indicates "NO (or not enough) Program Store."

KEYWORD UNIT	DEFINITION
ADD ccccccc	Octal address of PS Block to be returned to the idle trunk list with bit 20 indicating right or left half as part of the address.
ADD mmmmmmm	Octal address of MOD to be placed on the link list; right-half address (bit 20=0) of first word in the mod being placed on the (left and right half) link list. This address serves as a check on the module number specified.
MOD eef	Information Block and Module Number (Table B) ee = Octal Information Block Number (see Table B). f = Octal Module Number (see Table B).
NEW ¢ØØØb	Type and Length of Requested PS Block ¢ = R for Right-Half Block or L for Left-Half Block ØØØb = Decimal Length (Number of Words) in Block
OLD ØØØb	Decimal Length of PS Block Being Returned

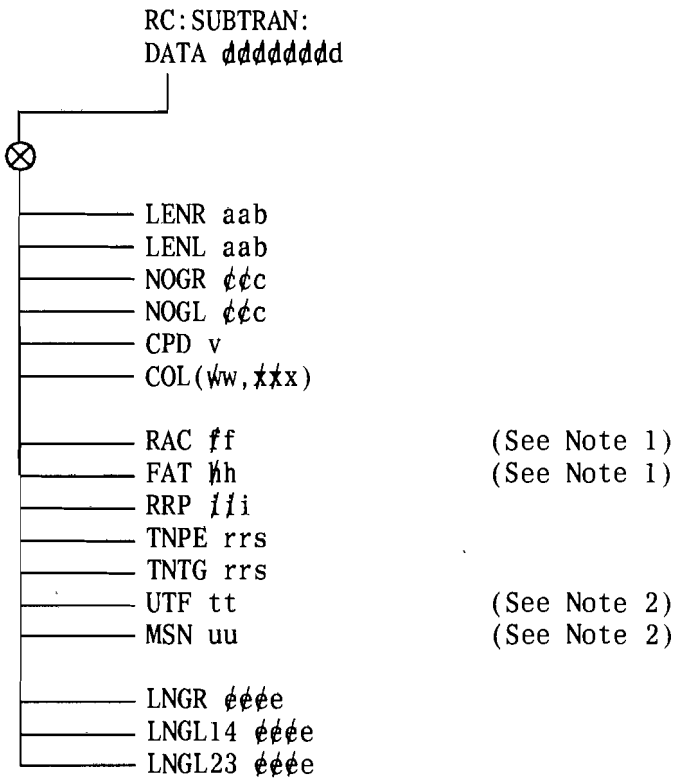
Verification and Hunting Information

There is no verification format provided for this type of RC. The changes to the idle link lists are effective immediately.

6. SEIZING AND INITIALIZING A SUBTRANSLATOR BLOCK RC:SUBTRAN (PR-1A356)

INITIAL CONDITIONS: The desired subtranslator does not exist, but its head table does. For NOGR or NOGL to be inputted, the DN head table contains a blank number intercept (RI86) entry. (See RC:NOCNOG in Section 231-118-324).

RESULTS OF MESSAGE: A PS memory block of the size required or specified is seized and the data indicated (keyword DATA) replaces existing data in each word of the block. The address of the subtranslator and the address to which it is to be linked is printed (see Note 4). For a directory number subtranslator (keyword NOGR or NOGL), a temporary recent change containing the address of the subtranslator is placed on the address in the DN head table to which it is to be linked. (See RC:DNHT in Section 231-118-324.)



Note 1: For the RAC and FAT keywords, subtract octal 323 (decimal 211) from the block address to derive the actual subtranslator address to place into the head table. To seize a 1000-word block for a toll 3- or 6-digit translator or a 258-word block from an abbreviated 6-digit FAT, use key word LNGL14 (or LNGR if right half entries are desired).

Note 2: UTF and MSN can be left side translator using two (2) left side words for each PTW. However, no special keywords are provided for such cases. To build such a translator, use the LNGL23 keyword and calculate the head table address that must be initialized to the address of the subtranslator block seized.

6.01 When attempting to seize a subtranslator block, the resulting message(s) will be one of the following:

(a) If successful

RC18 17 0 INFO
ADD aaaaaaaa OCT bbbbbbbb

RC18 17 0 INFO
HTA cccccccc

where 17 is the message index identifying the RC:SUBTRAN message and the second line gives the octal PS address and the octal length of the subtranslator block seized. The second RC18 INFO message gives the head table address that must be initialized to the address aaaaaaaa or aaaaaaaa minus decimal 211. (This second message is not printed if LNGR, LNGL14, or LNGL23 keywords are typed in.)

This message will then be followed by the standard RC18 ACPT message.

(b) If unsuccessful because a block of the requested type (left half or right half) and size is not available

RC18 17 0 NOPS

where NOPS indicates "NO (or not enough) Program Store."

KEYWORD UNIT	DEFINITION
COL(ẅw,x̄x̄x)	Chart Column ẅw = Chart (decimal) (See Note) x̄x̄x = Chart Column (decimal)
CPD v	Central Pulse Distributor Pair
DATA d̄d̄d̄d̄d̄d̄d̄d	Octal Data; data is placed in every word of the seized block
FAT ẖh	Foreign Area Translator (decimal)
LENL aab	Line Link Network and Line Switch Frame or Line Switch Circuit for Left Subtranslator aa = Line Link Network b = Line Switch Frame or Circuit
LENR aab	Line Link Network and Line Switch Frame or Line Switch Circuit for Right Subtranslator aa = Line Link Network b = Line Switch Frame or Circuit
LNGL14 ¢e	Block Length of 14-Bit Left-Half Words in Decimal
LNGL23 ¢e	Decimal Block Length of 23-Bit Left-Half Words; a 23-bit left-half word is formed from two 14-bit left-half words.
LNGR ¢e	Length of Right-Half PS Block in Decimal
MSN uu	Master Scanner Frame Number
NOGL ¢c	Number Group Number (decimal) for Left-Half DN Subtranslator
NOGR ¢c	Number Group Number (decimal) for Right-Half DN Subtranslator
RAC ff	Rate Center Index (decimal)
RRP j̄i	Rate and Route Pattern Number (decimal)
TNPE rrs	TNN-to-PEN Translator rr = Trunk Link Network s = Trunk Switch Frame or Circuit
TNTG rrs	TNN-to-TGN Translator rr = Trunk Link Network s = Trunk Switch Frame or Circuit
UTF tt	Universal Trunk Frame Number

Note: The chart is used only to determine whether the subtranslator block is 33 decimal words (for chart = 3, 7, 11, or 15) or 65 decimal words (for all other chart values from 1 to 15).

Verification and Hunting Information

There is no verification format provided for this type of RC. The changes to the idle link lists are effective immediately, but the block is not effective in call processing until the PS memory cards have been updated and the block is linked with the head table (see paragraph 7.05).

TABLE B

INFORMATION BLOCKS, MODULE NUMBERS, AND MODULE STARTING ADDRESSES

INFORMATION BLOCK AND MODULE NUMBER MOD:eef	STARTING ADDRESS (OCTAL)	ENDING ADDRESS (OCTAL)	INFORMATION BLOCK AND MODULE NUMBER MOD:eef	STARTING ADDRESS (OCTAL)	ENDING ADDRESS (OCTAL)	INFORMATION BLOCK AND MODULE NUMBER MOD:eef	STARTING ADDRESS (OCTAL)	ENDING ADDRESS (OCTAL)
030	1600000	1617777	060	2400000	2417777	110	3200000	3217777
031	1620000	1637777	061	2420000	2437777	111	3220000	3237777
032	1640000	1657777	062	2440000	2457777	112	3240000	3257777
033	1660000	1677777	063	2460000	2477777	113	3260000	3277777
034	1700000	1717777	064	2500000	2517777	114	3300000	3317777
035	1720000	1737777	065	2520000	2537777	115	3320000	3337777
036	1740000	1757777	066	2540000	2557777	116	3340000	3357777
037	1760000	1777777	067	2560000	2577777	117	3360000	3377777
040	2000000	2017777	070	2600000	2617777	120	3400000	3417777
041	2020000	2037777	071	2620000	2637777	121	3420000	3437777
042	2040000	2057777	072	2640000	2657777	122	3440000	3457777
043	2060000	2077777	073	2660000	2677777	123	3460000	3477777
044	2100000	2117777	074	2700000	2717777	124	3500000	3517777
045	2120000	2137777	075	2720000	2737777	125	3520000	3537777
046	2140000	2157777	076	2740000	2757777	126	3540000	3557777
047	2160000	2177777	077	2760000	2777777	127	3560000	3577777
050	2200000	2217777	100	3000000	3017777	130	3600000	3617777
051	2220000	2237777	101	3020000	3037777	131	3620000	3637777
052	2240000	2257777	102	3040000	3057777	132	3640000	3657777
053	2260000	2277777	103	3060000	3077777	133	3660000	3677777
054	2300000	2317777	104	3100000	3117777	134	3700000	3717777
055	2320000	2337777	105	3120000	3137777	135	3720000	3737777
056	2340000	2357777	106	3140000	3157777	136	3740000	3757777
057	2360000	2377777	107	3160000	3177777	137	3760000	3777777

Note: Section 231-104-302 contains a detailed breakdown of program store physical locations and addresses.

7. PROCEDURES UTILIZING RC:PSWD, RC:PSBLK, AND RC:SUBTRAN

SEIZING A BLOCK OF PROGRAM STORE MEMORY

7.01 The RC:SUBTRAN message should be used to seize a subtranslator. It permits every word in the block to be conveniently initialized to the same value (See Part 6). Except when one of the general keywords LNGR, LNGL14 or LNGL23 is used, the correct block length is supplied automatically and the head table address to which the block must be linked is printed.

The RC:PSBLK message is less specialized. It can be used to seize a block for any purpose, but provides no initialization facilities or automatic length information.

> ***Note:*** Before seizing a block of program store (PS) memory, it ***may be necessary*** to check the link list for available space. After returning a block of PS memory to the link list, it ***may also be necessary*** to check the link list to see that the block has been returned. See Part 8 for these procedures.

7.02 If space is not available, use paragraphs 7.10 and 7.11.

7.03 Seize a block of PS memory (without initializing) by typing in

```
RC:PSBLK:
NEW ddddb!
```

If the message is accepted, system prints OK followed by RC18 output message giving octal starting address and octal length of the reserved block.

7.04 Seize a block of PS memory (with initialization) by using the RC:SUBTRAN message explained in Part 6. Update PS memory cards as described in Section 231-104-301.

> ***Caution: PS memory cards must be updated before words seized by RC:SUBTRAN can be changed by RC:PSWD.***

LINKING A SUBTRANSLATOR TO A HEAD TABLE

7.05 Use one of the following procedures to link the new subtranslators.

A. Directory Number (DN) Subtranslators

(1) Seize a block of PS memory (with initialization) as in paragraph 7.04.

(2) Use RC:DNHT message in Section 231-118-324 to automatically link the subtranslator to the directory number head table.

System response should be a TACK of OK and an OM or RC 18 26 0 ACPT.

(3) If more than one subtranslator is being added, repeat (1) and (2) for each additional subtranslator.

(4) Update PS memory cards.

B. Other Subtranslators

(1) Seize a block of PS memory using the RC:SUBTRAN message (See Part 6 and paragraph 7.01). System response should be two RC18 INFO messages and an RC18 ACPT message.

(2) Update ***single*** PS memory card (Section 231-104-302).

(3) Verify the contents of the subtranslator block using the T-READ message.

(4) Obtain subtranslator address (ADD aaaaaaaa) and head table address (HTA cccccccc) printed out in paragraph 6.01 when subtranslator block was obtained. The user must compute cccccccc if keyword LNGR, LNGL14, or LNGL23 was used to build the subtranslator block.

(5) Link subtranslator by typing in

```
RC:PSWD
ADD ccccccc,
DAT kaaaaaaa!
```

ccccccc = Address (HTA found in paragraph 6.01)

*k = 0, or 3

aaaaaaaa = Address (ADD found in paragraph 6.01)

System response should be a TACK of OK and an OM of RC18 1 0 ACPT.

* The k represents the upper 2 bits in a head table entry, if needed, to distinguish subtranslator types for a particular translator. Usually k may be omitted or set to zero. Table C exhibits the different values of k for the 3/6-digit translator.

If a line switch frame pair (LENR or LENL keyword in Part 6 in a 2:1 line link network (LLN) office consists of a mate frame followed by its associated home frame (mate-home), subtract octal 20 from the ADD aaaaaaaa to derive the address to be placed into the line equipment number (LEN) head table. If the circuit pair (remreed) is even-odd or if the frame pair is home-home, home-mate, or mate-mate, the ADD aaaaaaaa in paragraph 6.01 is used.

(6) If more than one subtranslator is being added, repeat (1) through (5) for each additional subtranslator.

(7) Update PS memory cards.

Verification

7.06 To verify the new subtranslator linkage, use the following procedure.

(1) Type in

T-READ-ccccccc 01.

ccccccc = Address (HTA found in paragraph 6.01) that points to subtranslator.

System response TW02 message should contain the 8-digit octal number, kaaaaaaa, which is the starting address of the subtranslator block seized in paragraph 7.03.

(2) Repeat (1) for each subtranslator block seized in paragraph 7.03.

(3) If errors are found in (1) or (2), repeat appropriate RC:PSWD messages and update PS memory cards again. ***Do not continue until all linkages are correct.***

RETURNING A BLOCK OF PROGRAM STORE MEMORY

Caution: Before returning a block of PS memory to the link list of available space, it is necessary to ascertain that none of the addresses given in Table A are contained in the block being returned. (These addresses contain PS diagnostic or central control diagnostic words that may contain an intentionally bad Hamming code. This information cannot be regenerated with the normal system facilities). The result will be continuous PS diagnostic failures if any of these addresses are used for any other purpose.

7.07 The addresses of the PS test words depend on the number of the information block. Information blocks are numbered 0 through n, where n + 1 is the number of PSs in the office and n has a maximum value of 11 decimal. Information blocks 0 and 1 are omitted from Table A because they cannot be used for translation data. (They also contain more diagnostic test words than information blocks 2 through 11.) More detail is given in information notes in PF-1A021.

7.08 Record the octal starting address and ending address of the block being returned. Compare each address listed in Table A with the starting and ending addresses of the block being returned. If any of the addresses from Table A are between the starting and ending addresses of the block being returned, split the block into smaller blocks and skip all addresses of Table A.

7.09 For each block of memory being returned to PS memory, use the following procedure.

(1) Check link list of available space by typing in

VFY-SPACE-29 b cc 0.

b = 0 for right half of PS block

= 1 for left half of PS block

cc = 01 through 31 for decimal length of block

= 32 for blocks of length greater than or equal to decimal 32.

(2) Record system response (octal number of available blocks given in TR13 message) for later comparison in paragraph 8.04(3).

(3) Return a block of PS memory by typing in

RC:PSBLK:
OLD ƀƀƀb,
ADD ccccccc!

ƀƀƀb = For a block of right half 23-bit words, enter the decimal length of the block

= For a block of left half 14-bit words, enter the decimal length of the block

= For a block of left half 23-bit words, enter ***double*** the decimal number of 23-bit words in the block.

Note: The decimal length of left half blocks must be an even number, and the starting address of the block must be an even address.

ccccccc = Octal starting address of the block to be returned.

System response should be a TACK of OK and an OM of RC18 8 0 ACPT.

ADDING NEW TRANSLATION MODULES TO LINK LIST OF AVAILABLE SPACE

7.10 The telephone company may desire not to add all of the new translation modules to the link list initially. The advantage of not linking unused translation memory is that the system will be forced to build new translations in a more compact and controlled way, thus minimizing the number of modules that must be cardwritten. The translation data assembler (TDA) program does not put unused translation modules on the link list. Table B gives the module numbers and the starting addresses of the new translation modules for a growth PS.

7.11 Use the following procedure to add a new module to the link list. This procedure adds the new mod to both right half and left half link lists and automatically avoids diagnostic test words in those mods containing them.

(1) Type in

RC:PSBLK:
MOD eef,
ADD mmmmmmm!

eef = Information block and module number from Table B

mmmmmmm = First address in module from Table B.

System response should be a TACK of OK and an OM of RC18 8 0 ACPT.

(2) Repeat message in (1) for each additional module being added to link list.

8. CHECKING LINK LIST FOR AVAILABLE SPACE

8.01 It may become necessary to check the link list for available space (see paragraph 7.01).

8.02 Check link list for available space by typing in

VFY-SPACE-29 b cc 0.

b = 0 for right half of PS block

= 1 for left half of PS block

cc = 01 through 31 for decimal length of block

= 32 for blocks of length greater than or equal to decimal 32.

Record system response (octal number of available blocks given in TR13 message) for later comparison in paragraph 8.03(2).

8.03 Check that the block of memory seized was removed from link list of available space (recorded in paragraph 8.02) by using the following procedure.

(1) Type same message typed in paragraph 8.02.

(2) Record system response (octal number indicating the number of available blocks given in TR13 message).

(3) Compare octal number recorded in (2) with the octal number recorded in paragraph 8.02. Number in (2) should be one less than number in paragraph 8.02.

8.04 It may become necessary to check that the block of memory returned in paragraph 7.09 (3) was added to the link list of available space recorded in paragraph 7.09 (2). To do this:

(1) Type in

VFY-SPACE-29 b cc 0.

b = 0 for right half of PS block

= 1 for left half of PS block

cc = 01 through 31 for decimal length of the block

= 32 for blocks of length greater than or equal to decimal 32.

(2) Record system response (the octal number of available blocks given in TR13 message).

(3) Compare the octal number recorded in (2) with the octal number recorded in paragraph 7.09 (2). The number in (2) should be one greater than the number in paragraph 7.09 (2).

TABLE C

SUBTRANSLATOR TYPES FOR 3/6 DIGIT HEAD TABLE

INDEX	STTYP (BITS 22-21)	SIZE OF SUBTRANS-LATOR (WORDS)	WITH RC: SUBTRAN, USE KEYWORD	K	COMMENTS
RAC (Rate Center)	00	820	RAC	0	Subtract $323_8 (211_{10})$ from ADD aaaaaaaa printed out in paragraph 7.01 to derive the subtranslator address to be placed in the head table.
Abbreviated FAT Abbreviated TFAI	11	258	LNGL14 or LNGR	3	The address (ADD) printed in paragraph 7.01 is the actual subtranslator address to be placed in head table.

9. ABBREVIATIONS

ACPT	Accept
ADD	Address
ASC	Characters in 7-bit ASCII code
ASCI	Characters in 6-bit ASCII code
AX	Master Head Table Annex
CHG	Change
COL	Chart Column
CPD	Central Pulse Distributor
CPDL	Central Pulse Distributor Point, Quantity Left Adjusted
CPY	Copy
DAT	Octal Data
DATA	Octal Data
DD	Decimal Data
DI	Decimal Item
DN	Directory Number
ESS	Electronic Switching System
FAT	Foreign Area Translator
HTA	Head Table Address
INFO	Information
LEN	Line Equipment Number
LENL	Line Link Number and Line Switch Frame or Line Switch Circuit, Left Subtranslator
LENR	Line Link Number and Line Switch Frame or Line Switch Circuit, Right Subtranslator
LNG	Block Length (Decimal)
LH23	Left-Half Address, 23-Bit Word Pairs
LLN	Line Link Network
LNG14	Block Length, 14-Bit Left-Half Words
LNGL23	Block Length 23-Bit Left-Half Words
LNGR	Block Length, Right-Half
MOD	Information Block and Module Number
MSN	Master Scanner Number
MX	Master Head Table Index
NAUX	No Auxiliary Block Possible
NEW	Type and Length of Requested Memory Block
NOGL	Number Group Number, Left-Half ON Subtranslator
NOGR	Number Group Number, Right-Half DN Subtranslator
NOPS	No (or not enough) Program State
OD	Octal Data
OI	Octal Item
OLD	Decimal Length of Memory Block Being Returned
OM	Output Manual
ORD	Order Number
PAUX	Possible Auxiliary Block
PS	Program Store
PTW	Primary Translation Word
RAC	Rate Center Index
RC	Recent Change

RC:DNHT	Directory Number Head Table (Section 231-118-324)
RC:GENT	General Translator
RC:NOCNOG	Normalized Office Code/Number Group Number (Section 231-118-324)
RC:PSBLK	Program Store Block
RC:PSWD	Program Store Word
RC:SUBTRAN	Subtranslator
RRP	Rate and Route Pattern
SDN	Signal Distributor Number
SZ	Decimal Size of Auxiliary Block
TACK	TTY Acknowledgment
TDA	Translation Data Assembler
TFAI	Toll Foreign Area Index
TG1A	Translations Guide for No. 1 ESS
TN	Telephone (Directory) Number
TNN	Trunk Network Number
TNPE	TNN to PEN Translator
TNTG	Universal Trunk Frame Number
UT	Unit Type
UTF	Universal Trunk Frame Number
VFY	Verify
W	Word Number (decimal) in Auxiliary Block
WDN	Word Number
X	Decimal Index into Table Located by MX, AX, or UT Keyword Unit
XX	Decimal Index into Table Located by X Keyword Unit