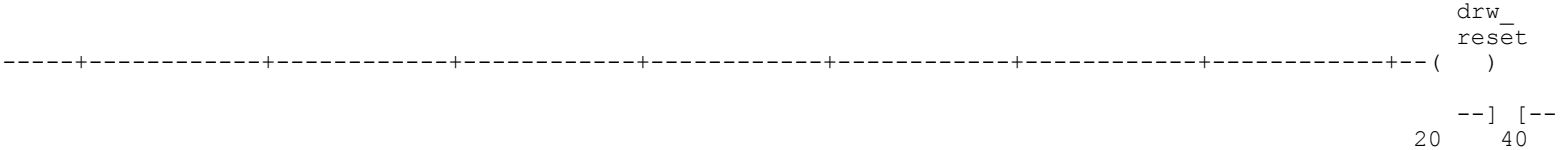
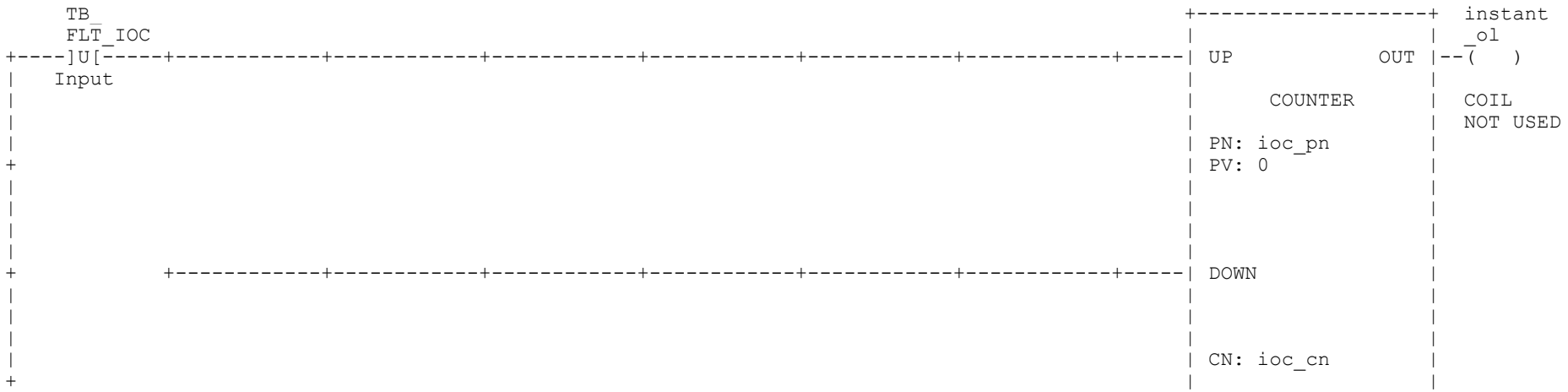
Sequence 20

Sequence 30



Sequence 40



```
|                                                                                                                  |                      |
|    drw_                                                                                                          |                      |
|    reset                                                                                                         |                      |
+----] [-----+------------+------------+------------+------------+------------+------------+-----| RESET                |
      30                                                                                                          |                      |
                                                                                                                  +--------------------+
```

Task Source Code Last Edit Date:19-Nov-10 09:00:50

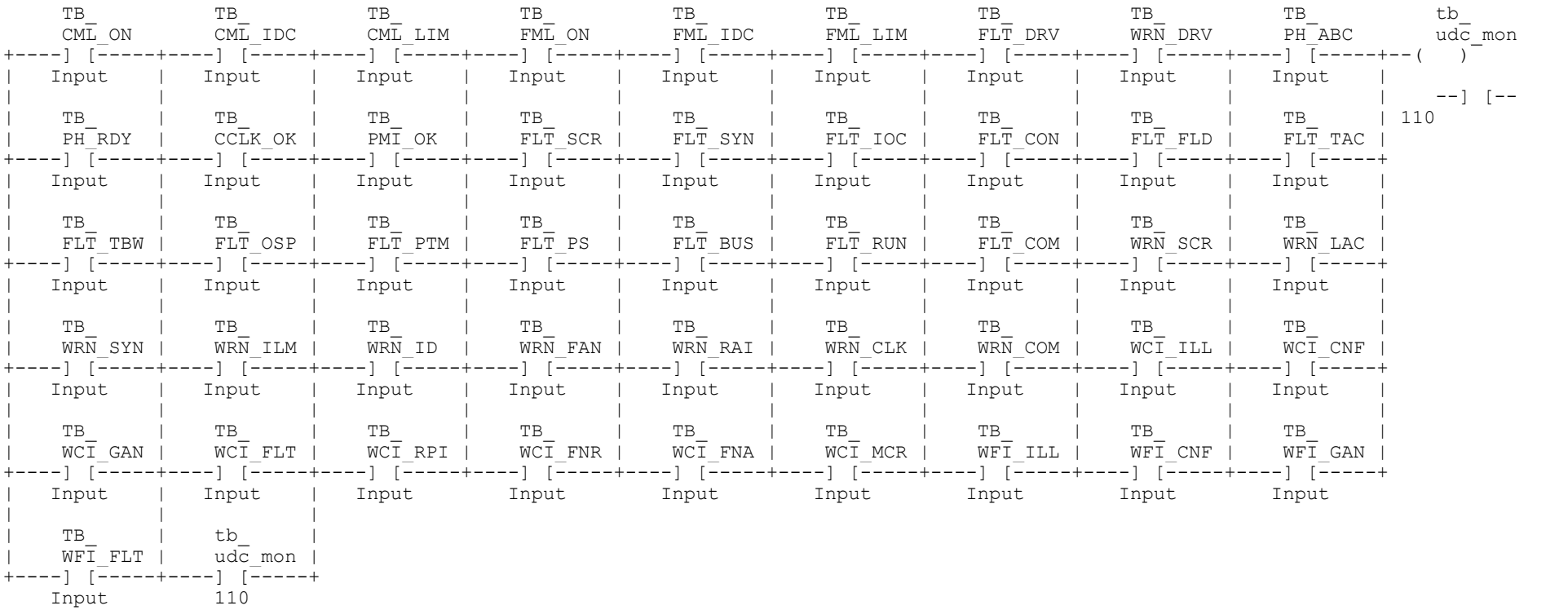
Sequence 50

Sequence 100 REMARKS Keyname: XXA

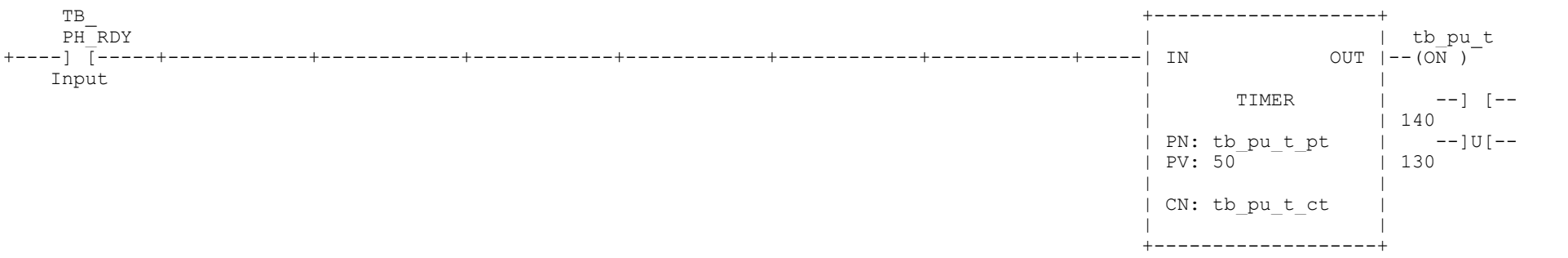
```
!XXA
************************************************
Thrust Buggy Drive Logic
************************************************
tb_udc_mon    UDC Monitor
tb_reset      Reset
tb_flt_ol     Application Faults
tb_sprm       Permissives / Interlocks
tb_test       Test Sequences
tb_jogrbr     Jog
tb_runrbr     Run
TB_CML_RUN    CML
TB_FML_RUN    FML
TB_SPD_ON     Speed Loop
************************************************
```

Task Source Code Last Edit Date:19-Nov-10 09:00:50

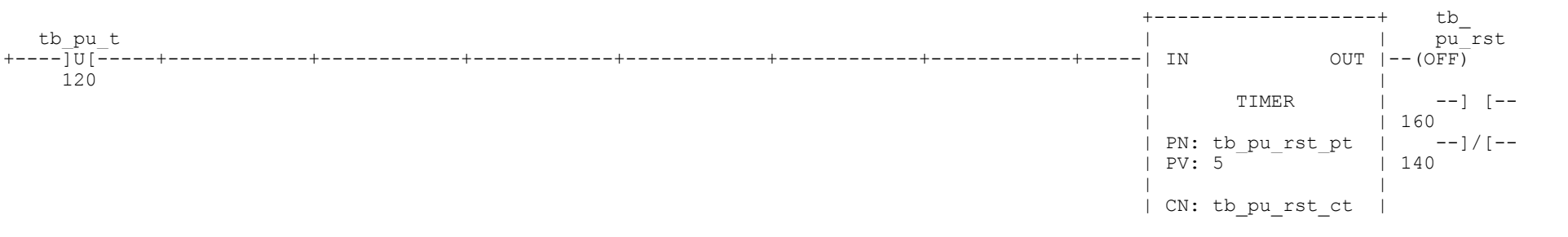
Sequence 110 THR BUGGY UDC MONITOR

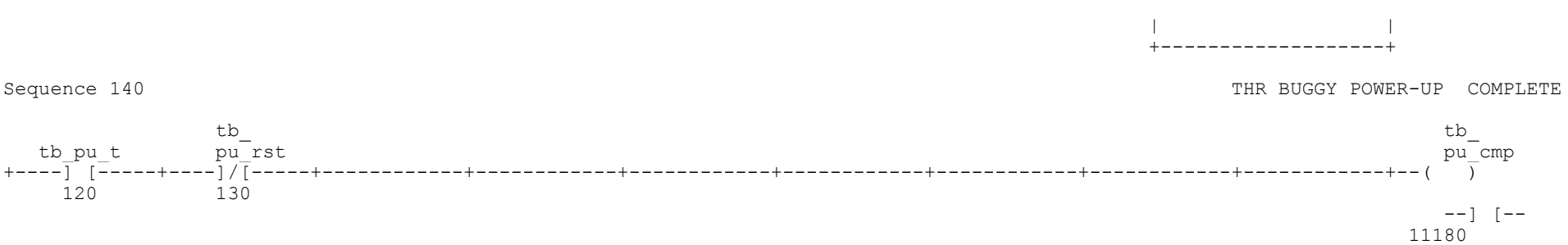


Sequence 120 THR BUGGY POWER-UP TIMER



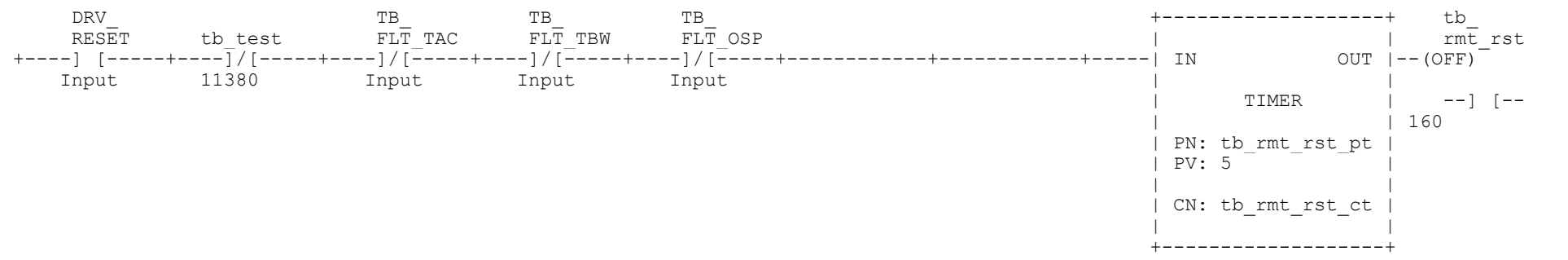
Sequence 130 THR BUGGY POWER-UP RESET



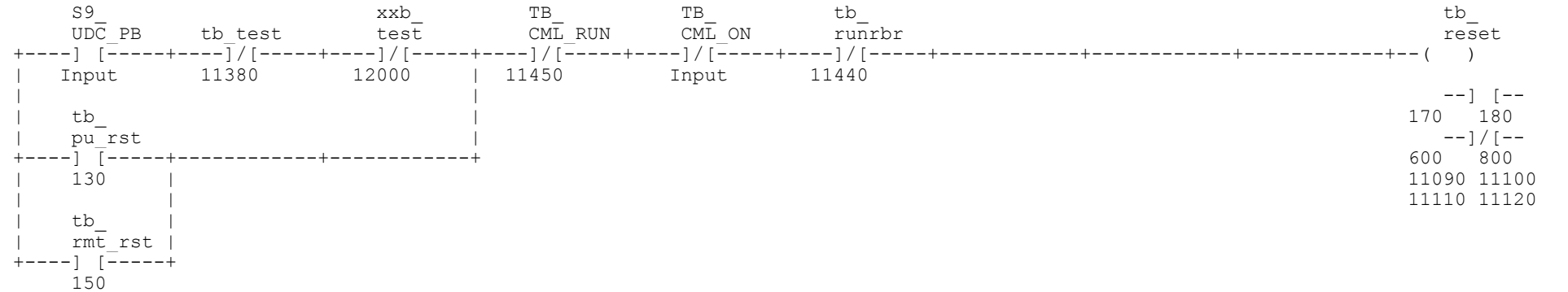

Sequence 140
THR BUGGY POWER-UP COMPLETE
tb_pu_t
120
tb_pu_rst
130
tb_pu_cmp
11180

Task Source Code Last Edit Date:19-Nov-10 09:00:50

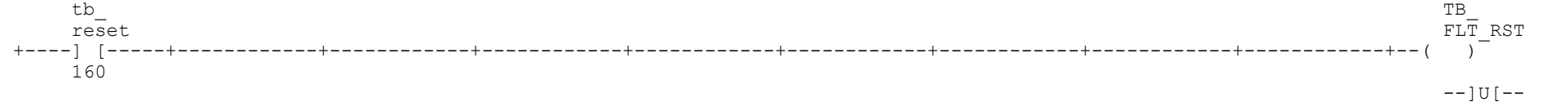
Sequence 150 THR BUGGY REMOTE RESET



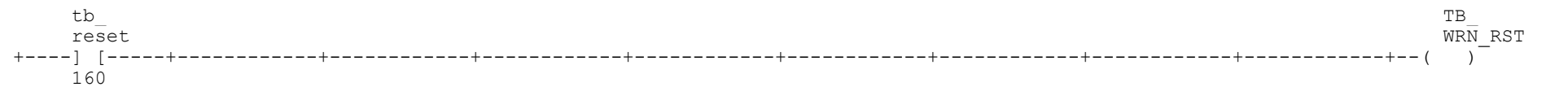
Sequence 160 THR BUGGY RESET



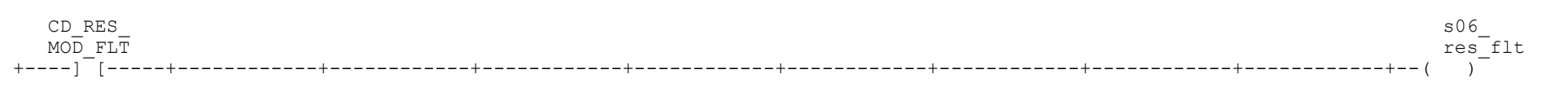
Sequence 170 THR BUGGY UDC FAULT RESET



Sequence 180 THR BUGGY UDC WARNING RESET



Sequence 600 SLOT 06 RESOLVER CARD FAULT

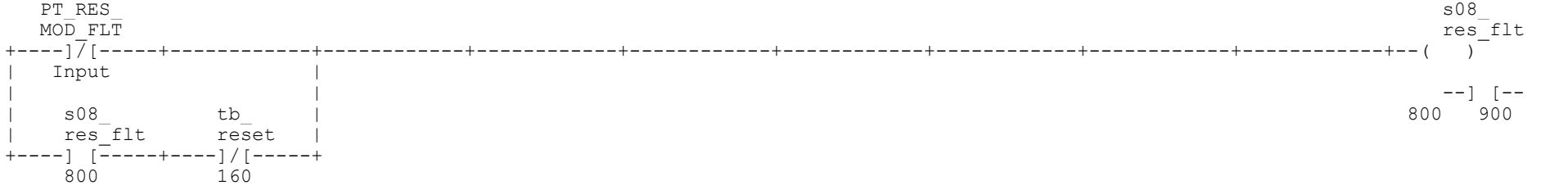


Sequence 650

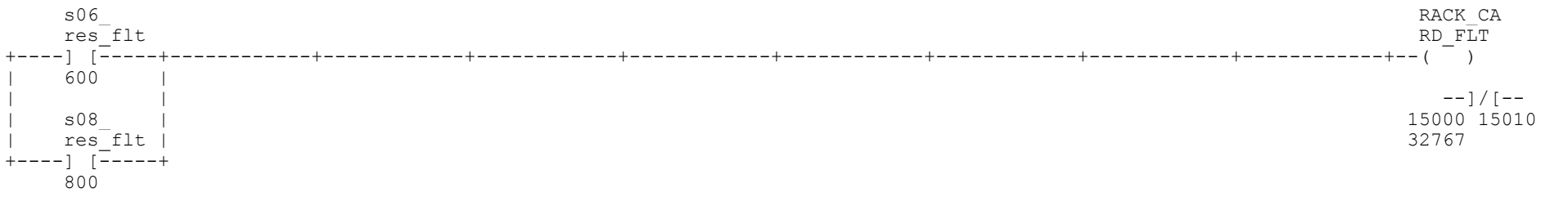
REMARKS Keyname: RES_FAULT

!RES_FAULT
PIERCER ROLL ADJUSTER RESOLVER NOT USED
SEQUENCE NUMBER 700
PIERCER TOP SHOE ADJUSTER RESOLVER NOT USED
SEQUENCE NUMBER 800
CONTACTS FOR PR_RES_MOD_FLT@ AND PT_RES_MOD_FLT@
WILL HAVE TO BE CHANGED TO NORMALLY OPEN CONTATS
IF THE RESOLVERS ARE WIRE.

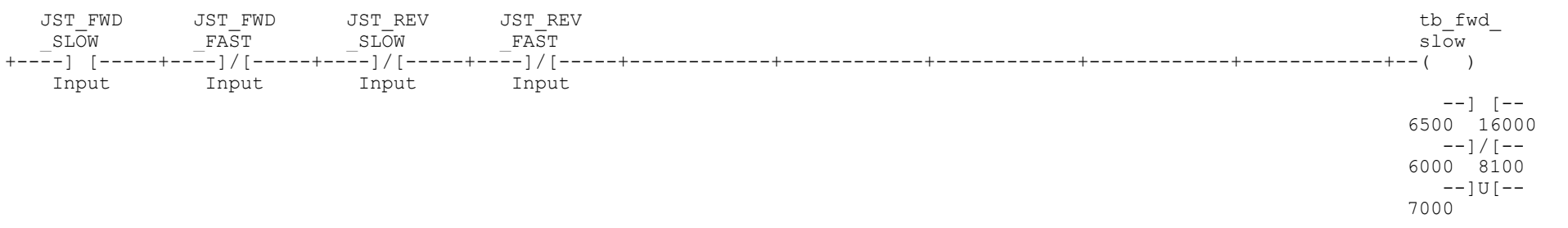
Sequence 800 SLOT 08 RESOLVER CARD FAULT



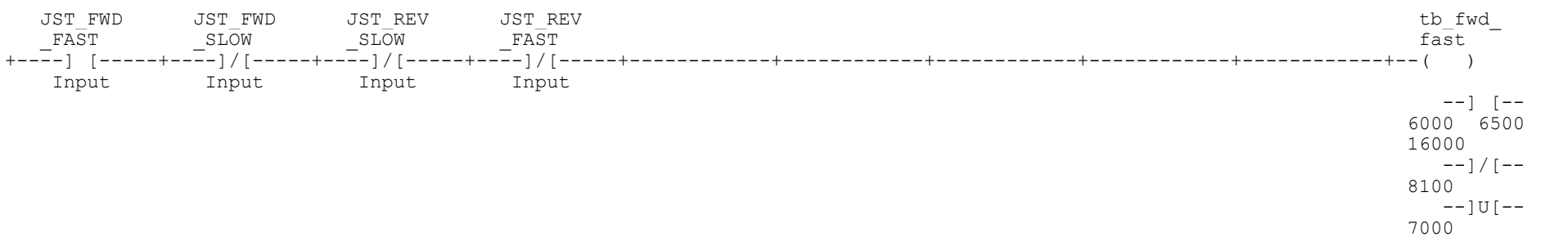
Sequence 900 AMX RACK CLOCKED CARDS OK



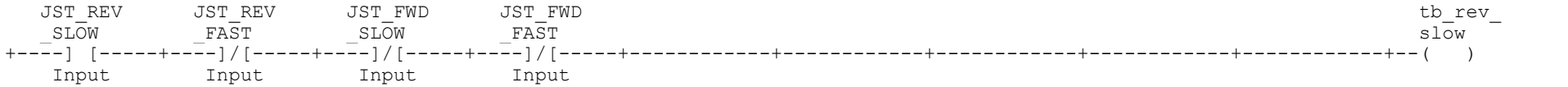
Sequence 5000 JOY STICK FORWARD SLOW

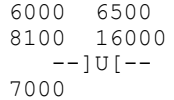


Sequence 5010 JOY STICK FORWARD FAST



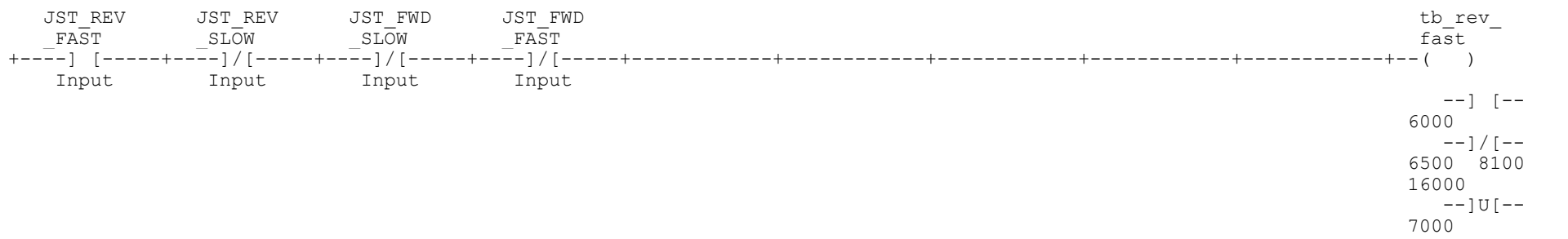
Sequence 5020 JOY STICK REVERSE SLOW



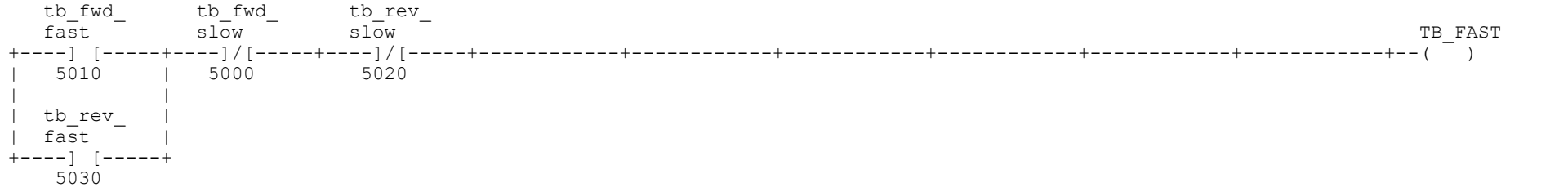

6000 6500
8100 16000
--]U[--
7000

Task Source Code Last Edit Date:19-Nov-10 09:00:50

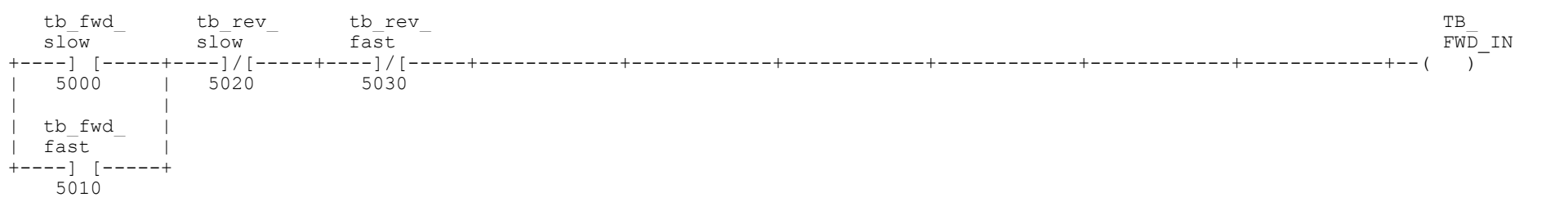
Sequence 5030 JOY STICK REVERSE FAST



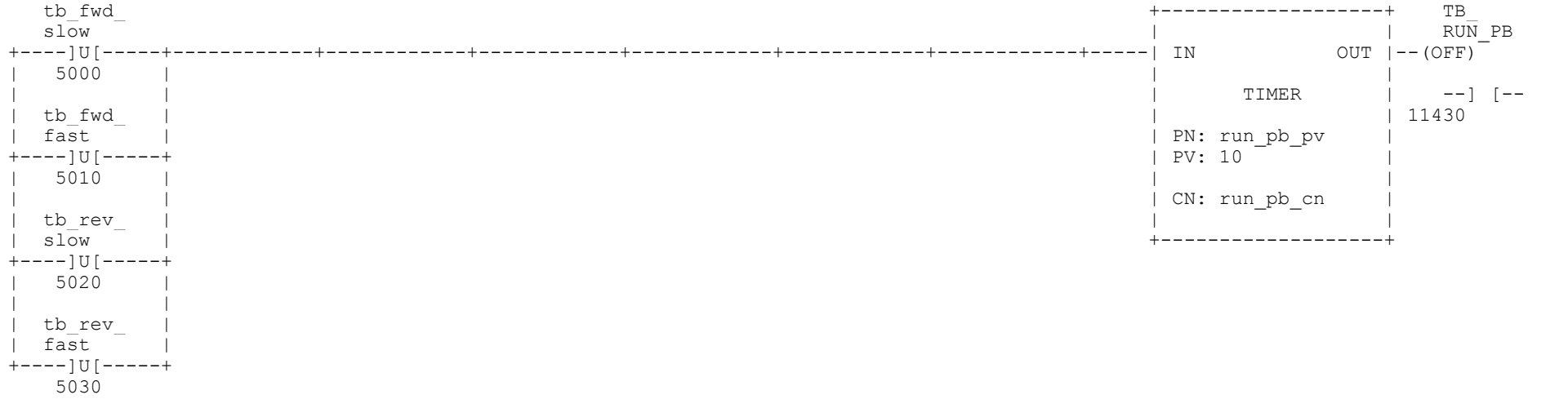
Sequence 6000 THR BUGGY FAST REF ENABLE



Sequence 6500 THR BUGGY FWD IN REFERENCE ENABLE



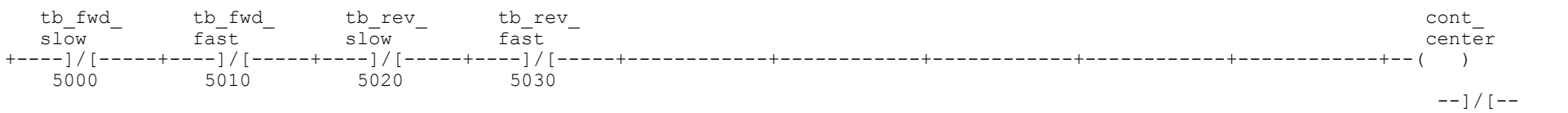
Sequence 7000 THR BUGGY RUN P/B



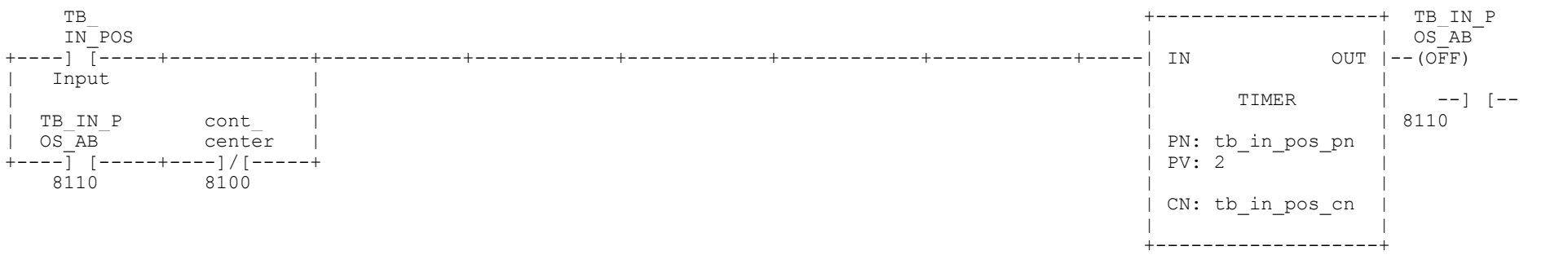
```
     TB_                                                                                                                             tb_
     IN_POS                                                                                                                          stop_pb
+----]/[-----+------------+------------+------------+------------+------------+------------+------------+------------+--(   )
    Input
                                                                                                                                     --] [--
                                                                                                                                  11430
```

Task Source Code Last Edit Date:19-Nov-10 09:00:50

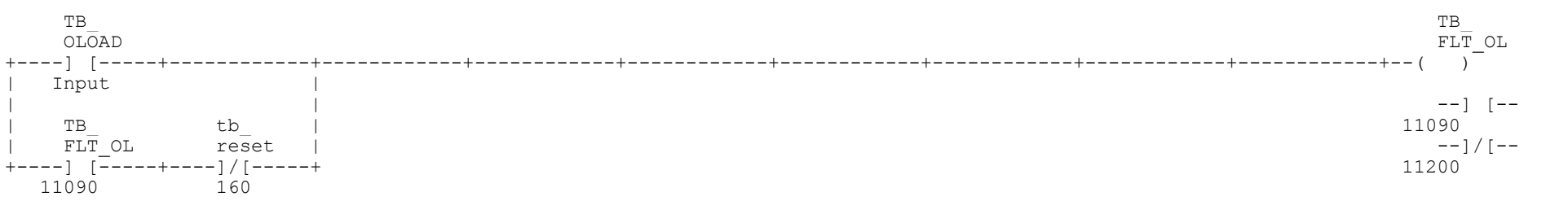
Sequence 8100 THR BUGGY STOP CONTROLER CENTERED



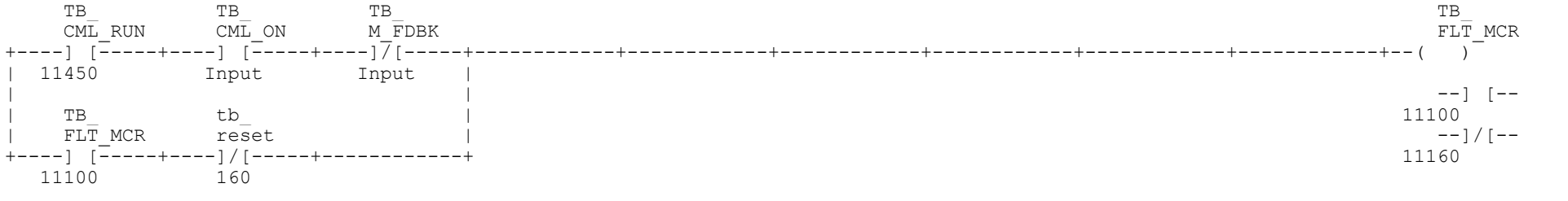
Sequence 8110 THRUST BUGGY IN POSITION LATCH



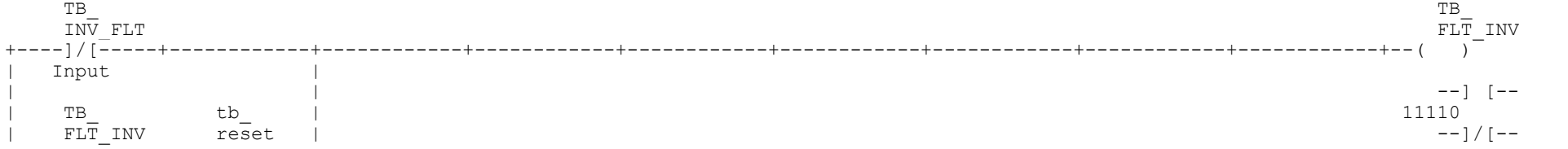
Sequence 11090 THR BUGGY OVERLOAD FAULT

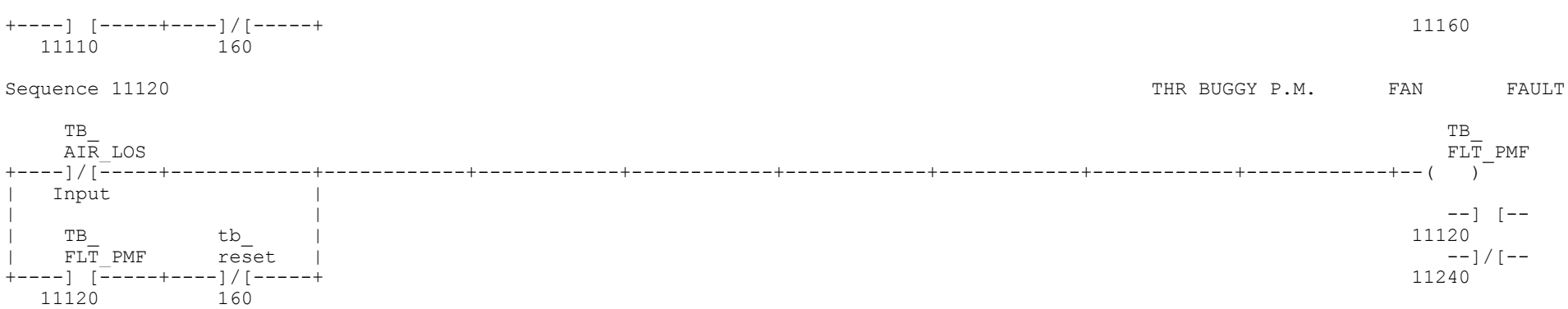


Sequence 11100 THR BUGGY MAIN CONTACTOR FAULT



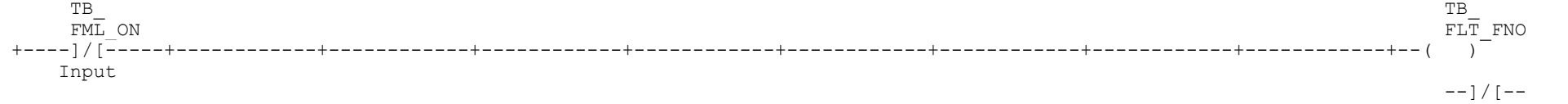
Sequence 11110 THR BUGGY INVERTING FAULT



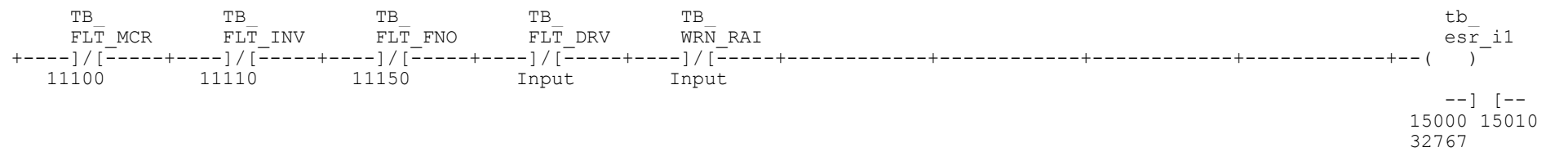

11110
160
11160
Sequence 11120
THR BUGGY P.M.
FAN
FAULT
TB_
AIR_LOS
TB_
FLT_PMF
Input
TB_
FLT_PMF
tb_
reset
11120
160
11120
11240

Task Source Code Last Edit Date:19-Nov-10 09:00:50

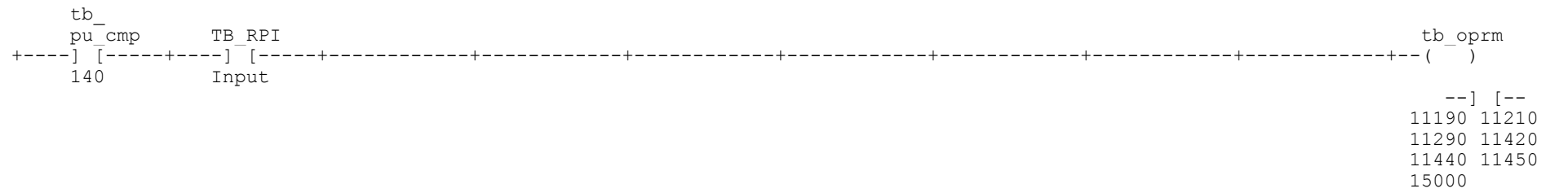
Sequence 11150 THR BUGGY FIELD NOT ON FAULT



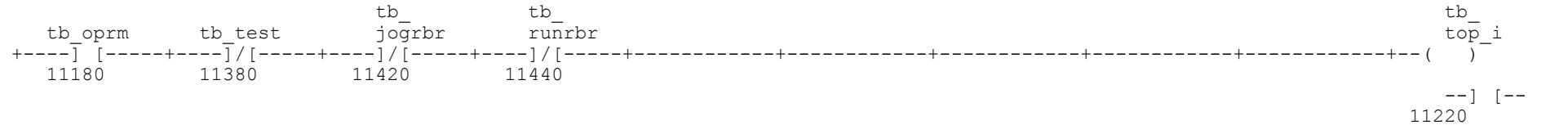
Sequence 11160 THR BUGGY EMERGENCY STOP INTRLCK #1



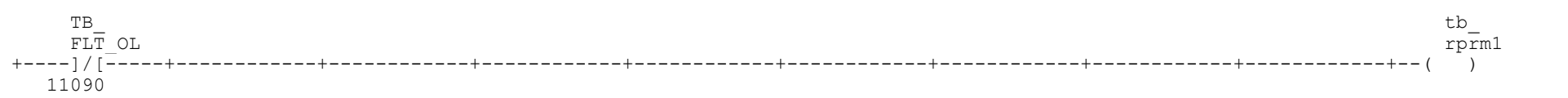
Sequence 11180 THR BUGGY ON PERMISSIVE



Sequence 11190 THR BUGGY TANDEM ON PERMISSIVEINTERLOCK



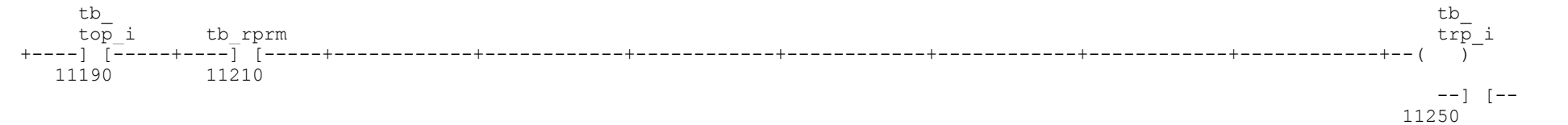
Sequence 11200 THR BUGGY RUN PERMISSIVE#1



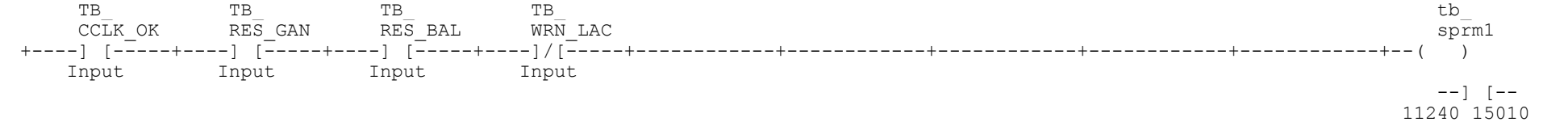
Sequence 11210 THR BUGGY RUN PERMISSIVE

```
   tb_oprm        rprm1                                                                                                                  tb_rprm
+----] [-----+----] [-----+------------+------------+------------+------------+------------+------------+------------+--(   )
   11180        11200
                                                                                                                             --] [--
                                                                                                                          11220 11240
                                                                                                                          11290 11430
                                                                                                                          15000
```

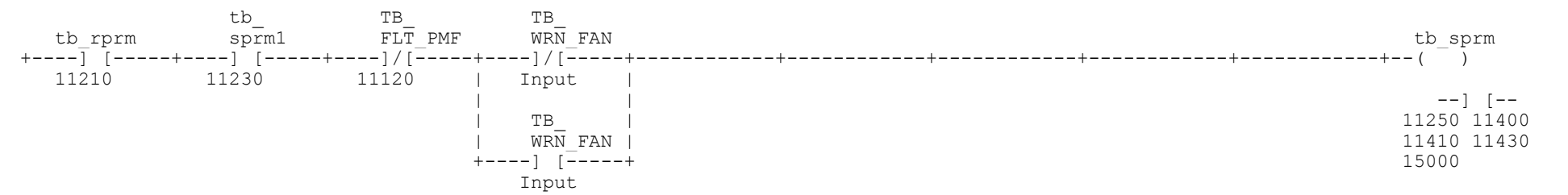
Sequence 11220 THR BUGGY TANDEM RUNPERMISSIVEINTERLOCK



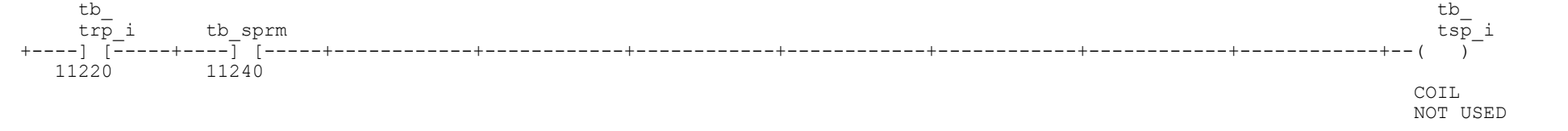
Sequence 11230 THR BUGGY START PERMISSIVE#1



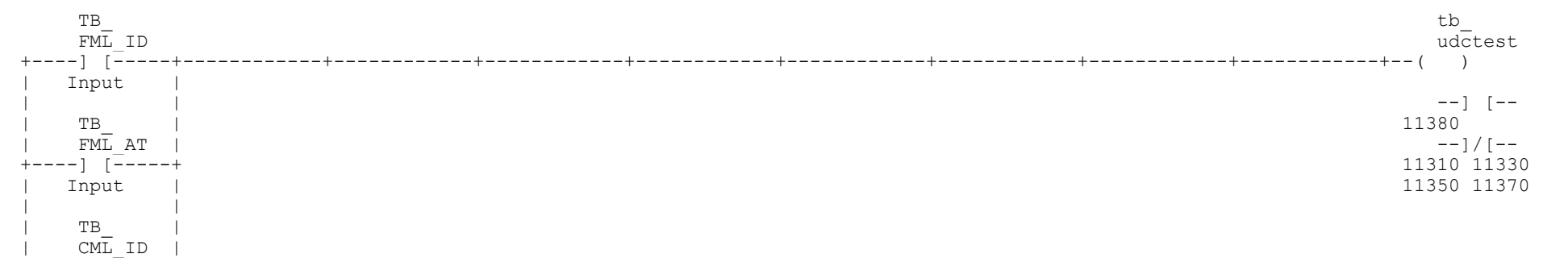
Sequence 11240 THR BUGGY START PERMISSIVE



Sequence 11250 THR BUGGY TANDEM ST PERMISSIVEINTERLOCK



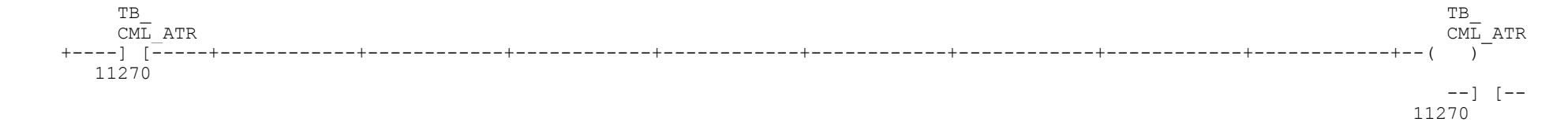
Sequence 11260 THR BUGGY UDC TEST



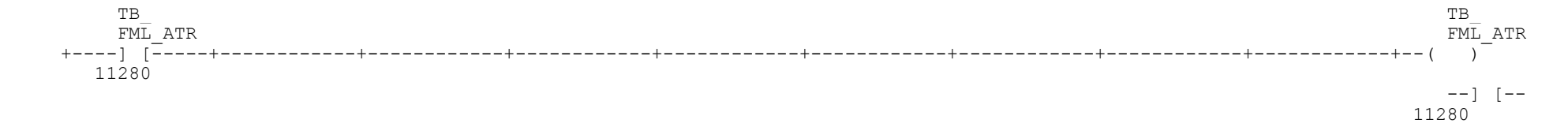
```
+----] [-----+
|   Input    |
|            |
|    TB_     |
|    CML_AT  |
+----] [-----+
|   Input    |
|            |
|    TB_     |
|    UDC_LB  |
+----] [-----+
    Input
```

Task Source Code Last Edit Date:19-Nov-10 09:00:50

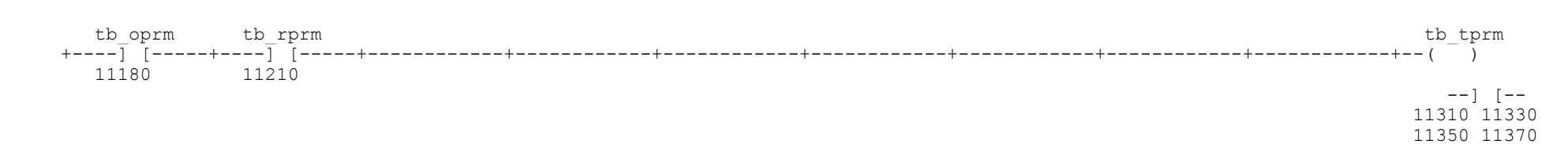
Sequence 11270 THR BUGGY CML ALPHA TESTPOLARITY



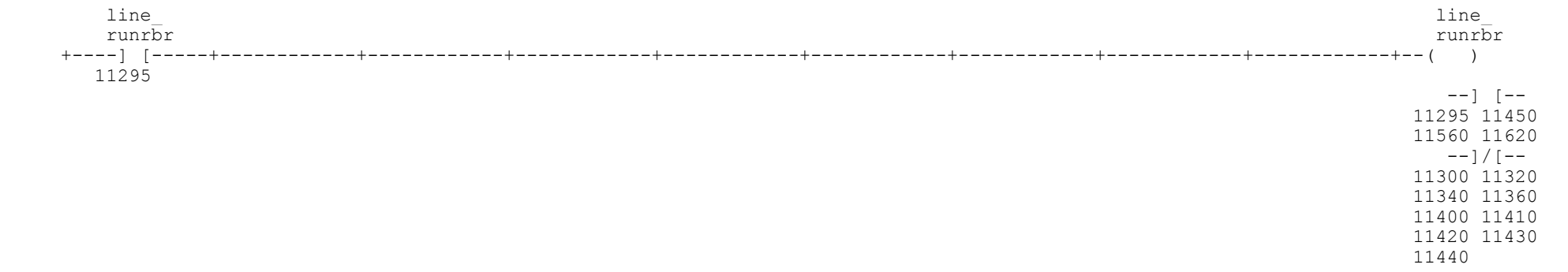
Sequence 11280 THR BUGGY FIELD ALPHA TESTPOLARITY



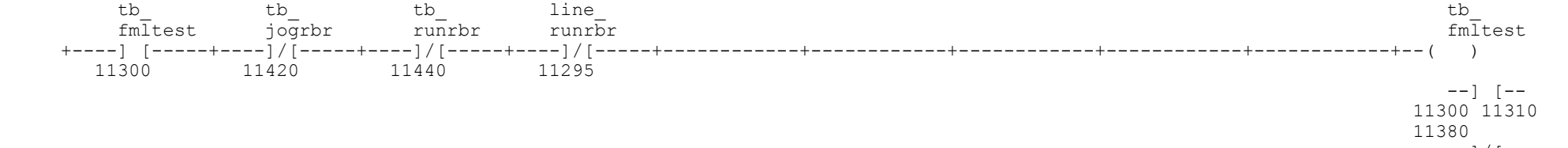
Sequence 11290 THR BUGGY TEST PERMISSIVE

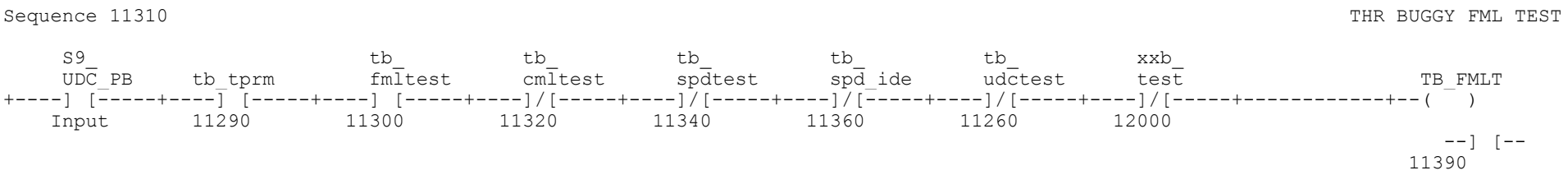


Sequence 11295 LINE RUN REGEN BRAKING

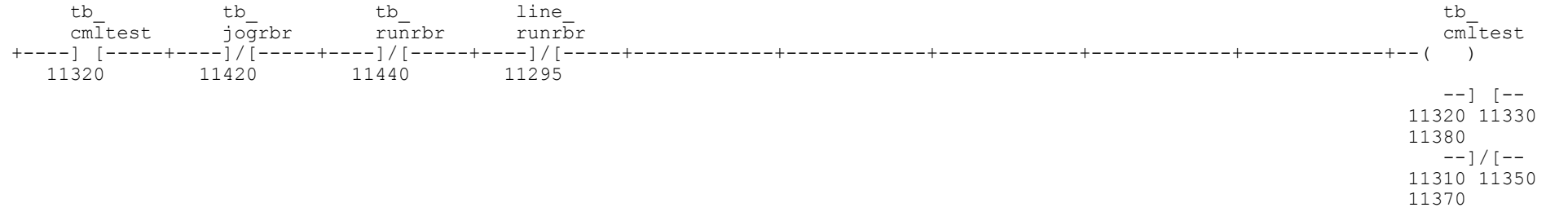


Sequence 11300 THR BUGGY FML TEST ENABLE

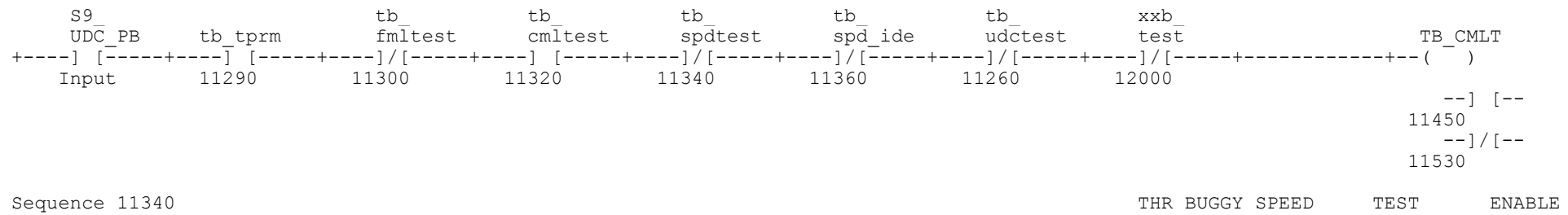



11330 11350
11370
Sequence 11310
THR BUGGY FML TEST
S9_
UDC_PB
tb_tprm
tb_
fmltest
tb_
cmltest
tb_
spdtest
tb_
spd_ide
tb_
udctest
xxb_
test
TB_FMLT
Input
11290
11300
11320
11340
11360
11260
12000
11390

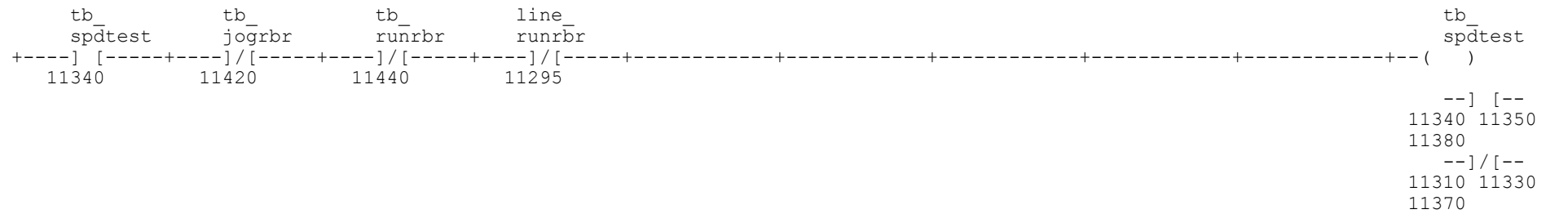
Sequence 11320 THR BUGGY CML TEST ENABLE



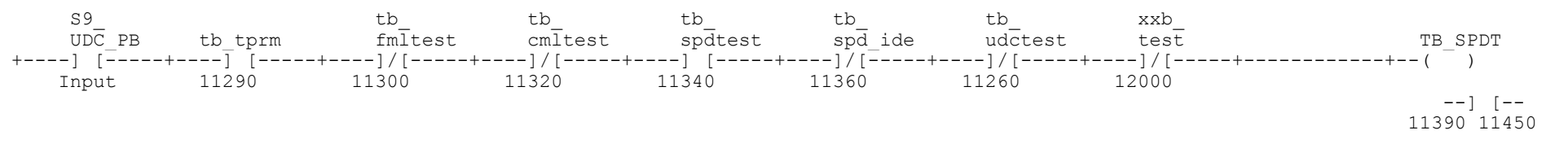
Sequence 11330 THR BUGGY CML TEST



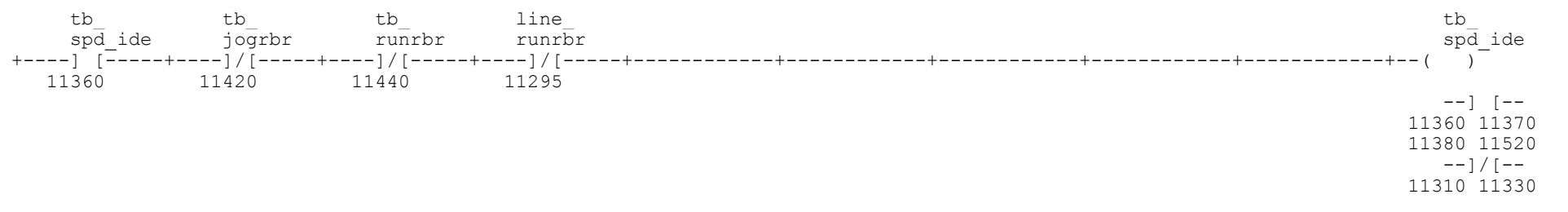
Sequence 11340 THR BUGGY SPEED TEST ENABLE

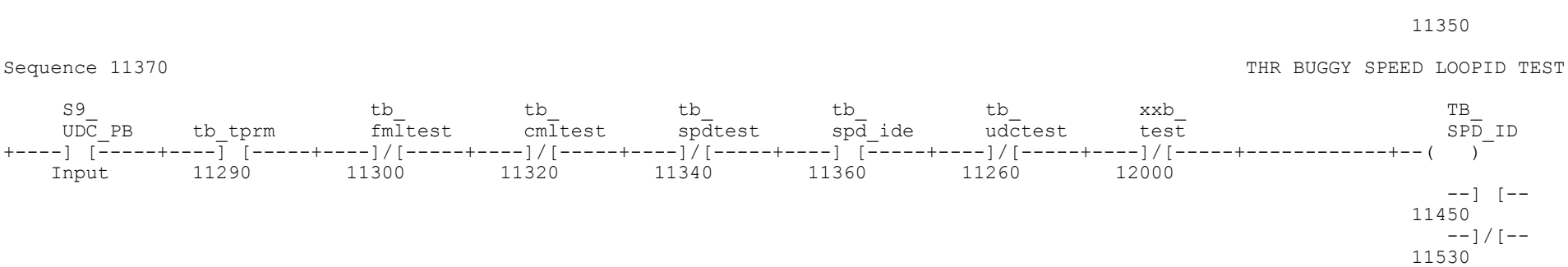


Sequence 11350 THR BUGGY SPEED TEST



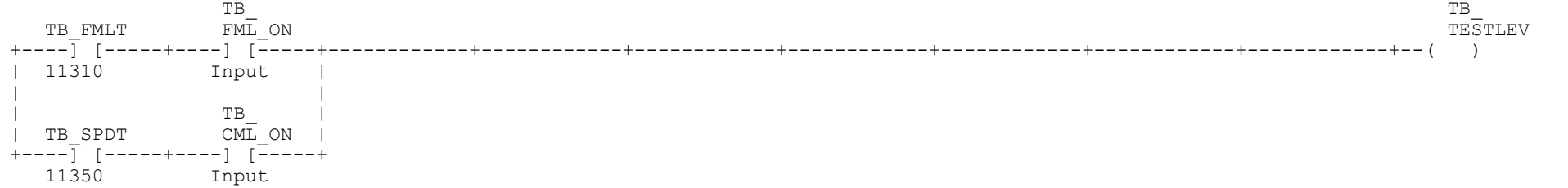
Sequence 11360 THR BUGGY SPEED LOOPID TEST ENABLE



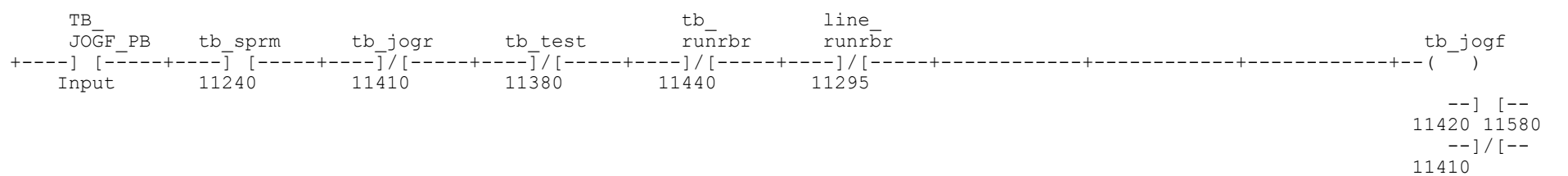

11350
Sequence 11370
THR BUGGY SPEED LOOPID TEST
S9_
UDC_PB
tb_tprm
tb_
fmltest
tb_
cmltest
tb_
spdtest
tb_
spd_ide
tb_
udctest
xxb_
test
TB_
SPD_ID
Input
11290
11300
11320
11340
11360
11260
12000
11450
11530

Sequence 11380 THR BUGGY TEST

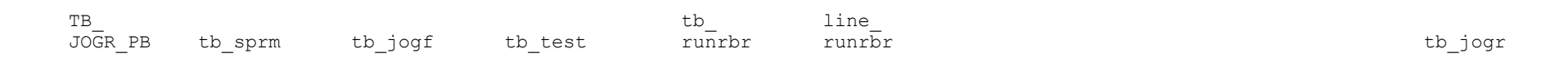
Sequence 11390 THR BUGGY TEST LEVELENABLE



Sequence 11400 THR BUGGY JOG FORWARD



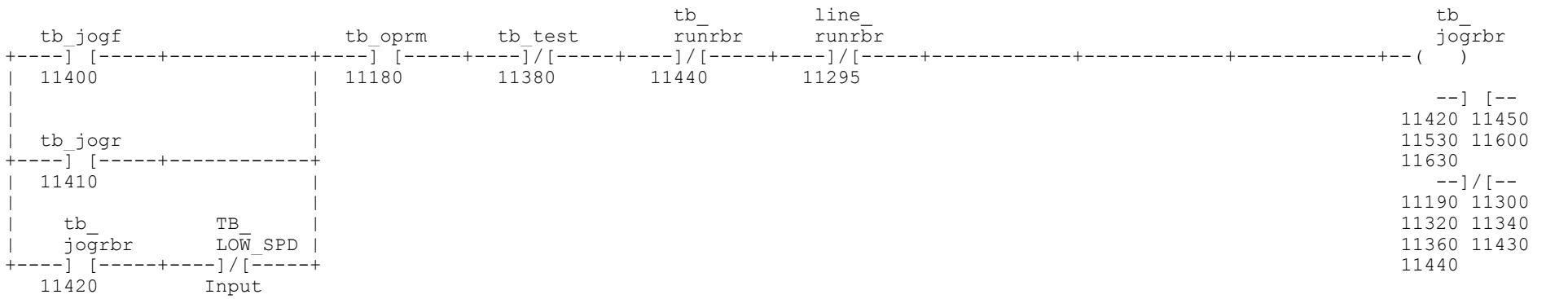
Sequence 11410 THR BUGGY JOG REVERSE



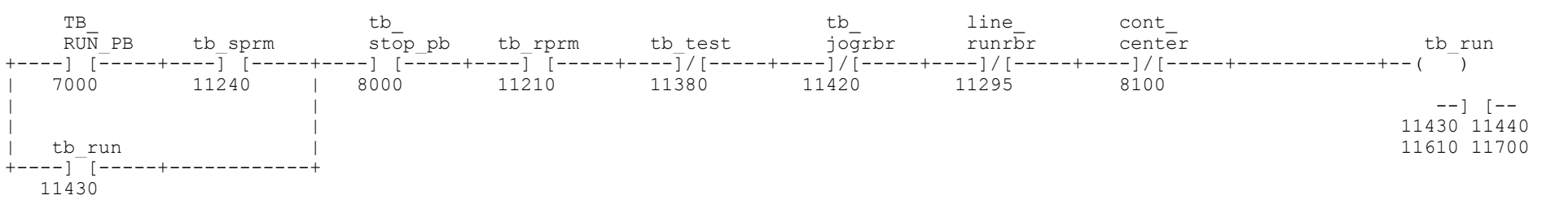
```
+----] [-----+----] [-----+----]/[-----+----]/[-----+----]/[-----+----]/[-----+-----------+-----------+-----------+--(   )
    Input       11240        11400        11380        11440        11295
                                                                                                                            --] [--
                                                                                                                         11420 11590
                                                                                                                            --]/[--
                                                                                                                         11400
```

Task Source Code Last Edit Date:19-Nov-10 09:00:50

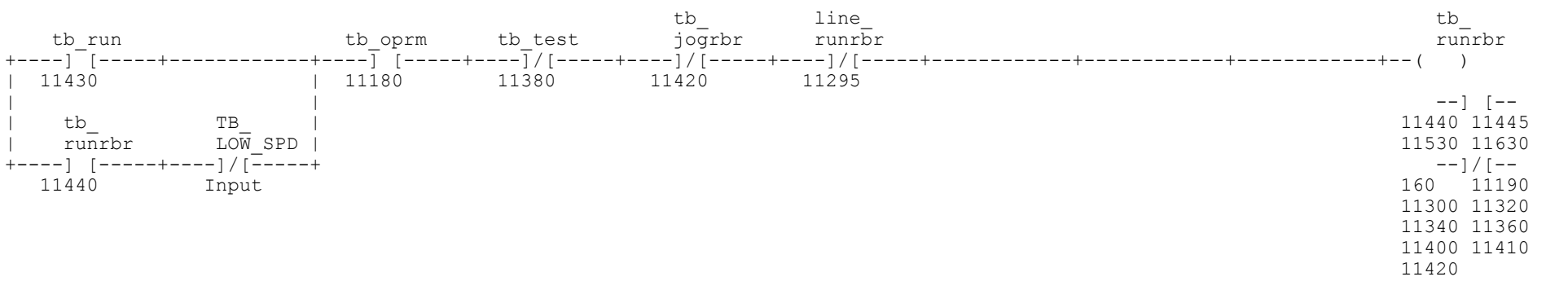
Sequence 11420 THR BUGGY JOG REGEN BRAKING



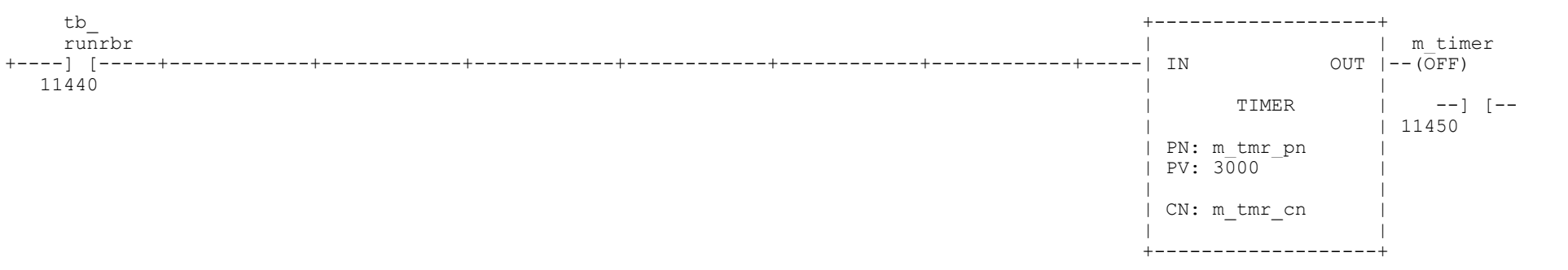
Sequence 11430 THR BUGGY RUN



Sequence 11440 THR BUGGY RUN REGEN BRAKING



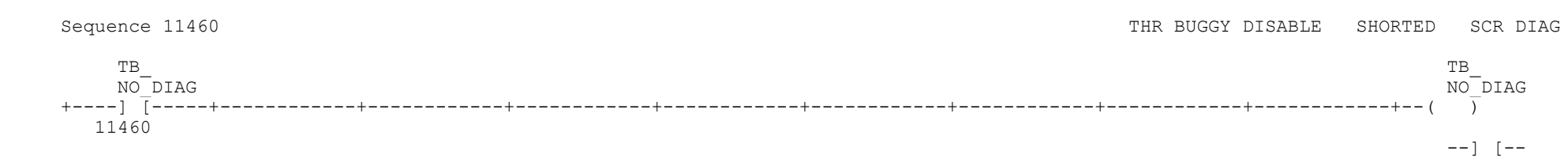
Sequence 11445 THR BUGGY RUN MAIN CONTACTOR OFF TIMER



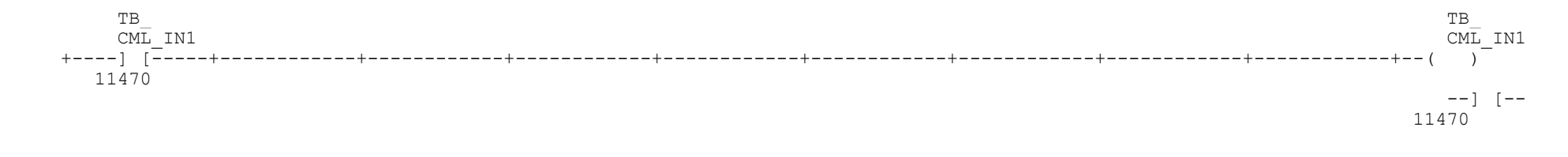
Task Source Code Last Edit Date:19-Nov-10 09:00:50

Sequence 11450 THR BUGGY CURRENT MINOR LOOPRUN

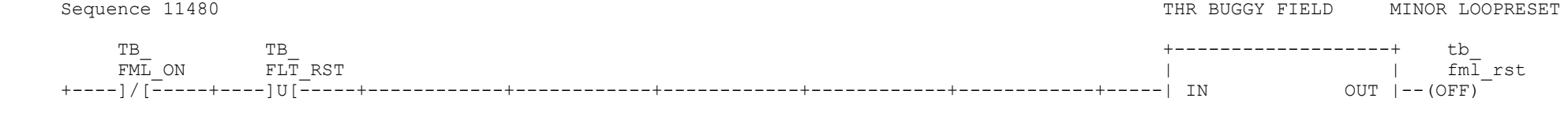
Sequence 11460 THR BUGGY DISABLE SHORTED SCR DIAG



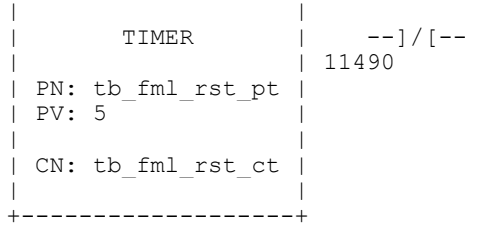
Sequence 11470 THR BUGGY CML INPUT #1 SELECT



Sequence 11480 THR BUGGY FIELD MINOR LOOPRESET

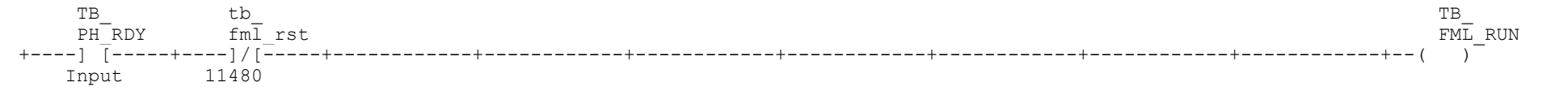


Input 170

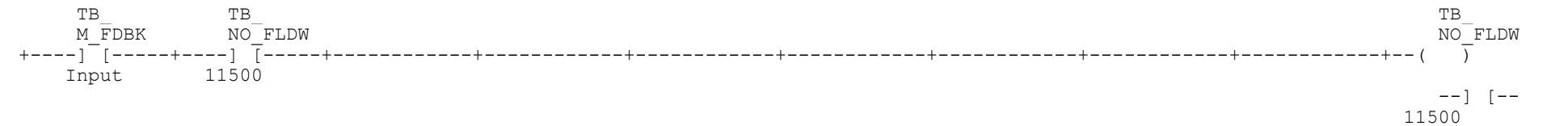


Task Source Code Last Edit Date:19-Nov-10 09:00:50

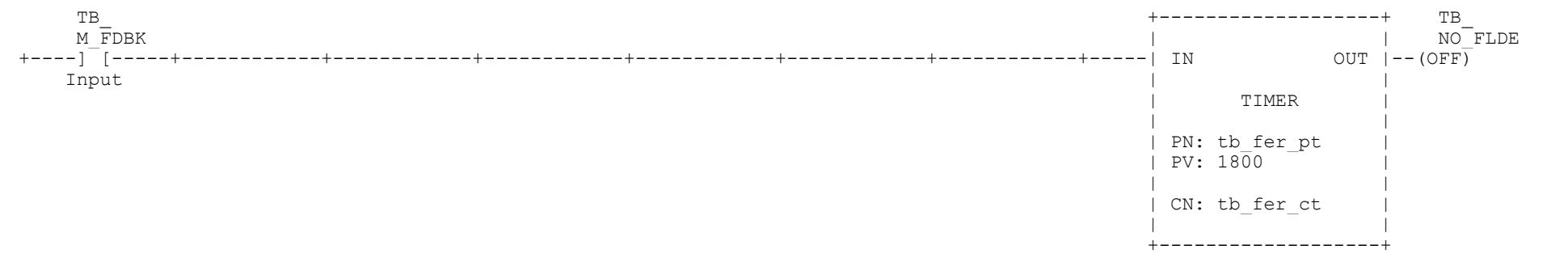
Sequence 11490 THR BUGGY FIELD MINOR LOOPRUN



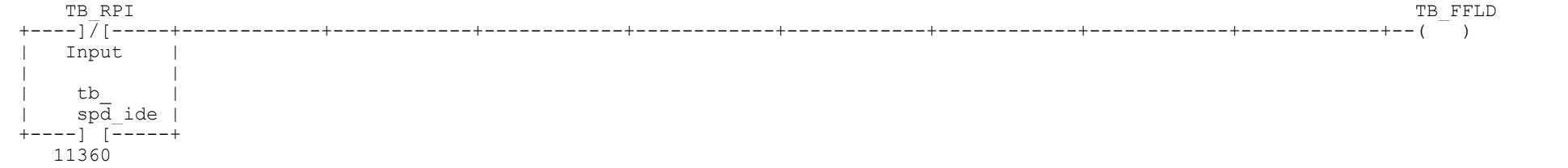
Sequence 11500 THR BUGGY DISABLE FIELD WEAKENING



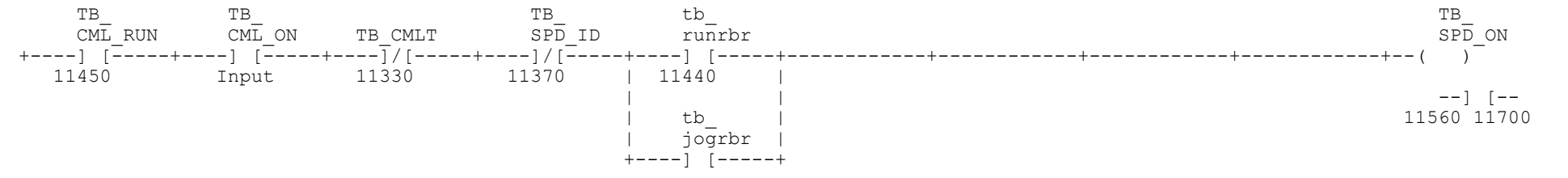
Sequence 11510 THR BUGGY DISABLE FIELD ECONOMY



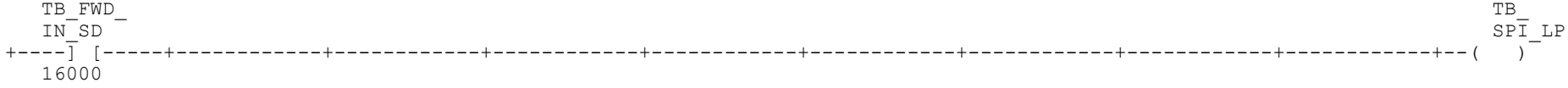
Sequence 11520 THR BUGGY FULL FIELD



Sequence 11530 THR BUGGY SPEED LOOPON

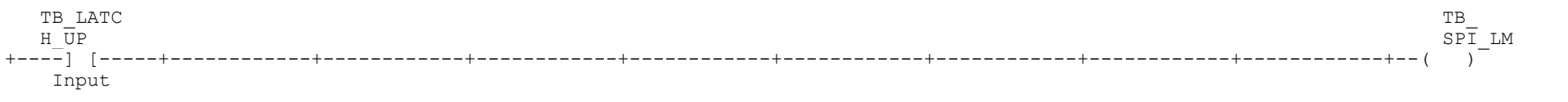


Sequence 11540 THR BUGGY SPEED PI LIMIT + SELECT

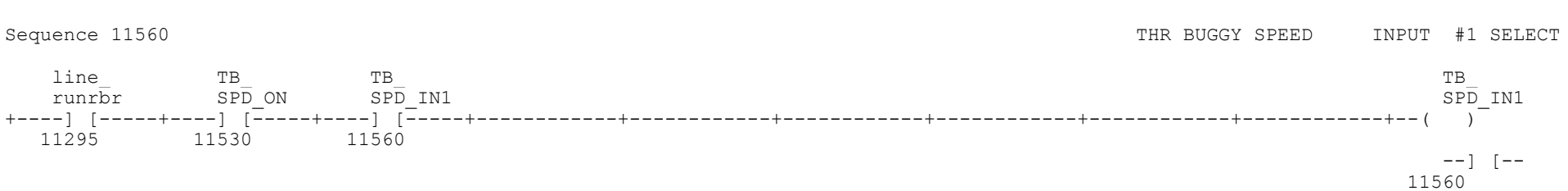


Task Source Code Last Edit Date:19-Nov-10 09:00:50

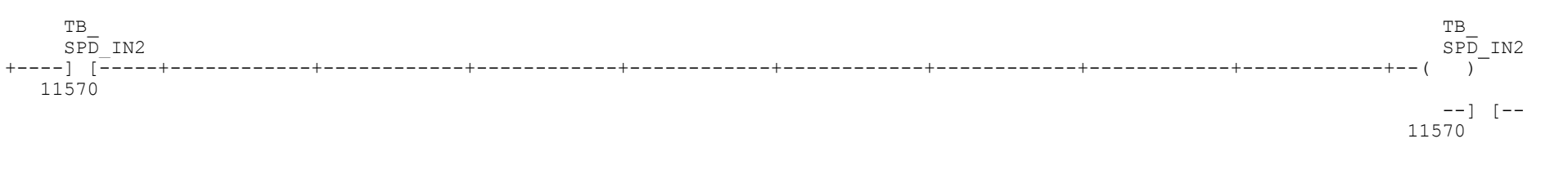
Sequence 11550 THR BUGGY SPEED PI LIMIT - SELECT



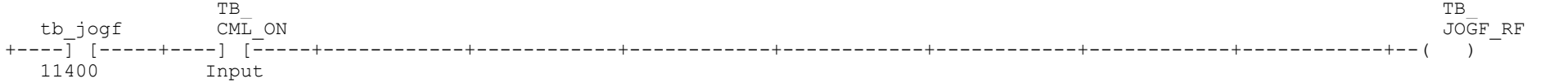
Sequence 11560 THR BUGGY SPEED INPUT #1 SELECT



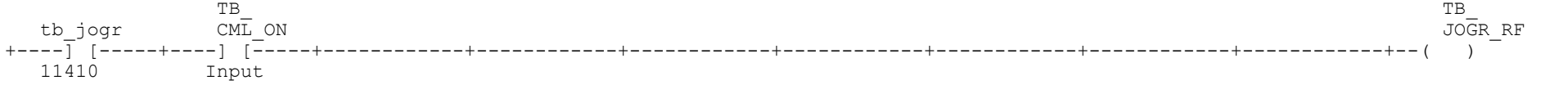
Sequence 11570 THR BUGGY SPEED INPUT #2 SELECT



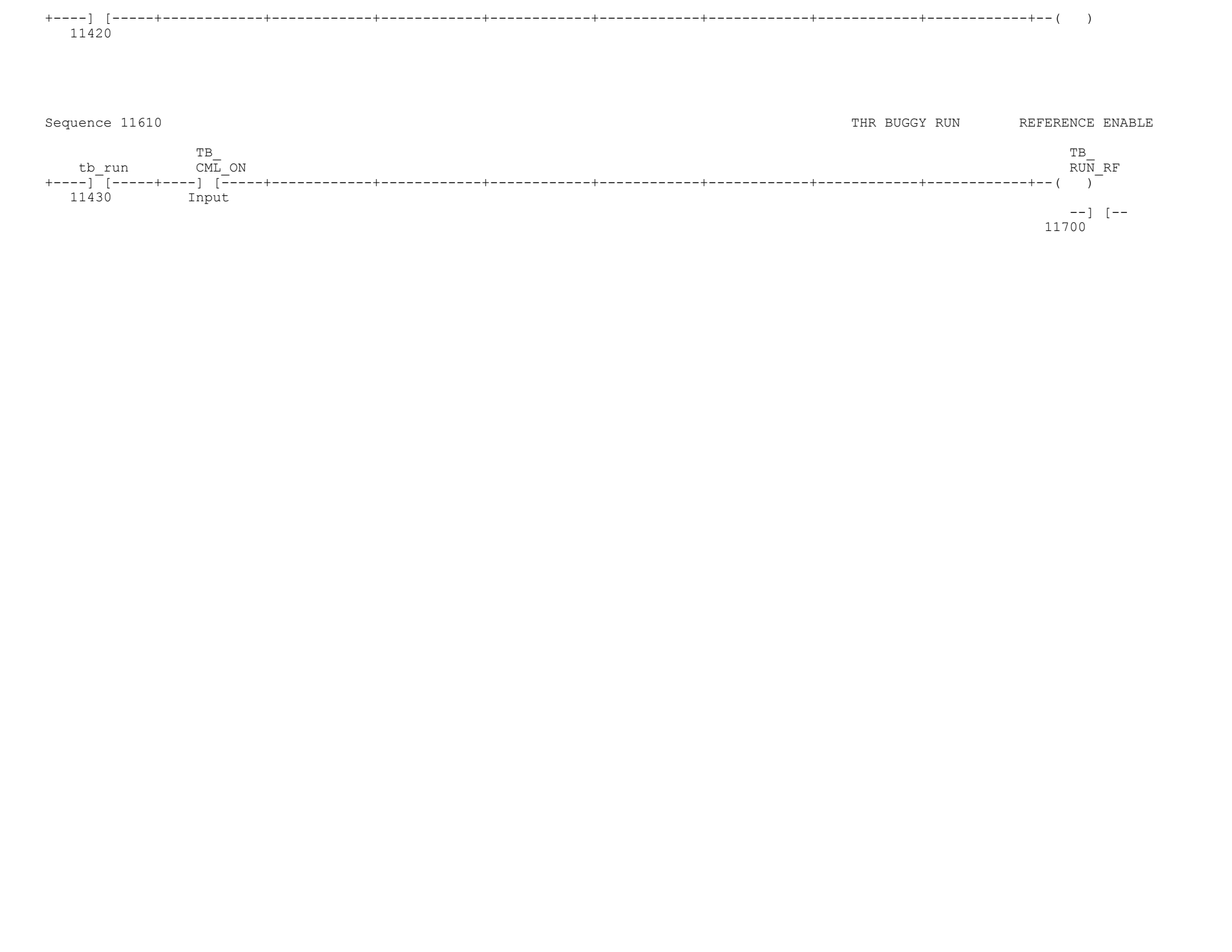
Sequence 11580 THR BUGGY JOG FORWD REFERENCE ENABLE



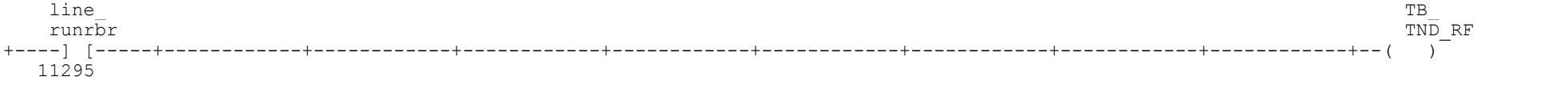
Sequence 11590 THR BUGGY JOG REV REFERENCE ENABLE



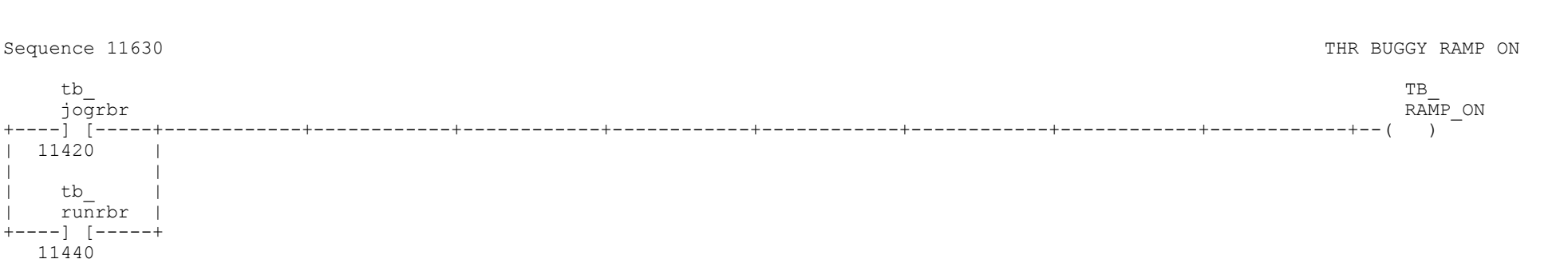
Sequence 11600 THR BUGGY JOG RATE ENABLE


11420
Sequence 11610
THR BUGGY RUN
REFERENCE ENABLE
tb_run
TB_
CML_ON
11430
Input
TB_
RUN_RF
11700

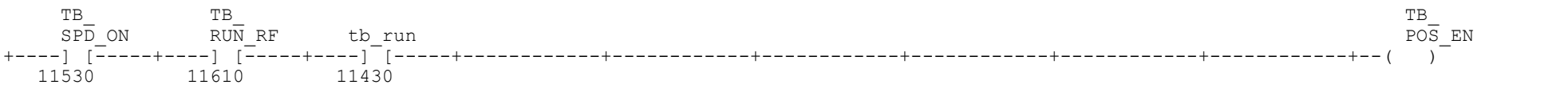
Sequence 11620 THR BUGGY TANDEM REFERENCE ENABLE



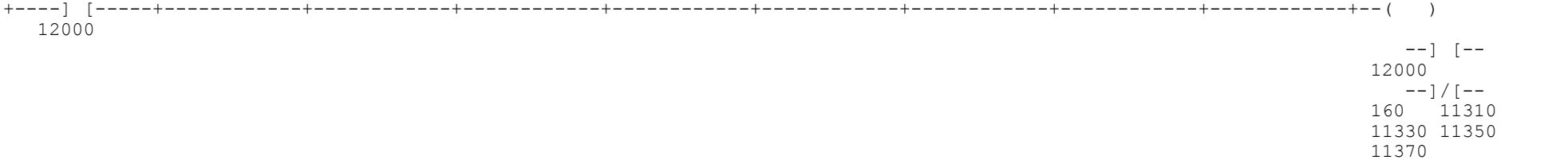
Sequence 11630 THR BUGGY RAMP ON



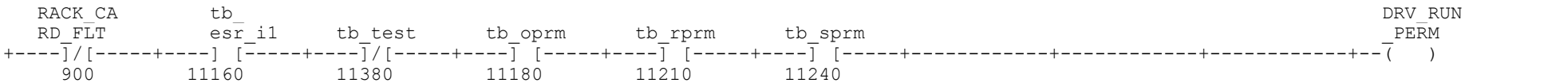
Sequence 11700 THR BUGGY POSITION LOOP ON



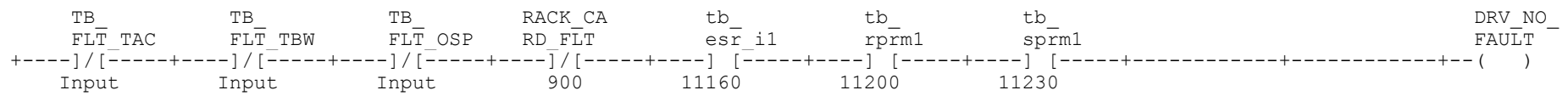
Sequence 12000 XXXXXXXXXBDRIVE TEST



Sequence 15000 THR BUGGY DRIVE RUN PERMISSIVE

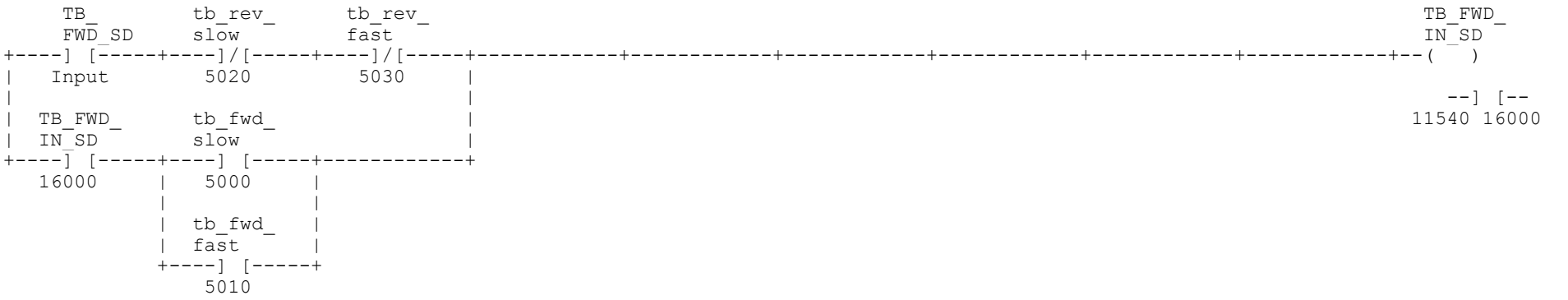


Sequence 15010 THR BUGGY DRIVE NO FAULTS

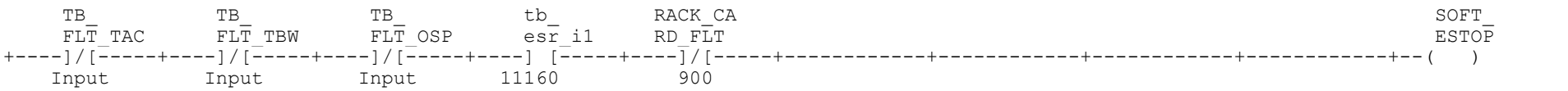


Task Source Code Last Edit Date:19-Nov-10 09:00:50

Sequence 16000 THRUST BUGGY FWD REF SD ENABLE



Sequence 32767 SOFTWARE E-STOP RELAY



```
1 !*
2 !* System:  FM570572
3 !* Rack:  A30
4 !* Task:  S9SPD.BLK
5 !*
7 !* Copyright Reliance Electric Industrial Company 1995
8 !* Extracted from AESOP Version 2.42b on 01-MAY-95
9 !*****************************************************************
999 !*****************************************************************&
  **                                                                 **&
  **    C o m m o n   V a r i a b l e s     **&
  **                                                                 **&
  *******************************************************************
1000  COMMON  S9_SWIT_DN@
1002  COMMON  S9_SWIT_UP@
1004  COMMON  TB_CMLT@
1006  COMMON  TB_CML_ALPHA%
1008  COMMON  TB_CML_FB%
1010  COMMON  TB_CML_IN1@
1012  COMMON  TB_CML_REF%
1014  COMMON  TB_CRF_SM@
1016  COMMON  TB_CRF_SP@
1018  COMMON  TB_FMLT@
1020  COMMON  TB_FML_ALPHA%
1022  COMMON  TB_FML_REF%
1024  COMMON  TB_JOGF_RF@
1026  COMMON  TB_JOGR_RF@
1028  COMMON  TB_JOG_RT@
1030  COMMON  TB_LOW_SPD@
1032  COMMON  TB_M_FDBK@
1034  COMMON  TB_NO_FLDE@
1036  COMMON  TB_OLOAD@
1038  COMMON  TB_RAMP_ON@
1040  COMMON  TB_RES_SCN_POS%
1042  COMMON  TB_RUN_REF%
1044  COMMON  TB_RUN_RF@
1046  COMMON  TB_SPDT@
1048  COMMON  TB_SPD_FB%
1050  COMMON  TB_SPD_ID@
1052  COMMON  TB_SPD_IN1@
1054  COMMON  TB_SPD_IN2@
1056  COMMON  TB_SPD_ON@
1060  COMMON  TB_SPD_REF%
1062  COMMON  TB_SPI_LM@
1064  COMMON  TB_SPI_LP@
1066  COMMON  TB_SPI_OUT%
1068  COMMON  TB_SPI_SM@
1070  COMMON  TB_SPI_SP@
1072  COMMON  TB_SRF_SM@
1074  COMMON  TB_SRF_SP@
1078  COMMON  TB_STEP_FB%
1080  COMMON  TB_TESTLEV@
1082  COMMON  TB_TND_RF@
1084  COMMON  TB_ZER_SPD@
2000  COMMON  TB_POS_RES_UDC% \! THR BUGGY POSITION FDBK (CHART RECORDER)
2999  !*****************************************************************&
  **                                                                 **&
  **    L o c a l   V a r i a b l e s     **&
  **                                                                 **&
  *******************************************************************
3000  LOCAL ACC_RATE%
```

```
3002  LOCAL ACC_WR2%
3004  LOCAL ASPD_FB%
3006  LOCAL DEC_RATE%
3008  LOCAL DEC_WR2_int%
3010  LOCAL DEC_WR2@
3012  LOCAL FLLS_1_ON@
3014  LOCAL FML_LP%
3016  LOCAL FML_REF%
3018  LOCAL JERK%
3020  LOCAL LLS_1_ON@
3022  LOCAL LLS_2_ON@
3024  LOCAL NEG_DEC@
3026  LOCAL POS_DEC@
3028  LOCAL POS_REF_RATE@
3030  LOCAL POS_SPD_REF@
3032  LOCAL REF_RATE_1%
3034  LOCAL REF_RATE_2%
3036  LOCAL REF_RATE_3%
3038  LOCAL SB_RPM%
3040  LOCAL SGI_RPM%
3042  LOCAL SID_ACC_TIME!
3044  LOCAL SID_SCAN_THR%
3046  LOCAL SID_SPD_1%
3048  LOCAL SID_SPD_LIM%
3050  LOCAL SID_SPD_THR%
3052  LOCAL SID_SP_TIME%
3054  LOCAL SID_STEP%
3056  LOCAL SID_ST_SPD%
3058  LOCAL SID_ST_TRQ%
3060  LOCAL SID_TIME_1%
3062  LOCAL SID_TRQ_1%
3064  LOCAL SID_TRQ_REF
3066  LOCAL SID_TRQ_REF_int%
3068  LOCAL SID_WT_int%
3070  LOCAL SID_WT@
3072  LOCAL SPD_ERR%
3074  LOCAL SPD_FB_1%
3076  LOCAL SPD_FB_2%
3078  LOCAL SPD_ID@
3080  LOCAL SPD_OUT_1%
3082  LOCAL SPD_OUT_2%
3084  LOCAL SPD_OUT_3%
3086  LOCAL SPD_OUT_4%
3088  LOCAL SPD_REF_1%
3090  LOCAL SPD_REF_2%
3092  LOCAL SPD_REF_3%
3094  LOCAL SPD_REF_4%
3096  LOCAL SPD_REF_5%
3098  LOCAL SPI_LM%
3100  LOCAL SPI_LP%
3102  LOCAL TB_ASPD_REF%
3104  LOCAL TB_CML_IN1_int%
3106  LOCAL TB_REF_RATE%
3110  LOCAL TB_SPD_FBJ%
3112  LOCAL TB_SPD_FB_EU%
3116  LOCAL TB_SPD_RFJ%
3120  LOCAL TB_TND_RATE%
3122  LOCAL TB_TND_REF%
3124  LOCAL TB_WR2_COMP%
3126  LOCAL TEST_LEVEL_1%
3128  LOCAL TEST_RATE%
3130  LOCAL TEST_STEP%
```

```
3132  LOCAL WR2%
3134  LOCAL WR2_1%
3136  LOCAL WR2_RATE%
4999  !*****************************************************************&
  **                                                                   **&
  **  L o c a l   T u n a b l e   V a r i a b l e s   **&
  **                                                                   **&
  *********************************************************************
5000  LOCAL ACC_WR2_GAIN%[ CURRENT = 1000, HIGH = 10000, LOW = 100, &
     STEP = 1 ]
5002  LOCAL ARM_CCT%[ CURRENT = 593, HIGH = 32767, LOW = 0, STEP = 1 ]
5004  LOCAL ARM_R%[ CURRENT = 78, HIGH = 32767, LOW = 0, STEP = 1 ]
5006  LOCAL ARM_TE%[ CURRENT = 321, HIGH = 5000, LOW = 0, STEP = 1 ]
5008  LOCAL BASE_SPEED%[ CURRENT = 4095, HIGH = 32767, LOW = 0, STEP = 1 ]
5010  LOCAL CML_ALPHA%[ CURRENT = 180, HIGH = 180, LOW = 0, STEP = 10 ]
5012  LOCAL CML_WCO%[ CURRENT = 200, HIGH = 400, LOW = 0, STEP = 1 ]
5014  LOCAL CUR_LM%[ CURRENT = -4095, HIGH = 0, LOW = -5000, STEP = 1 ]
5016  LOCAL CUR_LP%[ CURRENT = 4095, HIGH = 5000, LOW = 0, STEP = 1 ]
5018  LOCAL DEC_WR2_GAIN%[ CURRENT = 1000, HIGH = 10000, LOW = 100, &
     STEP = 1 ]
5020  LOCAL FLD_R%[ CURRENT = 551, HIGH = 32767, LOW = 0, STEP = 1 ]
5022  LOCAL FLD_TE%[ CURRENT = 866, HIGH = 32767, LOW = 0, STEP = 1 ]
5024  LOCAL FML_ALPHA%[ CURRENT = 180, HIGH = 180, LOW = 0, STEP = 10 ]
5026  LOCAL FML_ECON_REF%[ CURRENT = 2047, HIGH = 4095, LOW = 0, &
     STEP = 1 ]
5028  LOCAL FML_GEAR_IN_REF%[ CURRENT = 4095, HIGH = 4095, LOW = 0, &
     STEP = 1 ]
5030  LOCAL FML_WCO%[ CURRENT = 5, HIGH = 75, LOW = 0, STEP = 1 ]
5032  LOCAL GEAR_IN_EU%[ CURRENT = 4095, HIGH = 32767, LOW = 0, STEP = 1 ]
5034  LOCAL JBAR[ CURRENT = 0.0, HIGH = 240.0, LOW = 0.0, STEP = 0.01 ]
5036  LOCAL JOG_ACC_RATE%[ CURRENT = 100, HIGH = 32767, LOW = 1, &
     STEP = 1 ]
5038  LOCAL JOG_DEC_RATE%[ CURRENT = 100, HIGH = 32767, LOW = 1, &
     STEP = 1 ]
5040  LOCAL JOG_FWD_REF%[ CURRENT = 102, HIGH = 4095, LOW = 50, STEP = 1 ]
5042  LOCAL JOG_JERK%[ CURRENT = 10, HIGH = 32767, LOW = 1, STEP = 1 ]
5044  LOCAL JOG_REV_REF%[ CURRENT = -102, HIGH = -50, LOW = -4095, &
     STEP = 1 ]
5046  LOCAL KPS[ CURRENT = 25.0, HIGH = 120.0, LOW = 0.001, STEP = 0.001 ]
5048  LOCAL KPS_MAX[ CURRENT = 60.0, HIGH = 120.0, LOW = 20.0, &
     STEP = 1.0 ]
5050  LOCAL LIM_BAR%[ CURRENT = 150, HIGH = 400, LOW = 115, STEP = 1 ]
5052  LOCAL LOW_SPD_THR%[ CURRENT = 100, HIGH = 4095, LOW = 0, STEP = 1 ]
5054  LOCAL RES_BAL%[ CURRENT = 44, HIGH = 79, LOW = 0, STEP = 1 ]
5056  LOCAL RES_GAN%[ CURRENT = 120, HIGH = 255, LOW = 0, STEP = 1 ]
5058  LOCAL RES_TYPE%[ CURRENT = 2, HIGH = 5, LOW = 1, STEP = 1 ]
5060  LOCAL RUN_ACC_RATE%[ CURRENT = 5758, HIGH = 32767, LOW = 100, &
     STEP = 10 ]
5062  LOCAL RUN_DEC_RATE%[ CURRENT = 5758, HIGH = 32767, LOW = 100, &
     STEP = 10 ]
5064  LOCAL RUN_JERK%[ CURRENT = 144, HIGH = 3276, LOW = 10, STEP = 10 ]
5066  LOCAL SCAN_MS[ CURRENT = 5.5, HIGH = 10.0, LOW = 2.0, STEP = 0.5 ]
5068  LOCAL SCR_DBAND%[ CURRENT = 10, HIGH = 50, LOW = 0, STEP = 1 ]
5070  LOCAL SCR_DECAY%[ CURRENT = 99, HIGH = 99, LOW = 0, STEP = 1 ]
5072  LOCAL SCR_GAN%[ CURRENT = 20, HIGH = 500, LOW = 0, STEP = 1 ]
5074  LOCAL SCR_TRIP%[ CURRENT = 3000, HIGH = 3000, LOW = 500, STEP = 1 ]
5076  LOCAL SMAX_RPM%[ CURRENT = 1200, HIGH = 10000, LOW = 1, STEP = 1 ]
5078  LOCAL SMIN_RPM%[ CURRENT = 485, HIGH = 10000, LOW = 1, STEP = 1 ]
5080  LOCAL SOFT_TYPE%[ CURRENT = 2, HIGH = 5, LOW = 1, STEP = 1 ]
5082  LOCAL SPD_FB_CAL%[ CURRENT = 18613, HIGH = 32767, LOW = 6000, &
     STEP = 1 ]
5084  LOCAL SPD_LM%[ CURRENT = -4095, HIGH = 0, LOW = -4095, STEP = 1 ]
```

```
5086  LOCAL SPD_LP%[ CURRENT = 4095, HIGH = 4095, LOW = 0, STEP = 1 ]
5088  LOCAL TB_SPD_IN1%[ CURRENT = 0, HIGH = 4095, LOW = 0, STEP = 10 ]
5090  LOCAL TB_SPD_IN2%[ CURRENT = 0, HIGH = 4095, LOW = -4095, &
     STEP = 10 ]
5092  LOCAL TB_SPI_LM%[ CURRENT = -1024, HIGH = 0, LOW = -4095, &
     STEP = 10 ]
5094  LOCAL TB_SPI_LP%[ CURRENT = 1500, HIGH = 4095, LOW = 0, STEP = 10 ]
5096  LOCAL TEST_LEVEL%[ CURRENT = 500, HIGH = 32767, LOW = -32768, &
     STEP = 1 ]
5098  LOCAL TEST_LEV_RATE%[ CURRENT = 100, HIGH = 32767, LOW = 0, &
     STEP = 1 ]
5100  LOCAL TEST_STEP_MINUS%[ CURRENT = 0, HIGH = 0, LOW = -4095, &
     STEP = 1 ]
5102  LOCAL TEST_STEP_PLUS%[ CURRENT = 0, HIGH = 4095, LOW = 0, STEP = 1 ]
5104  LOCAL WCOS[ CURRENT = 20.0, HIGH = 57.0, LOW = 0.01, STEP = 0.01 ]
5106  LOCAL WFLDS_1[ CURRENT = 50.0, HIGH = 200.0, LOW = 30.0, &
     STEP = 0.1 ]
5108  LOCAL WFLGS_1[ CURRENT = 100.0, HIGH = 200.0, LOW = 50.0, &
     STEP = 0.1 ]
5110  LOCAL WLDS[ CURRENT = 3.0, HIGH = 20.0, LOW = 0.001, STEP = 0.001 ]
5112  LOCAL WLDS_1[ CURRENT = 60.0, HIGH = 250.0, LOW = 2.0, STEP = 0.1 ]
5114  LOCAL WLDS_2[ CURRENT = 100.0, HIGH = 200.0, LOW = 50.0, &
     STEP = 0.1 ]
5116  LOCAL WLGS_1[ CURRENT = 30.0, HIGH = 50.0, LOW = 1.0, STEP = 0.1 ]
5118  LOCAL WLGS_2[ CURRENT = 50.0, HIGH = 200.0, LOW = 30.0, STEP = 0.1 ]
5120  LOCAL WR2_FACT%[ CURRENT = 1, HIGH = 20, LOW = 1, STEP = 1 ]
5122  LOCAL ZETAS[ CURRENT = 1.1, HIGH = 100.0, LOW = 0.5, STEP = 0.1 ]
5124  LOCAL ZER_SPD_THR%[ CURRENT = 39, HIGH = 41, LOW = 0, STEP = 1 ]
6999  !*****************************************************************&
  **                                                                   **&
  **  E v e n t   a n d   O p e n   S t a t e m e n t s **&
  **                                                                   **&
  *********************************************************************
7999  !*****************************************************************&
  **                                                                   **&
  **  P r o g r a m   I n i t i a l i z a t i o n   **&
  **                                                                   **&
  *********************************************************************
8999  ! BACKLASH COMPENSATION - (CONTACT CONTROL ENGINEER BEFORE ENABLING)
9000  LLS_1_ON@ = FALSE  :! SET TRUE TO ENABLE 1ST FORWARD PATH LAG-LEAD
9010  LLS_2_ON@ = FALSE  :! SET TRUE TO ENABLE 2ND FORWARD PATH LAG_LEAD
9020  FLLS_1_ON@ = FALSE :! SET TRUE TO ENABLE FEEBACK PATH LEAD-LAG
9900  SID_SCAN_THR%=200
9910  SID_SPD_THR%=750
9998  ! ****************************************************************&
  ! SPEED LOOP; DBAR=1; FBAR=1 &
  ! INDEX: &
  ! 10000 - OL / TEST REFERENCE &
  ! 11000 - SPEED REFERENCE &
  ! 14000 - WR2 COMPENSATION &
  ! 19000 - RESOLVER FEEDBACK &
  ! 20000 - SPEED REGULATOR & &
  ! 25000 - SHUNT FIELD REFERENCE &
  ! 29000 - SPEED LOOP ID TEST
9999  !*****************************************************************&
  **                                                                   **&
  **      M a i n   P r o g r a m     **&
  **                                                                   **&
  *********************************************************************
10000 CALL  SCAN_LOOP( TICKS = 11 )
10100 CALL  THERMAL_OVERLOAD( I_FDBK    = TB_CML_FB%,   &
          LIM_BAR   = LIM_BAR%,     &
```

```
          OVERLOAD  = TB_OLOAD@ )
10200 CALL  MOVE( OUTPUT1 = TB_CML_ALPHA%,        &
          INPUT1  = CML_ALPHA%,       &
          OUTPUT2 = TB_FML_ALPHA%,        &
          INPUT2  = FML_ALPHA% )
10300 CALL  RAMP( RESET         = -TB_TESTLEV@,     &
          SCALE         = 182,        &
          INPUT         = TEST_LEVEL%,      &
          ACCEL_RATE    = TEST_LEV_RATE%,     &
          DECEL_RATE    = TEST_LEV_RATE%,     &
          RATE          = TEST_RATE%,     &
          OUTPUT        = TEST_LEVEL_1% )
10310 CALL  SELECT( OUTPUT  = TEST_STEP%,       &
      SELECT1 = S9_SWIT_UP@,        &
      INPUT1  = TEST_STEP_PLUS%,      &
      SELECT2 = S9_SWIT_DN@,        &
      INPUT2  = TEST_STEP_MINUS% )
11000 CALL  SELECT( OUTPUT  = SPD_REF_5%,       &
      SELECT1 = TB_RUN_RF@,       &
      INPUT1  = TB_RUN_REF%,        &
      SELECT2 = TB_JOGF_RF@,        &
      INPUT2  = JOG_FWD_REF%,       &
      SELECT3 = TB_JOGR_RF@,        &
      INPUT3  = JOG_REV_REF% )
11010 CALL  SWITCH( SELECT = TB_JOG_RT@,        &
      INPUT1 = JOG_ACC_RATE%,       &
      INPUT2 = RUN_ACC_RATE%,       &
      OUTPUT = ACC_RATE% )
11020 CALL  SWITCH( SELECT = TB_JOG_RT@,        &
      INPUT1 = JOG_DEC_RATE%,       &
      INPUT2 = RUN_DEC_RATE%,       &
      OUTPUT = DEC_RATE% )
11030 CALL  SWITCH( SELECT = TB_JOG_RT@,        &
      INPUT1 = JOG_JERK%,       &
      INPUT2 = RUN_JERK%,       &
      OUTPUT = JERK% )
11040 CALL  S_CURVE( SCALE         = 182,       &
       RESET         = -TB_RAMP_ON@,      &
       INITIAL_VALUE = TB_SPD_FB_EU%,     &
       INPUT         = SPD_REF_5%,      &
       ACCEL_RATE    = ACC_RATE%,     &
       DECEL_RATE    = DEC_RATE%,     &
       REVERSE       = -POS_SPD_REF@,     &
       JERK_RATE     = JERK%,       &
       RATE          = REF_RATE_3%,     &
       OUTPUT        = SPD_REF_4% )
11100 CALL  SELECT( OUTPUT  = SPD_REF_3%,       &
      SELECT1 = TB_TND_RF@,       &
      INPUT1  = TB_TND_REF%,        &
      SELECT2 = TB_RAMP_ON@,        &
      INPUT2  = SPD_REF_4% )
11110 CALL  SELECT( OUTPUT  = REF_RATE_2%,        &
      SELECT1 = TB_TND_RF@,       &
      INPUT1  = TB_TND_RATE%,       &
      SELECT2 = TB_RAMP_ON@,        &
      INPUT2  = REF_RATE_3% )
11200 CALL  MULTIPLY_DIVIDE( INPUT1 = 4095,       &
         INPUT2 = SPD_REF_3%,     &
         INPUT3 = GEAR_IN_EU%,      &
         OUTPUT = SPD_REF_2% )
11210 CALL  SWITCH( SELECT = TB_SPDT@,        &
      INPUT1 = TEST_LEVEL_1%,       &
```

```
      INPUT2 = SPD_REF_2%,        &
      OUTPUT = TB_SPD_REF% )
11220 CALL  COMPARE( INPUT1     = TB_SPD_REF%,      &
       INPUT2     = -1,       &
       OUTPUT_GTR = POS_SPD_REF@ )
11230 CALL  MULTIPLY_DIVIDE( INPUT1 = 4095,       &
         INPUT2 = REF_RATE_2%,      &
         INPUT3 = GEAR_IN_EU%,      &
         OUTPUT = REF_RATE_1% )
11240 CALL  SWITCH( SELECT = TB_SPDT@,        &
      INPUT1 = TEST_RATE%,        &
      INPUT2 = REF_RATE_1%,       &
      OUTPUT = TB_REF_RATE% )
11250 CALL  COMPARE( INPUT1     = TB_REF_RATE%,     &
       INPUT2     = -1,       &
       OUTPUT_GTR = POS_REF_RATE@ )
11300 CALL  ABSOLUTE_VALUE( INPUT  = TB_SPD_REF%,     &
        OUTPUT = TB_ASPD_REF% )
14200 WR2% = JBAR*409500/LIM_BAR%/WR2_FACT%
14210 CALL  MULTIPLY_DIVIDE( INPUT1 = DEC_WR2_GAIN%,    &
         INPUT2 = WR2%,       &
         INPUT3 = 1000,       &
         OUTPUT = DEC_WR2_int% )
14220 CALL  MULTIPLY_DIVIDE( INPUT1 = ACC_WR2_GAIN%,    &
         INPUT2 = WR2%,       &
         INPUT3 = 1000,       &
         OUTPUT = ACC_WR2% )
14230 CALL  AND( INPUT1 = POS_SPD_REF@,       &
         INPUT2 = -POS_REF_RATE@,       &
         OUTPUT = POS_DEC@ )
14240 CALL  AND( INPUT1 = -POS_SPD_REF@,        &
         INPUT2 = POS_REF_RATE@,        &
         OUTPUT = NEG_DEC@ )
14250 CALL  OR( INPUT1 = POS_DEC@,          &
        INPUT2 = NEG_DEC@,          &
        OUTPUT = DEC_WR2@ )
14260 CALL  SWITCH( SELECT = DEC_WR2@,        &
      INPUT1 = DEC_WR2_int%,        &
      INPUT2 = ACC_WR2%,        &
      OUTPUT = WR2_1% )
14300 CALL  MULTIPLY_DIVIDE( INPUT1 = WR2_FACT%,      &
         INPUT2 = TB_REF_RATE%,     &
         INPUT3 = 1,        &
         OUTPUT = WR2_RATE% )
14400 CALL  MULTIPLY_DIVIDE( INPUT1 = WR2_1%,     &
         INPUT2 = WR2_RATE%,      &
         INPUT3 = BASE_SPEED%,      &
         OUTPUT = TB_WR2_COMP% )
19000 CALL  PULSE_MULT( INPUT         = TB_RES_SCN_POS%,    &
         WORD_SIZE     = 16,       &
         OUTPUT        = SPD_FB_2% )
19010 CALL  MULTIPLY_DIVIDE( INPUT1 = SOFT_TYPE%,     &
         INPUT2 = SPD_FB_2%,      &
         INPUT3 = 1,        &
         OUTPUT = SPD_FB_1% )
19020 CALL  PULSE_MULT( INPUT         = SPD_FB_1%,      &
         MULTIPLIER    = SPD_FB_CAL%,    &
         OUTPUT        = TB_SPD_FB% )
19100 CALL  ABSOLUTE_VALUE( INPUT  = TB_SPD_FB%,      &
        OUTPUT = ASPD_FB% )
19110 CALL  ALARM( INPUT           = ASPD_FB%,      &
         ALARM_LOW       = TB_LOW_SPD@,     &
```

```
          LOW_LIMIT       = LOW_SPD_THR%,      &
          ALARM_LOW_LOW   = TB_ZER_SPD@,     &
          LOW_LOW_LIMIT   = ZER_SPD_THR% )
19200 CALL  MULTIPLY_DIVIDE( INPUT1 = GEAR_IN_EU%,      &
        INPUT2 = TB_SPD_FB%,     &
        INPUT3 = 4095,       &
        OUTPUT = TB_SPD_FB_EU% )
20000 CALL  SELECT( OUTPUT  = SPD_REF_1%,       &
      SELECT1 = TRUE,         &
      INPUT1  = TEST_STEP%,       &
      SELECT2 = TRUE,         &
      INPUT2  = TB_SPD_REF%,        &
      SELECT3 = TB_SPD_IN1@,        &
      INPUT3  = TB_SPD_IN1%,        &
      SELECT4 = TB_SPD_IN2@,        &
      INPUT4  = TB_SPD_IN2% )
20010 CALL  LIMIT( INPUT           = SPD_REF_1%,      &
          LIMIT_PLUS      = SPD_LP%,     &
          LIMIT_MINUS     = SPD_LM%,     &
          SATURATED_PLUS  = TB_SRF_SP@,      &
          SATURATED_MINUS = TB_SRF_SM@,      &
          OUTPUT          = TB_SPD_RFJ% )
20100 CALL  LEAD_LAG( RESET         = -FLLS_1_ON@,      &
        INITIAL_VALUE = TB_SPD_FB%,     &
        INPUT         = TB_SPD_FB%,     &
        OUTPUT        = TB_SPD_FBJ%,      &
        WLG           = WFLGS_1,      &
        WLD           = WFLDS_1 )
20200 CALL  DIFFERENCE( INPUT1 = TB_SPD_RFJ%,     &
         INPUT2 = TB_SPD_FBJ%,     &
         OUTPUT = SPD_ERR% )
20310 CALL  SWITCH( SELECT = TB_SPI_LP@,        &
      INPUT1 = TB_SPI_LP%,        &
      INPUT2 = CUR_LP%,       &
      OUTPUT = SPI_LP% )
20320 CALL  SWITCH( SELECT = TB_SPI_LM@,        &
      INPUT1 = TB_SPI_LM%,        &
      INPUT2 = CUR_LM%,       &
      OUTPUT = SPI_LM% )
20330 CALL  PROP_INT( RESET           = -TB_SPD_ON@,    &
        INPUT           = SPD_ERR%,     &
        HOLD_PLUS       = TB_CRF_SP@,     &
        HOLD_MINUS      = TB_CRF_SM@,     &
        LIMIT_PLUS      = SPI_LP%,      &
        LIMIT_MINUS     = SPI_LM%,      &
        OUTPUT          = TB_SPI_OUT%,    &
        SATURATED_PLUS  = TB_SPI_SP@,     &
        SATURATED_MINUS = TB_SPI_SM@,     &
        KP              = KPS,      &
        WLD             = WLDS )
20340 CALL  LEAD_LAG( RESET         = -LLS_1_ON@,     &
        INITIAL_VALUE = TB_SPI_OUT%,      &
        INPUT         = TB_SPI_OUT%,      &
        OUTPUT        = SPD_OUT_1%,     &
        WLG           = WLGS_1,     &
        WLD           = WLDS_1 )
20350 CALL  LEAD_LAG( RESET         = -LLS_2_ON@,     &
        INITIAL_VALUE = SPD_OUT_1%,     &
        INPUT         = SPD_OUT_1%,     &
        OUTPUT        = SPD_OUT_2%,     &
        WLG           = WLGS_2,     &
        WLD           = WLDS_2 )
```

```
20400 CALL  SELECT( OUTPUT  = SPD_OUT_3%,       &
      SELECT1 = TB_CMLT@,       &
      INPUT1  = TEST_STEP%,       &
      SELECT2 = TB_SPD_ON@,       &
      INPUT2  = SPD_OUT_2%,       &
      SELECT3 = TB_SPD_ON@,       &
      INPUT3  = TB_WR2_COMP%,       &
      SELECT4 = TB_CML_IN1@,       &
      INPUT4  = TB_CML_IN1_int% )
20410 CALL  LIMIT( INPUT           = SPD_OUT_3%,      &
           LIMIT_PLUS      = CUR_LP%,     &
           LIMIT_MINUS     = CUR_LM%,     &
           SATURATED_PLUS  = TB_CRF_SP@,      &
           SATURATED_MINUS = TB_CRF_SM@,      &
           OUTPUT          = SPD_OUT_4% )
20420 CALL  SWITCH( SELECT = TB_SPD_ID@,        &
      INPUT1 = SID_TRQ_REF_int%,        &
      INPUT2 = SPD_OUT_4%,        &
      OUTPUT = TB_CML_REF% )
20500 CALL  DIFFERENCE( INPUT1 = TB_SPD_FB%,      &
          INPUT2 = TB_SPD_REF%,     &
          OUTPUT = TB_STEP_FB% )
25100 CALL  SWITCH( SELECT = TB_NO_FLDE@,       &
      INPUT1 = 4095,          &
      INPUT2 = FML_ECON_REF%,       &
      OUTPUT = FML_LP% )
25110 CALL  SELECT( OUTPUT  = FML_REF%,       &
      SELECT1 = TB_FMLT@,       &
      INPUT1  = TEST_LEVEL_1%,      &
      SELECT2 = TB_FMLT@,       &
      INPUT2  = TEST_STEP%,       &
      SELECT3 = -TB_FMLT@,        &
      INPUT3  = FML_GEAR_IN_REF% )
25120 CALL  LIMIT( INPUT           = FML_REF%,      &
           LIMIT_PLUS      = FML_LP%,     &
           OUTPUT          = TB_FML_REF% )
29000 CALL  AND( INPUT1 = TB_SPD_ID@,       &
         INPUT2 = TB_M_FDBK@,       &
         OUTPUT = SPD_ID@ )
29010 REM  "SID1A.INC"
32767 END
```