IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZKGA-B-D

PRODUCT NAME: KG11A - CYCLIC REDUNDANCY CHECK TEST

DATE RELEASED: MARCH, 1977

MAINTAINER: DIAGNOSTIC GROUP

NOTE: REV. A OF THIS PROGRAM OBSOLETE MD-11-DBK

COPYRIGHT 1971-1977 BY DIGITAL EQUIPMENT CORPORATION

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT. THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>3</td>
</tr>
<tr>
<td>REQUIREMENTS</td>
<td>3</td>
</tr>
<tr>
<td>LOADING--STORAGE</td>
<td>3</td>
</tr>
<tr>
<td>OPERATION</td>
<td>3</td>
</tr>
<tr>
<td>SWITCH REGISTER</td>
<td>4</td>
</tr>
<tr>
<td>NOTES</td>
<td>4</td>
</tr>
<tr>
<td>FULL TEST MODE</td>
<td>5</td>
</tr>
<tr>
<td>SELECT TEST MODE</td>
<td>6</td>
</tr>
<tr>
<td>ERRORS</td>
<td>8</td>
</tr>
<tr>
<td>SCOPE LOOP</td>
<td>9</td>
</tr>
<tr>
<td>FORCED ERROR TYPEOUT</td>
<td>10</td>
</tr>
<tr>
<td>INSTRUCTION TABLE</td>
<td>13</td>
</tr>
<tr>
<td>DATA WORD TABLE</td>
<td>16</td>
</tr>
<tr>
<td>TESTA</td>
<td>23</td>
</tr>
<tr>
<td>TESTB</td>
<td>24</td>
</tr>
</tbody>
</table>
ABSTRACT.

------------

THIS PROGRAM TESTS THE LOGIC OF THE CYCLIC REDUNDANCY CHECK DEVICE (KGII). 

REQUIREMENTS.

--------------

A STANDARD PDP-11 (WITH OR WITHOUT A HARDWARE SWITCH REGISTER) AND A KGII

LOADING--STORAGE.

-------------------

LOADING PROCEDURE FOR NORMAL BINARY TAPES SHOULD BE FOLLOWED.

DEVICE ADDRESS CHANGE.

--------------------------

TO CHANGE THE DEVICE ADDRESS (WHICH WAS LOADED AS 170700), CHANGE THE CONTENTS OF THE LOCATION "DEVADR" SHOWN ON PAGE 24/ SET THIS LOCATION TO THE ADDRESS OF THE CSR DESIRED, THEN START (OR RESTART) AS GIVEN IN OPERATION BELOW.

OPERATION.

-------------------------

1. THIS PROGRAM MAY BE OPERATED IN TWO MODES.

A. FULL TEST MODE.

   THIS IS THE MAIN BODY OF THE PROGRAM AND SHOULD BE USED TO ACCEPT OR DIAGNOSE A DEVICE.
   TO RUN: START AT LOC. 200 WITH SWRIS SET.
   (DETAIL ON PAGE 5.)

B. SELECT TEST MODE.

   THIS IS A SUBPROGRAM TO ALLOW THE OPERATOR TO RUN A SELECT INSTRUCTION ON A SELECT DATA WORD.
   TO RUN: START AT LOC. 204 WITH SWRIS SET.
   SELECT INSTRUCTION ON SWRS-0.
   SELECT DATA WORD ON SWR11-6.

   (DETAIL ON PAGE 6.)
SWITCH REGISTER.

---

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER. IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH REGISTER (LOC: 176) IS USED.

CONTROL:

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER (LOC: 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY DOING THE FOLLOWING:

1) TYPE CONTROL G (<G>): THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.

2) THE MACHINE WILL THEN TYPE: SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER.)

3) AFTER THE 'NEW=' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:
   A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>. (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS WILL BE ALLOWED.) IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.
   B) IF A CONTROL U (<U>) IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO_STEP 2.

1. SWR15  SET, HALT ON ERROR.
          RESET, BYPASS ERROR.

2. SWR14  SET, SCOPE LOOP ON ERROR.
          RESET, BYPASS ERROR.

3. SWR13  SET, INHIBIT PRINTOUTS DURING SCOPE LOOP.
          RESET, ALLOW PRINTOUTS DURING SCOPE LOOP.

4. SWR12  SET, INHIBIT TRACING.
          RESET, ALLOW TRACING.

5. SWR11  SET, INHIBIT ITERS.
          RESET, ALLOW ITERS.

6. SWR11 - SWR5 AND SWR5 - SWRO ARE DEFINED IN THE SELECT TEST MODE SECTION ON PG. 6 & 7.
FULL TEST MODE.

-----------------

1. START OR RESTART.
   A. ZERO THE SWR
   B. LOAD 200 AND START

2. PROGRAM ACTION.
   A. WHENEVER THE PROGRAM IS STARTED, OR RESTARTED, THE
      TTY WILL TYPE: KG11A.
   B. IF THE SOFTWARE SWITCH REGISTER IS SELECTED
      THEN THE FOLLOWING WILL BE TYPED:
      SWR=XXXXXX NEW=(REFER TO SWITCH REGISTER
      SECTION FOR OPERATOR ACTIONS)
   C. IF THERE ARE 10 ERRORS DETECTED, THE PROGRAM WILL LOOP
      INDEFINITELY AND WILL RING THE TTY BELL ONCE FOR EACH
      PASS. TRACE TRAP IS EFFECTIVE AND EACH SUBTEST IS BEING
      ITERATED 100 TIMES.
   D. IF AN ERROR OCCURS, A ONE LINE ERROR MESSAGE WILL BE
      TYPED ON THE TTY AND THE PROGRAM WILL HALT. (SEE THE
      ERROR SECTION ON PAGE 8.)

3. OPERATOR CONTROL VIA THE SWR.
   (REFER TO SECTION ON SWITCH REGISTER SETTINGS
   FOR SOFTWARE SWITCH REGISTER.)
   A. SWR15 SET. (NORMAL)
      HALT ON ERROR AFTER ERROR MESSAGE IS TYPED.
      SWR15 RESET.
      TYPE THE ERROR MESSAGE AND CONTINUE TO THE NEXT ITERATION
      OF THE CURRENT TEST OR TO THE NEXT SUBTEST.
   B. SWR12 SET.
      INHIBIT TRACE TRAP.
      SWR12 RESET (NORMAL)
      ALLOW TRACE TRAP.
   C. SWR11 SET.
      INHIBIT ITERATIONS
      SWR11 RESET (NORMAL)
      ALLOW ITERATIONS
SELECT TEST MODE.

1. START OR RESTART.
   A. RESET SWR15 THRU SWRO.
   B. LOAD ADDRESS 204
   C. SELECT THE INSTRUCTION TO BE TESTED FROM THE LIST OF INSTRUCTIONS ON PAGE NN OF THE PROGRAM LISTING. SET SWR5 THRU SWRO TO THE "I" NUMBER GIVEN WITH SAID INSTRUCTION.
   D. SELECT THE DATA WORD TO BE TESTED FROM THE LIST OF DATA WORDS GIVEN ON PAGES NN AND NN OF THE PROGRAM LISTING. SET SWR11 THRU SWR6 TO THE "D" NUMBER GIVEN WITH THE SAID DATA WORD.
   E. START.
   F. ****REFER TO SECTION ON SWITCH SETTINGS FOR SOFTWARE SWITCH REGISTER ACTION.

SPECIAL NOTES ON C. AND D. ABOVE.

1. THE SELECTED INSTRUCTION AND/OR THE SELECTED DATA MAY BE CHANGED WHILE THE PROGRAM IS RUNNING.
2. IF SWR11 THRU SWR6 ARE ALL RESET, THEN THE SELECTED INSTRUCTION WILL RUN ON ALL DATA WORDS.
3. IF SWR5 THRU SWRO ARE ALL RESET THEN ALL INSTRUCTIONS WILL BE RUN ON THE SELECTED DATA WORD.
4. IF SWR11 THRU SWR6 AND SWR5 THRU SWRO ARE ALL RESET, THEN ALL INSTRUCTIONS WILL BE RUN ON ALL DATA WORDS.
2. PROGRAM ACTION

A. IF THERE ARE NO ERRORS THE PROGRAM WILL LOOP THE SELECTED INSTRUCTION (SWR5-D) USING THE SELECTED DATA (SWR11-6). TRACE TRAP IS EFFECTIVE.

B. IF AN ERROR OCCURS, A ONE LINE ERROR MESSAGE WILL BE TYPED ON THE TTY AND THE PROGRAM WILL HALT. (SEE THE ERROR SECTION ON PAGE 8.)

(REFER TO SWITCH SETTING SECTION FOR USE OF SOFTWARE SWITCH REGISTER)

3. OPERATOR CONTROL VIA THE SWR.

(REFER TO SECTION ON SWITCH SETTING'S FOR SOFTWARE SWITCH REGISTER DYNAMIC CHANGING.)

A. SWR15 SET. (NORMAL)
HALT ON ERROR AFTER ERROR MESSAGE IS TYPED.

SWR15 RESET.
TYPE THE ERROR MESSAGE AND CONTINUE TESTING THE SELECTED INSTRUCTION (SWR5-D) USING THE SELECTED DATA (SWR11-6)

B. SWR12 SET.
INHIBIT TRACE TRAP.

SWR12 RESET. (NORMAL)
ALLOW TRACE TRAP.

C. SWR11 THRU SWR6.
SELECTED DATA WORD.

D. SWR5 THRU SRO.
SELECTED INSTRUCTION.
**ERRORS**

1. **ERROR MESSAGE**

   ***ROUTINE CHECKS FOR THE DYNAMIC CHANGING OF THE SOFTWARE SWITCH REGISTERS TO SWITCH SETTING SECTION FOR OPERATOR ACTION.***

   TESTX  IXX  DXX SXXXXX  HXXXXX

   TESTX  (X = A OR B)
   THIS IS THE NAME (TAG) OF THE TEST THAT WAS IN USE
   AT THE TIME OF THE ERROR.
   TESTA IS ON PAGE 27 OF THE PROG. LIST.
   TESTB IS ON PAGE 28 OF THE PROG. LIST.

   IXX  (XX = A NUMBER FROM 01 THRU 57)
   THIS IS THE NAME/NUMBER (TAG) OF THE INSTRUCTION IN
   USE AT THE TIME OF THE ERROR.
   RI (INST. POINTER) POINTS AT THIS INSTRUCTION.
   THE INSTRUCTION TABLE IS ON PAGE 16 OF THE PROG. LIST.

   DXX  (XX = A NUMBER FROM 01 THRU 77)
   THIS IS THE NAME/NUMBER (TAG) OF THE DATA WORD IN
   USE AT THE TIME OF THE ERROR.
   R2 (DATA WORD POINTER) POINTS AT THIS DATA WORD.
   THE DATA WORD TABLE IS ON PG. 10 OF THE PROG. LIST.

   SXXXXX  (SXXXXX = ANY 6 DIGIT, 16 BIT, OCTAL NUMBER)
   THIS IS THE SIMULATED (GOOD) BCC WORD.

   HXXXXX  (HXXXXX = ANY 6 DIGIT, 16 BIT, OCTAL NUMBER)
   THIS IS THE HARDWARE (BAD) BCC WORD.

   DONE BIT, S\S (EACH X CAN =0 OR 1)
   THIS IS THE STATUS OF THE DONE BIT (BIT 7 OF THE CSR).
   THE FIRST NUMBER (X\) IS WHAT THE DONE BIT
   SHOULD BE (GOOD). THE SECOND NUMBER (\X) IS
   WHAT THE DONE BIT ACTUALLY WAS.

2. **OPTIONS AFTER A HALT ON ERROR.**

   ***IF THE SOFTWARE SWITCH REGISTER IS USED THEN THE OPERATOR
   CAN CHANGE THE SWREG LOCATION BY TYPING A 1G AND THEN CONTINUING.***

   A.  SCOPE LOOP. SEE SCOPE LOOP SECTION.

   B.  BYPASS THIS ERROR AND CONTINUE TO THE NEXT ITERATION
       OR SUBTEST.

      1.  RESET SWR14 AND SWR15.

      2.  CONTINUE.

   C. UTILIZE SELECT TEST MODE (PAGE 6)
SCOPE LOOP

***ROUTINE CHECKS FOR +G FUNCTION.***

1. SET UP.
   A. SET SWR14.
   B. RESET SWR15.
   C. CONTINUE.

2. PROGRAM ACTION.
   THE PROGRAM WILL SCOPE LOOP ON THE FAILING TEST FOR AS LONG AS SWR14 IS SET, AND SWR15 IS RESET.

3. OPERATOR CONTROL VIA THE SWR.
   A. SWR13 SET.
      INHIBIT ERROR TYPEOUTS.
      SWR13 RESET.
      ALLOW ERROR TYPEOUTS.
   B. SWR12 SET.
      INHIBIT TRACE TRAPS.
      SWR12 RESET.
      ALLOW TRACE TRAPS.
FORCEC ERROR TYPEOUT.

IN THE EVENT OF AN UNEXPECTED OR ILLEGAL TRAP, OR AT ANY TIME THE OPERATOR DEEMS IT USEFUL, THE PROGRAM CAN BE HALTED (IF IT HASN'T ALREADY) AND A TYPEOUT CAN BE OBTAINED AS TO THE STATUS OF THE PROGRAM.

THIS TYPEOUT IS THE SAME ONE USED IN THE EVENT OF AN ACTUAL ERROR. (SEE ERROR MESSAGE ON PG. 8.)

1. HALT THE PROGRAM
2. START AT LOC. 210. (SWR SETTINGS ARE IMATERIAL.)

THE TYPEOUT WILL BE MADE AND THE PROGRAM WILL HALT. THE PROGRAM CAN NOW BE RESTARTED. (SEE OPERATION, PG. 3.)
:KGII-A CYCLIC REDUNDANCY CHECK DEVICE TEST

:PROGRAM OBSCIDES MD-11-DSK
:RELEASED 21 MAY 76 BY SAM CARPENTER
: SUPPORTS THE SOFTWARE SWITCH REGISTER LOC. 175
: ALSO SUPPORTS THE DYNAMIC LOADING OF LOC. 176
: REVISED TO MEET ALL A811 SPEC. S; REV. B RELEASED FEB. 1977

; TRAP CATCHER (LOC. 0 TO LOC. 776)  *************

000030  .REPT  30
         .HALT
         .ENDR  ; TRAPPED TO PREVIOUS ADDRESS.

000030  .REPT  30
         .HALT
         .ENDR  ; TRAPPED TO PREVIOUS ADDRESS.

000030  .REPT  30
         .HALT
         .ENDR  ; TRAPPED TO PREVIOUS ADDRESS.

000030  .REPT  30
         .HALT
         .ENDR  ; TRAPPED TO PREVIOUS ADDRESS.

000030  .REPT  30
         .HALT
         .ENDR  ; TRAPPED TO PREVIOUS ADDRESS.

000010  .REPT  10
         .HALT
         .ENDR  ; TRAPPED TO PREVIOUS ADDRESS.
; SOFTWARE SWITCH REGISTER**********

; HWREG: 0

; SOFTWARE SWITCH REGISTER

; PROGRAM STARTS************

NORMAL: JMP IDENT

SELECT: JMP INTA

FORERR: JMP FORCER
<table>
<thead>
<tr>
<th>Instruction</th>
<th>Table Value</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>I01</td>
<td>000002</td>
<td>LRCB 1 STEP.</td>
</tr>
<tr>
<td>I02</td>
<td>000002</td>
<td>LRCB 2 STEPS.</td>
</tr>
<tr>
<td>I03</td>
<td>001002</td>
<td>LRCB 3 STEPS.</td>
</tr>
<tr>
<td>I04</td>
<td>000202</td>
<td>LRCB 4 STEPS.</td>
</tr>
<tr>
<td>I05</td>
<td>000202</td>
<td>LRCB 5 STEPS.</td>
</tr>
<tr>
<td>I06</td>
<td>003002</td>
<td>LRCB 6 STEPS.</td>
</tr>
<tr>
<td>I07</td>
<td>003002</td>
<td>LRCB 7 STEPS.</td>
</tr>
<tr>
<td>I08</td>
<td>004002</td>
<td>LRCB 8 STEPS.</td>
</tr>
<tr>
<td>I09</td>
<td>004002</td>
<td>LRCB 9 STEPS.</td>
</tr>
<tr>
<td>I10</td>
<td>005002</td>
<td>LRCB 10 STEPS.</td>
</tr>
<tr>
<td>I11</td>
<td>006002</td>
<td>LRCB 11 STEPS.</td>
</tr>
<tr>
<td>I12</td>
<td>007002</td>
<td>LRCB 12 STEPS.</td>
</tr>
<tr>
<td>I13</td>
<td>007002</td>
<td>LRCB 13 STEPS.</td>
</tr>
<tr>
<td>I14</td>
<td>008002</td>
<td>LRCB 14 STEPS.</td>
</tr>
<tr>
<td>I15</td>
<td>009002</td>
<td>LRCB 15 STEPS.</td>
</tr>
<tr>
<td>I16</td>
<td>009002</td>
<td>LRCB 16 STEPS.</td>
</tr>
<tr>
<td>I17</td>
<td>010002</td>
<td>LRCB 17 STEPS.</td>
</tr>
<tr>
<td>I18</td>
<td>010002</td>
<td>LRCB 18 STEPS.</td>
</tr>
<tr>
<td>I19</td>
<td>011002</td>
<td>LRCB 19 STEPS.</td>
</tr>
<tr>
<td>I20</td>
<td>011002</td>
<td>LRCB 20 STEPS.</td>
</tr>
<tr>
<td>I21</td>
<td>012002</td>
<td>LRCB 21 STEPS.</td>
</tr>
<tr>
<td>I22</td>
<td>013002</td>
<td>LRCB 22 STEPS.</td>
</tr>
<tr>
<td>I23</td>
<td>014002</td>
<td>LRCB 23 STEPS.</td>
</tr>
<tr>
<td>I24</td>
<td>015002</td>
<td>LRCB 24 STEPS.</td>
</tr>
<tr>
<td>I25</td>
<td>015002</td>
<td>LRCB 25 STEPS.</td>
</tr>
<tr>
<td>I26</td>
<td>016002</td>
<td>LRCB 26 STEPS.</td>
</tr>
<tr>
<td>I27</td>
<td>017002</td>
<td>LRCB 27 STEPS.</td>
</tr>
<tr>
<td>I28</td>
<td>017002</td>
<td>LRCB 28 STEPS.</td>
</tr>
<tr>
<td>I29</td>
<td>018002</td>
<td>LRCB 29 STEPS.</td>
</tr>
<tr>
<td>I30</td>
<td>019002</td>
<td>LRCB 30 STEPS.</td>
</tr>
<tr>
<td>I31</td>
<td>020002</td>
<td>LRCB 31 STEPS.</td>
</tr>
<tr>
<td>I32</td>
<td>021002</td>
<td>LRCB 32 STEPS.</td>
</tr>
<tr>
<td>Address</td>
<td>Value</td>
<td>Comment</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>133</td>
<td>001405</td>
<td>CCITT 3 STEPS.</td>
</tr>
<tr>
<td>134</td>
<td>002005</td>
<td>CCITT 4 STEPS.</td>
</tr>
<tr>
<td>135</td>
<td>002405</td>
<td>CCITT 5 STEPS.</td>
</tr>
<tr>
<td>136</td>
<td>003005</td>
<td>CCITT 6 STEPS.</td>
</tr>
<tr>
<td>137</td>
<td>003405</td>
<td>CCITT 7 STEPS.</td>
</tr>
<tr>
<td>140</td>
<td>004005</td>
<td>CCITT 10 STEPS.</td>
</tr>
<tr>
<td>141</td>
<td>004400</td>
<td>CRC12 1 STEP.</td>
</tr>
<tr>
<td>142</td>
<td>006000</td>
<td>CRC12 2 STEPS.</td>
</tr>
<tr>
<td>143</td>
<td>004000</td>
<td>CRC12 3 STEPS.</td>
</tr>
<tr>
<td>144</td>
<td>002000</td>
<td>CRC12 4 STEPS.</td>
</tr>
<tr>
<td>145</td>
<td>002400</td>
<td>CRC12 5 STEPS.</td>
</tr>
<tr>
<td>146</td>
<td>003000</td>
<td>CRC12 6 STEPS.</td>
</tr>
<tr>
<td>147</td>
<td>006000</td>
<td>CRC12B ONE BYTE DATA.</td>
</tr>
<tr>
<td>150</td>
<td>00112</td>
<td>CRCB ONE BYTE DATA.</td>
</tr>
<tr>
<td>151</td>
<td>003000</td>
<td>CRCB ONE BYTE DATA.</td>
</tr>
<tr>
<td>152</td>
<td>003000</td>
<td>CRCB ONE BYTE DATA.</td>
</tr>
<tr>
<td>153</td>
<td>002000</td>
<td>CRCB ONE BYTE DATA.</td>
</tr>
<tr>
<td>154</td>
<td>002400</td>
<td>CRCB ONE BYTE DATA.</td>
</tr>
<tr>
<td>155</td>
<td>003000</td>
<td>CRCB ONE BYTE DATA.</td>
</tr>
<tr>
<td>156</td>
<td>002400</td>
<td>CRCB ONE BYTE DATA.</td>
</tr>
<tr>
<td>157</td>
<td>002400</td>
<td>CRCB ONE BYTE DATA.</td>
</tr>
<tr>
<td>160</td>
<td>000102</td>
<td>RESERVED.</td>
</tr>
<tr>
<td>161</td>
<td>000102</td>
<td>RESERVED.</td>
</tr>
<tr>
<td>162</td>
<td>000102</td>
<td>RESERVED.</td>
</tr>
<tr>
<td>163</td>
<td>000102</td>
<td>RESERVED.</td>
</tr>
<tr>
<td>164</td>
<td>000102</td>
<td>RESERVED.</td>
</tr>
<tr>
<td>165</td>
<td>000102</td>
<td>RESERVED.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DATA WORD TABLE</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>-----------------</td>
</tr>
<tr>
<td>617</td>
<td></td>
<td>D01: 000001</td>
</tr>
<tr>
<td>616</td>
<td></td>
<td>D02: 000002</td>
</tr>
<tr>
<td>615</td>
<td></td>
<td>D03: 000004</td>
</tr>
<tr>
<td>614</td>
<td></td>
<td>D04: 000010</td>
</tr>
<tr>
<td>613</td>
<td></td>
<td>D05: 000020</td>
</tr>
<tr>
<td>612</td>
<td></td>
<td>D06: 000040</td>
</tr>
<tr>
<td>611</td>
<td></td>
<td>D07: 000100</td>
</tr>
<tr>
<td>610</td>
<td></td>
<td>D08: 000500</td>
</tr>
<tr>
<td>609</td>
<td></td>
<td>D09: 001000</td>
</tr>
<tr>
<td>608</td>
<td></td>
<td>D10: 002000</td>
</tr>
<tr>
<td>607</td>
<td></td>
<td>D11: 004000</td>
</tr>
<tr>
<td>606</td>
<td></td>
<td>D12: 008000</td>
</tr>
<tr>
<td>605</td>
<td></td>
<td>D13: 010000</td>
</tr>
<tr>
<td>604</td>
<td></td>
<td>D14: 020000</td>
</tr>
<tr>
<td>603</td>
<td></td>
<td>D15: 040000</td>
</tr>
<tr>
<td>602</td>
<td></td>
<td>D16: 080000</td>
</tr>
<tr>
<td>601</td>
<td></td>
<td>D17: 100000</td>
</tr>
<tr>
<td>600</td>
<td></td>
<td>D18: 177776</td>
</tr>
<tr>
<td>599</td>
<td></td>
<td>D19: 177776</td>
</tr>
<tr>
<td>598</td>
<td></td>
<td>D20: 177776</td>
</tr>
<tr>
<td>597</td>
<td></td>
<td>D21: 177776</td>
</tr>
<tr>
<td>596</td>
<td></td>
<td>D22: 177776</td>
</tr>
<tr>
<td>595</td>
<td></td>
<td>D23: 177776</td>
</tr>
<tr>
<td>594</td>
<td></td>
<td>D24: 177776</td>
</tr>
<tr>
<td>593</td>
<td></td>
<td>D25: 177776</td>
</tr>
<tr>
<td>592</td>
<td></td>
<td>D26: 177776</td>
</tr>
<tr>
<td>591</td>
<td></td>
<td>D27: 177776</td>
</tr>
<tr>
<td>590</td>
<td></td>
<td>D28: 177776</td>
</tr>
<tr>
<td>589</td>
<td></td>
<td>D29: 177776</td>
</tr>
<tr>
<td>588</td>
<td></td>
<td>D30: 177776</td>
</tr>
<tr>
<td>587</td>
<td></td>
<td>D31: 177776</td>
</tr>
<tr>
<td>586</td>
<td></td>
<td>D32: 177776</td>
</tr>
<tr>
<td>585</td>
<td></td>
<td>D33: 177776</td>
</tr>
<tr>
<td>584</td>
<td></td>
<td>D34: 177776</td>
</tr>
<tr>
<td>583</td>
<td></td>
<td>D35: 177776</td>
</tr>
<tr>
<td>582</td>
<td></td>
<td>D36: 177776</td>
</tr>
<tr>
<td>581</td>
<td></td>
<td>D37: 177776</td>
</tr>
<tr>
<td>580</td>
<td></td>
<td>D38: 177776</td>
</tr>
<tr>
<td>579</td>
<td></td>
<td>D39: 177776</td>
</tr>
<tr>
<td>578</td>
<td></td>
<td>D40: 177776</td>
</tr>
<tr>
<td>577</td>
<td></td>
<td>D41: 177776</td>
</tr>
<tr>
<td>576</td>
<td></td>
<td>D42: 177776</td>
</tr>
<tr>
<td>575</td>
<td></td>
<td>D43: 177776</td>
</tr>
<tr>
<td>574</td>
<td></td>
<td>D44: 177776</td>
</tr>
<tr>
<td>573</td>
<td></td>
<td>D45: 177776</td>
</tr>
<tr>
<td>572</td>
<td></td>
<td>D46: 177776</td>
</tr>
<tr>
<td>571</td>
<td></td>
<td>D47: 177776</td>
</tr>
<tr>
<td>570</td>
<td></td>
<td>D48: 177776</td>
</tr>
</tbody>
</table>
; :EQUATES, CONSTANTS AND VARIABLES

RO=0:
R1=0:
R2=2:
R3=3:
R4=4:
RS=5:
SP=6:
PC=7:
R7=PC:
R=PC:

SUR: 177570:
PS: 177776:
SCOE=TRAP:
HLT=ENT:
NOP=240:

TCICER= 177560:
TXDNR= 177562:
TPDNR= 177566:

INTBEG=101:
INTEND=157:
DRIEG=901:
DATEN=577:

DEVADR: 170700:
DFLAG: 0:
DFLAG: 0:
EFLAG: 0:
HDONE: 0:
IFLAG: 0:
ITACNT: 0:
ITACNT: 0:
OPSTAT: 0:
OCTACNT: 0:
OTWAKA: 0:
OTWAKB: 0:
SELFAC: 0:
SCORTN: 0:
SMBCC: 0:
SMBCDA: 0:
SMCNT: 0:
SMDATA: 0:
SMDONE: 0:
SMWKA: 0:
SMWKB: 0:
STCNT: 0:
TEKW: 0:

; TITLE PRINTED = 1
IDENT: RESET

MOV #1000, SP ; IN ACTI1 AUTOMATIC MODE?
CMP #44, #46
BEQ INIT
STI TITLE ; HAS TITLE BEEN PRINTED BEFORE?
BNE INIT ; IN ACTI1 AUTOMATIC MODE?
JSR RS, TYPE ; TYPE THE PROG. NAME.

ASCI1 /177/ K11A - CYCLIC REDUNDANCY CHECK TEST, MC-11-DZKGA-B/177/13/

MOV #1, TITLE ; SET FLAG
INIT: CLR SELFLG
CLR INTB ; CLEAR SELFLG (START WAS FROM LOC. 200.)
BR INTB ; BRANCH.
INTA: INC SELFLG
INTB: CLR SELFLG (START WAS FROM LOC. 204.)

ARE WE IN ACTI1 AUTOMATIC MODE?

NEVER MIND ABOUT SWR

CHECK FOR HARDWARE SWITCH REGISTER
R3 + DEVICE ADDRESS.

ADJUST R3 TO POINT AT BCC.

READ SWR & SET UP PROG. ACCORDINGLY.

START FROM LOC. 200?

BRANCH.

NO. - JUMP.

STORE SWR.

SET "T" BIT IN PS.

INHIBIT TRACE TRAP.

NO. - BRANCH.

YES. - CLEAR "T" BIT IN PS.

SELECT MODE (START WAS FROM 202)?

YES. - BRANCH.

NO. - BRANCH.

YES. - CLEAR ITERATE COUNTER

RETURN.
FULL TEST MODE. CONTROL THE TESTING OF ALL KG11 INSTRUCTIONS.
SELECT TEST MODE. CONTROL THE TESTING OF THE INSTRUCTION SELECTED 
ON SWRS-0 (DO-ALL INST.) USING THE DATA WORD SELECTED ON SWRII-6.

SELMOD: MOV #INTEG,R1 
INST. POINTER (R1) SET TO THE INST.

MOV OPSTAT,IFLG 
;SELECTED ON SR BITS 5-0.

BIC #1770D,IFLG 

MOV IFLG,IFLGA 

SEL: MOV #DRTBEG,R2 
;DATA POINTER (R2) SET TO THE DATA

MOV OPSTAT,DFLGA 

;WORD SELECTED ON SR BITS 11-6

RDL: MOV DFLO,DFLGA 

BEQ SLEB 

ROR DFLG 

ROR DFLG 

ROR DFLG 

ROR DFLG 

ROR DFLG 

SUB #2,DFLGA 

ADD DFLGA,DFLGA 

JSR PC,CKSR 

;CHECK FOR CNTL 0 TO LOAD SWREG

JSR R5,*+2(R1) 

;STEP COUNT IF ANY) SET FOR SIMULATE.

SMCNT R5 

;SIMULATED BCC SET TO 0

ROR R5 

;SET THE SIMULATED DONE BIT.

JSR R5,*R(I) 

;SIMULATE INST. (VIA R5) ON DATA (VIA R2).

BNE 

IS THE INST. A STEP TYPE?

BNE 

;NO - BRANCH.

SEL: MOV #200,SMONE 

;BIT 200, SMONE

JSR R5,TESTA 

;SET AND IMPRT IN ASSISTANT.

EB 

;SIMULATE DONE BIT.

JSR R5,TESTB 

;DO TEST A (STEP INST TEST).

SEL: BR 

;BRANCH.

JSR R5,TESTB 

;BRANCH.

JSR R5,OPCHG 

;IS DATA FLAG NON-ZERO?

BNE SELH 

;YES. - DATA POINTER (R2) IS FROZEN. - BRANCH.

SELH 

;NO. - HAVE ALL DATA WORDS BEEN TESTED?

BNE SELF 

;NO. - BRANCH.

SEL: BR 

;IS DATA POINTER (R2) SET TO BEGIN OF DATA TABLE.

ADD #2,R2 

BR SELB 

;YES. - INST. POINTER (R1) IS FROZEN (BRANCH).

ADD #1,R2 

BR SELH 

;NO. - HAVE ALL INST. BEEN TESTED?
898: BEQ SELI ; "ES. - BRANCH.
899: ADD #4,R1 ; "NO. - ADVANCE INST. POINTER (R1)."
900: BR SELB ; "BRANCH.
901: SELI: MOV #INTBEG,R1 ; INST. POINTER (R1) SET TO BEGIN OF INST. TABLE.
902: BR SELB ; "BRANCH."
TEST A:

TEST THE INSTRUCTION POINTED AT BY R1 USING THE DATA WORD POINTED AT BY R2. (HIGHEST BYTE OF THE INST. WORD IS THE STEP COUNT & LOW BYTE IS THE ACTUAL INST.) THE FINAL RESULT IS STORED INTO LOC. "HBCC" TO BE COMPARED WITH THE SIMULATED RESULT WHICH HAS BEEN STORED IN LOC. "SMBC".

TESTA:  SCOPE:  GO RECORD ENTRY TO THIS TEST.
         MOV A +1(R1), STCNT
         MOV B #2D, -2(R3)
         MOV (R1), -2(R3)
         MOV (R3), #4D, -2(R3)
         MOV TAA
         BNE TAA
         NO - BRANCH.
         BNE TAB
         NO - BRANCH. (ERROR)
         CMP HBCC, SMBC
         BEQ TAB
         YES - BRANCH.
         CMP HBCC, SMBC
         BEQ TAC
         YES - BRANCH.
         CMP HBCC, SMBC
         BEQ TAC
         YES - BRANCH.
         TAB:  HLT
         TAB:  TST EFLG
         BNE TAB
         YES - BRANCH.
         RTS R5
         NO - RETURN.

           TESTA:
            SCOPE:
            MOV A +1(R1), STCNT
            MOV B #2D, -2(R3)
            MOV (R1), -2(R3)
            MOV (R3), #4D, -2(R3)
            MOV TAA
            BNE TAA
            NO - BRANCH.
            BNE TAB
            NO - BRANCH. (ERROR)
            CMP HBCC, SMBC
            BEQ TAB
            YES - BRANCH.
            CMP HBCC, SMBC
            BEQ TAC
            YES - BRANCH.
            CMP HBCC, SMBC
            BEQ TAC
            YES - BRANCH.
            TAB:  HLT
            TAB:  TST EFLG
            BNE TAB
            YES - BRANCH.
            RTS R5
            NO - RETURN.
; TEST B.
; TEST THE INSTRUCTION POINTED AT BY R1 USING THE DATA WORD
; POINTED AT BY R2. THE RESULT IS STORED IN LOC. "MBCC" TO BE
; COMPARED WITH THE SIMULATED RESULT WHICH HAS BEEN STORED
; IN LOC. "SMBCC".

TESTB:  SCOPE
GO RECORD ENTRY TO THIS TEST.
CLEAR THE BCC.

10440  MOV  #20,-2(R3)  MOVE INST. TO CSR.
10446  MOV  (R1),-2(R3)  MOVE DATA TO DBR (COMPUTE BCC).
10452  MOV  (R2),+2(R3)  PAUSE.
10458  NOP

10464  MOV  -2(R3),HDONE  READ & STORE CSR (DONE BIT)
10470  MOV  (R3),MBCC  READ & STORE BCC.
10476  MOV  (R3),HBBCC  IS DONE BIT = 1?
10482  BIT  #200,HDONE  YES. - BRANCH.
10488  BEQ  TBA  NO. - BRANCH. (ERROR)
10494  CMP  MBCC,SMBCC  HARDWARE BCC = SIMULATED BCC?
1049A  BEQ  TBB  YES. - BRANCH.
104A6  BEQ  TBA  NO. - TRAP TO ERROR HANDLER.
104B2  CMP  EFLG  IS AN ERROR SCOPE LOOP RUNNING.
104B8  BNE  TBA  YES. - BRANCH.
104C4  RTS  R5  NO. - RETURN.
;TRAP SERVICE ROUTINE (CALLED BY PSEUDO-OPT VR0)
;ERROR HANDLER - TYPEOUTS - HALT - SCOPE LOOP.

954 003154 004767 001564 ERR: JSR PC,CKSWR
955 003160 005767 176416 TST EFLG
956 003165 001003 BNE ERR
957 003166 006267 176410 INC EFLG
958 003167 002144 BRC ERR
959 003174 004004 Q20000 176410 ERR: BIT #20000,OPSTAT
960 003200 001002 BNE EFLG
961 003204 004567 000140 ERR: JSR RS,TYPEERR
962 003221 023737 000420 000046 ERE: CMP #042,046
963 003226 014046 BEQ +12
964 003254 001404 BEQ EFLG
965 003260 014004 BEQ ERR
966 003286 002800 HALT
967 003288 000000 ERR: JSR PC,CKSWR
968 003296 004767 176330 176354 CMP #35R,OPSTAT
969 003296 001402 BEQ ERR
970 003296 004567 176622 JSR RS,OPCHE
971 003296 001403 ERE: BIT #40000,OPSTAT
972 003296 026256 BEQ EFLG
973 003300 00177 BNE SELMOD
974 003300 00177 CLR EFLG
975 003300 00177 TST SELFGL
976 003300 001001 BNE ERR
977 003300 000002 NO - RETURN (CONTINUE TESTING).
978 003312 001410 JSR #77,35WR
979 003314 000007 176260 ERR: BIT #77,35WR
980 003314 001404 BEQ ERR
981 003314 001001 NO - RETURN (CONTINUE TESTING).
982 003314 000770 176250 JSR #770D,35WR
983 003314 001002 BEQ ERR
984 003314 001002 NO - RETURN (CONTINUE TESTING).
985 003314 000006 JSR #6,SP
986 003314 001002 NO - RETURN (CONTINUE TESTING).
987 003314 001002 REALAT SLOD
988 003314 000000 JSR #177064
989 003314 000000 ERR: RTI
990 003314 000000 FORCER: RESET
991 003316 000000 JSR RS,TYPEERR
992 003333 000004 FER: HALT
993 003340 004567 000004 JSR RS,TYPEERR
994 003340 000004 THEN HALT.
;SUBROUTINE - SET UP AND TYPE THE ERROR MESSAGE THAT STARTS AT EMA.

TYPEERR: TSTB  +1(R1)
BEQ  TER
MOV  #20101,EMC
GR  TEB
TER
MOV  #20102,EMC
TEB.

MOV  R1,TEWK
SUB  #ATBEG-4,TEWK
ROR  TEBK

BIC  #177700,TEWK
JSR  R5,0TA

EMF

2
MOV  R2,TEWK
SUB  #DATBEG-2,TEWK
ROR  TEBK

BIC  #177700,TEWK
JSR  R5,0TA

TEWK

EMH+1

2
JSR  R5,0TA

SMBC

EMM

6
JSR  R5,0TA

MBC

EMQ+1

6
JSR  R5,0TA

MOV  #61,EMX+1
BIT  #200,SMDONE
BNE  TEC

MOV  #60,EMX+1

BIT  #200,HDONE
BNE  TED

MOV  #60,EMY+1

TED:  JSR  R5,TYP

EM:  .ASCII  /TE/
EM:  .ASCII  /ST/
EM:  .ASCII  /E/
EM:  .ASCII  /E/
EM:  .ASCII  /E/
EM:  .ASCII  /O/
<table>
<thead>
<tr>
<th>EMH</th>
<th>ASCII</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMI</td>
<td>ASCII</td>
<td>/S/</td>
</tr>
<tr>
<td>EMJ</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMK</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMM</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMN</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMO</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMP</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMQ</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMR</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMS</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMU</td>
<td>ASCII</td>
<td>/E/</td>
</tr>
<tr>
<td>EMV</td>
<td>ASCII</td>
<td>/BI/</td>
</tr>
<tr>
<td>EMX</td>
<td>ASCII</td>
<td>//</td>
</tr>
<tr>
<td>EMY</td>
<td>ASCII</td>
<td>/++/</td>
</tr>
<tr>
<td>EMZ</td>
<td>ASCII</td>
<td>RTS</td>
</tr>
</tbody>
</table>

**803**

2ZKGA-B KG11-A CYCLIC REDUNDANCY CHECK TEST

MACY11 27(1006) 25-OCT-76 22:42 PAGE 27

2ZKGA.B.P11 25-OCT-76 22:40

| 047 | 003612 | 020077 |  
| 048 | 003614 | 051440 |  
| 049 | 003618 | 027977 |  
| 050 | 003620 | 027977 |  
| 051 | 003622 | 027977 |  
| 052 | ASCII | /++/ |  
| 053 | 003626 | 020077 |  
| 054 | 003628 | 037950 |  
| 055 | 003630 | 037977 |  
| 056 | 003632 | 020077 |  
| 057 | ASCII | /++/ |  
| 058 | 003636 | 020077 |  
| 059 | 003640 | 047117 |  
| 060 | 003642 | 020105 |  
| 061 | 003644 | 044508 |  
| 062 | 003644 | 051294 |  
| 063 | 003650 | 037954 |  
| 064 | 003654 | 057536 |  
| 065 | 003656 | 050205 |  

RTS

RE
SUBROUTINE - TYPE ON THE TTY THE MESSAGE IMMEDIATELY FOLLOWING
THE CALL TO THIS SUBROUTINE. UP-ARROW (↑) CAUSES A CR LF AND BACK-ARROW
(→) CAUSES TERMINATION OF TYPEOUT. RETURN WILL BE TO THE INSTRUCTION
FOLLOWING THE MESSAGE.

TYPE:

TSTB  TPDSR
BPL  TYPE
CMPB  (RS),#136
BNE  TPA
TSTB  (RS)+
JSR  RS,TYPE
005015
000137

TPA:

CMPB  (RS),#137
BEQ  TPB
MOV  (RS)+,TPDBR
BR  TYPE

TPB:

TSTB  (RS)+
BIT  #1,RS
BEQ  TPC
INC  RS

TPC:

RTS  RS
; SUBROUTINE - OCTAL TO ALPHA CONVERSION ACCORDING TO THE THREE WORDS
; IMMEDIATELY FOLLOWING THE CALL TO THIS SUBROUTINE, THE FIRST WORD POINTS
; AT THE WORD TO BE CONVETED. THE SECOND WORD POINTS AT THE STARTING
; LOC. WHERE THE RESULT IS TO BE MOVED. THE THIRD WORD IS THE NUMBER
; OF CHARACTERS TO BE CONVERTED AND MOVED. RETURN WILL BE TO THE
; INSTRUCTION FOLLOWING THESE THREE WORDS.

003736 012567 175662  OTR: MOV 3:(R5)+, OTAWKA
003737 012504  OTR: MOV (R5)+, R4
003738 012567 175652  OTR: MOV (R5)+, OTACNT
003739 016767 175650 175650  OTALA: MOV OTAWKA, OTAWKB
1076 000241  CLC
1077 000241  ROR OTAWKA
1078 006067 175640
1079 006067 175632
1080 006067 175632
1081 006067 175632
1082 006067 175632
1083 006067 175632
1084 006067 175632
1085 006067 175632
1086 006067 175632
1087 006067 175632
1088 006067 175632
1089 006067 175632
1090 006067 175632
1091 006067 175632
1092 006067 175632
1093 006067 175632
1094 006067 175632
1095 006067 175632
1096 006067 175632
1097 006067 175632
1098 006067 175632
1099 006067 175632
1100 006067 175632
1101 006067 175632
1102 006067 175632
1103 006067 175632
1104 006067 175632
1105 006067 175632
1106 006067 175632
1107 006067 175632
1108 006067 175632
1109 006067 175632
1110 006067 175632
1111 006067 175632
1112 006067 175632
1113
; TRAP SERVICE ROUTINE - SCOPE = TRAP.

SCORCD: MOV (SP), SCORTN ; SAVE THE STARTING LOC. OF TESTA OR TESTB.
JSR PC, CKSWR ; CHECK FOR_CNTL G TO LOAD SWREG
RTI ; RETURN.

; TRAP SERVICE ROUTINE - "T" BIT SET = TRAP.

TRTRTN: RTI ; GOOD BOY, NOW RETURN.
SUBROUTINE - SIMULATE ALL MODES OF LRC

SMLRCQ: MOV #10,SMCNT ;ENTRY LRC16 DOUBLE BYTE DATA.
SMLRCH: MOV #10,SMCNT ;ENTRY LRC16 SINGLE BYTE DATA.
SMLRCC: ADD #100000,SMLRCQ ;ENTRY STEP LRC16.
SMLRCD: MOV #10,SMCNT ;ENTRY LRCB DOUBLE BYTE DATA.
SMLRCE: ADD #10,SMCNT ;ENTRY LRCB SINGLE BYTE DATA.
SMLRCF: BNE SMINT ;ENTRY STEP LRCB.
SMLRCH: TST SMCNT
SMRCG: BNE SMINT
SMINT: INC SMCNT
SMBC,SMBCA
MOV Z,(R2),SMDATA
SMBC,SMKA
MOV Z,(SMDATA),SMWKA
ROR SMWKA
SMRCJ: ROR SMBC ;INSTR. MODIFIED: LRC16=ROR LRCB=RORB.
CLC
SMDATA
DEC SMCNT
BNE SMLRCI
RTS RS
; SUBROUTINE - SIMULATE ALL MODES OF CRC16.
004200 012767 000010 175432 SMC16A: MOV 10,SMCNT ;ENTRY CRC16 DOUBLE BYTE DATA.
004205 062767 000010 175424 SMC16B: ADD 10,SMCNT ;ENTRY CRC16 SINGLE BYTE DATA.
004214 000767 175420 SMC16C: TST SMCNT ;ENTRY STEP CRC16.
004220 001002 INCK SMC16D
004226 000767 175412 SMC16E: MOV SMBCC,SMBCCA
004234 011267 175402 SMC16F: MOV (R2),SMDATA
004240 016767 175400 SMC16G: MOV SMBCC,SMWKA
004246 066767 175370 SMC16H: ADD SMDATA,SMWKA
00424C 002414 175352 SMC16I: CLR
004252 002414 175352 SMC16J: ROR SMBCC
004258 006067 175352 SMC16K: ROR SMDATA
00425E 006067 175352 SMC16L: ROR SMWKA
004264 010314 SMC16M: BCC SMC16F
00426A 012767 120001 175342 SMC16N: MOV #120001,SMWKA
004274 046767 175324 SMC16O: BIC SMBCC,SMWKA
004280 046767 175324 SMC16P: BIC SMBCC,SMWKA
004286 066767 175306 SMC16Q: ADD SMWKA,SMBCCA
004292 009367 175306 SMC16R: DEC SMCNT
00429E 000205 SMC16S: RTS RS

; SUBROUTINE - SIMULATE ALL MODES OF CCITT.
004336 012767 000010 175274 SMC1T: MOV 10,SMCNT ;ENTRY CCITT DOUBLE BYTE DATA.
004341 062767 000010 175266 SMC1T2: ADD 10,SMCNT ;ENTRY CCITT SINGLE BYTE DATA.
004350 000767 175262 SMC1T3: TST SMCNT ;ENTRY STEP CCITT.
004356 001002 INCK SMC1T4
004360 000767 175254 SMC1T5: MOV SMBCC,SMBCCA
004366 011267 175244 SMC1T6: MOV (R2),SMDATA
004372 016767 175232 SMC1T7: MOV SMBCC,SMWKA
004378 066767 175234 SMC1T8: ADD SMDATA,SMWKA
004384 000241 175214 SMC1T9: CLR
00438A 000241 175214 SMC1T10: ROR SMBCC
00438E 006067 175214 SMC1T11: ROR SMDATA
004394 006067 175214 SMC1T12: ROR SMWKA
00439A 012767 103014 SMC1T13: BCC SMC1T9
00439E 012767 103014 SMC1T14: MOV #102010,SMWKA
004402 046767 175166 175176 SMC1T15: BIC SMBCC,SMWKA
004408 042767 102010 175156 SMC1T16: MOV SMBCC,SMWKA
004414 042767 102010 175156 SMC1T17: BIC SMBCC,SMWKA
004420 066767 175150 SMC1T18: ADD SMWKA,SMBCCA
004426 000367 175150 SMC1T19: DEC SMCNT
004432 001342 SMC1T20: RTS RS

; SUBROUTINE - SIMULATE ALL MODES OF CRC12.
004474 012767 000006 175136 SMC12A: MOV 6,SMCNT ;ENTRY CRC12 SINGLE BYTE DATA.
004479 005767 175132 SMC12B: TST SMCNT ;ENTRY STEP CRC12.
;SWITCH REGISTER SIZING ROUTINE

SUSWR: MOV @#6,-(SP) ;SAVE VECTORS
       MOV @#4,-(SP)
       MOV @#6,$A4
       CMP #$F1,$A4
       BR $66

64$: CMP (SP)+,(SP)+ ;ADJUST STACK
       MOV $SREG,SWR
       MOV (SP)+,#4
       MOV (SP)+,#0
       CMP $SREG,SWR
       BNE $67
       JSR PC,CNTLU ;ALLOW SWREG TO BE LOADED
       RTS PC

;CHECK SWITCH REGISTER ROUTINE. CHECKS FOR TG TO ALLOW CHANGING
;OF LOC.176.

;LOCATIONS USED:
;TEMPST: WORD 0
;COUNT: WORD 0
;TIB: WORD 0

004744 022767 000176 174620 DKSWR: CMP $SREG,SWR ;SOFTWARE SWITCH REGISTER PRESENT
       BNE OUT
       TSTB TCSR
       BPL OUT
       MOV TDBR,TIB
       BIC $17800,TIB
       CMP #7,TIB
       BNE OUT
       JSR RS,TYPE ;TYPE CNTL G
       .ASCIIZ /"CNTL G"/

005024 004567 16706 CNTLU: JSR RS,OTA ;MOVE CONTENTS
      SWREG RS,OTA
      MSREG
      JSR RS,TYPE ;TYPE THE COMPLETE MESSAGE

005042 05136 051127 075 MSWR: .ASCIIZ /"MSWR="/ SWR= XXXXXX
      .ASCIIZ /"=="/ 

005070 057490 .EVEN

005070 057490 .EVEN
$READ: CLR TEMPST  
MOV $7,COUNT  
JSR PC,TIN  
;GO READ A CHARACTER  
STRIP OFF GARBAGE  
;IS IT A 1U?  
BRANCH IF NOT  
;START OVER  
IS IT A (CR)?  
BRANCH IF NOT  
;TYPE LF,CR  
ASCII /**  
;WAS IT FIRST CHARACTER  
;CHANGE SWR IF NOT FIRST ONE  
GET OUT  
;START OVER IF NOT LEGAL CHARACTER  
ASCII /**  
;GET NITTY-GRITTY  
;ONLY WANT 6 DIGITS  
;CHANGE SWITCH REGISTER CONTENTS  
RETURN TO PROGRAM
; TTY INPUT SUBROUTINE*********

005264 005067 17227D
005270 005067 172266
005274 005067 172242
005300 005267 172254
005304 105767 172250
005310 100375
005312 016767 172244 177422
005320 105767 172240
005324 100375
005326 116767 177410 172232
005334 000207
000001

; .END
M03

ZKGAB-B KG11-A CYCLIC REDUNDANCY CHECK TEST MACY11 27:1006 25-OCT-76 22:42 PAGE 39
ZKGAB.P11 25-OCT-76 23:40 SYMBOL TABLE

STCMT DC1652 TBA 003142 TESTA 002750 TPB 003722 TTIN2 005320
SUSMC DC4650 TBB 003144 TESTB 003066 TPC 003734 TYPE 003666
SMR DC1672 TCB 003366 TEBK 001654 TPCSR = 177564 TYPERR 003350
SWREG DC0170 TCB 003374 TIB 004742 TPDBR = 177566 "$ENDRO 002906
RRA DC2776 TEC 003544 TKCSR = 177560 TRTRTN 004042 "$READ 005076
TAB DC3054 TED 003570 TKDBR = 177562 TTIN 005264 . = 005336
RC DC3056 TEMPST 004736 TPA 003706 TTINI 005304

ABS. 005336 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

DSKZ:ZKGAB,DSKZ:ZKGAB/SOL=DSKZ:ZKGAB.P11
RUN-TIME: 4.91 SECONDS
RUN-TIME RATIO: 145/14=10.0
CORE USED: 5K (9 PAGES)