U.S. Securities and Exchange Commission
Washington, D.C. 20549

OMB APPROVAL
OMB Number: 3235-0327
Expires: July 31, 2004
Estimated average burden
hours per response.... 0.15

FORM SE

FORM FOR SUBMISSION OF PAPER FORMAT EXHIBITS BY ELECTRONIC FILERS

Residential Asset Mortgage Products, Inc.
Exact Name of Registrant as Specified in Charter

0001099391
Registrant CIK Number

FOR 10/29/04
Current Report on Form 8-K 2004-RS10
Electronic Report, Schedule or Registration Statement of Which the Documents Are a Part (give period of report)

333-117232
SEC File Number of Registration Statement

04047919

Name of Person Filing the Document
(if Other than the Registrant)

RECD S.E.C.
NOV 1 2004
1086

SIGNATURES

Filings Made By the Registrant:

The Registrant has duly caused this form to be signed on its behalf by the undersigned, thereunto duly authorized in the City of Minneapolis, State of Minnesota, on the 29th day October 2004.

Residential Asset Mortgage Products, Inc.
(Registrant)

By: _____
Name: Mark White
Title: Vice President

PROCESSED
NOV 0 3 2004
THOMSON FINANCIAL

Filings Made by Person Other Than the Registrant:

After reasonable inquiry and to the best of my knowledge and belief, I certify on _____, 2004, that the information set forth in this statement is true and complete.

By: _____
(Name)
(Title)

```
!  RFC04RS10.CDI  #CMOVER_3.0D ASSET_BACKED_HOMEEQUITY  !  MAX_CF_VECTSIZE 620

!
!! Created by Intex Deal Maker v3.7.229  ,  subroutines 3.1a1
!!   10/29/2004   10:48 AM
!
DEAL_COMMENT _
"The tables and other statistical analyses (the 'Hypothetical Performance
Data') that you will produce using Intex with the attached information are
privileged and intended solely for use by you (the party to whom CSFB LLC
provided the computer model used to generate them).  The Hypothetical
Performance Data will be generated by you using a computer model prepared by
CSFB LLC in reliance upon information furnished by the issuer of the
securities and its affiliates, the accuracy and completeness of which has not
been verified by CSFB LLC or any other person.  The computer model that you
will use to prepare the Hypothetical Performance Data was furnished to you
solely by CSFB LLC and not by the issuer of the securities.  It may not be (a)
used for any purpose other than to make a preliminary evaluation of the
referenced securities or (b) provided by you to any third party other than
your legal, tax, financial and/or accounting advisors for the purposes of
evaluating the Hypothetical Performance Data.  You agree that the Hypothetical
Performance Data will be generated by or on behalf of you, and that neither
CSFB LLC nor anyone acting on its behalf has generated or is in any way
responsible for any Hypothetical Performance Data. ; _
 Numerous assumptions were used in preparing the computer model you will use
to generate the Hypothetical Performance Data.  Those assumptions may or may
not be reflected in the Hypothetical Performance Data.  As such, no assurance
can be give as to the Hypothetical Performance Data's accuracy,
appropriateness or completeness in any particular context nor as to whether
the Hypothetical Performance Data and/or the assumptions upon which it is
based reflect present market conditions of future market performance.  The
Hypothetical Performance Data should not be construed as either projections or
predictions or as legal, tax, financial or accounting advice."
!
!  Modeled in the Intex CMO Modeling Language, (WNYC120016657)
!  which is copyright (c) 2004 by Intex Solutions, Inc.
!  Intex shall not be held liable for the accuracy of this data
!  nor for the accuracy of information which is derived from this data.
!
 COLLAT_GROUPS 1 2
!
  DEFINE PREPAY PPC  GROUP 1 RISE_PERS 1 START_CPR 0 END_CPR 0
  DEFINE PREPAY PPC GROUP 2 RAMP 2 4.54545454545455 7.09090909090909
9.63636363636364 12.1818181818182 14.7272727272727 17.2727272727273
19.8181818181818 22.3636363636364 24.9090909090909 27.4545454545455 30 30 30
30 30 30 30 30 30 30 30 30 50 50 50 50 50 35



!
!
  DEFINE CONSTANT #OrigCollBal = 1250000000.00
  DEFINE CONSTANT #OrigCollBal1 = 300000000.00
  DEFINE CONSTANT #OrigCollBal2 = 950000000.00
!
  DEFINE CONSTANT #OrigBondBal = 1250000000.00
  DEFINE CONSTANT #OrigBondBal1 = 300000000.00
  DEFINE CONSTANT #OrigBondBal2 = 950000000.00
!
  DEFINE CONSTANT #SpecSenEnhPct1 = 33%
  DEFINE CONSTANT #AITargPct = 67.000000000000%
  DEFINE CONSTANT #MI1TargPct = 81.500000000000%
  DEFINE CONSTANT #MI2TargPct = 88.000000000000%
  DEFINE CONSTANT #MI3TargPct = 93.000000000000%
  DEFINE CONSTANT #SBITargPct = 93.000000000000%
  DEFINE CONSTANT #SpecSenEnhPct2 = 46.5%
  DEFINE CONSTANT #AIITargPct = 53.500000000000%
```

```
  DEFINE CONSTANT #MII1TargPct = 68.000000000000%
  DEFINE CONSTANT #MII2TargPct = 79.500000000000%
  DEFINE CONSTANT #MII3TargPct = 84.000000000000%
  DEFINE CONSTANT #MII4TargPct = 88.500000000000%
  DEFINE CONSTANT #MII5TargPct = 92.000000000000%
  DEFINE CONSTANT #SBIITargPct = 92.000000000000%
  DEFINE #BondBal1                        = 300000000.00
  DEFINE #BondBal2                        = 950000000.00
  DEFINE #BondBal                         = 1250000000.00
!
   FULL_DEALNAME:        RAMP Series 2004-RS10 Trust
!
   DEAL SIZE:            1250000000.00
   PRICING SPEED:        GROUP 1 HEP 20%
   PRICING SPEED:        GROUP 2 PPC 100%
!  ISSUE DATE:           20041001
   SETTLEMENT DATE:      20041029
!
  Record date delay:     24
!
 DEFINE TR_INDEXDEPS_ALL
!
DEFINE TRANCHE "AI1", "AI2", "AI3", "AI4", "AI5", "AI6", "MI1", "MI2", "MI3",
"SBI", "AII2", "AII1", "AII3", "AII4", "MII1", "MII2", "MII3", "MII4", "MII5",
"SBII", "AII4B", "AII4A", "R_1", "R_2"
!
 DEFINE VARNAMES #ReqPerc1, #TrigEnhFrac1, #CumLossShft1, #TrigCumLossFrac1,
#ReqPerc2, #TrigEnhFrac2, #CumLossShft2, #TrigCumLossFrac2, #StepDown1,
#StepDown2
!
  MASTER_SERVICE_FEE    GROUP 1     0.05
!
  MASTER_SERVICE_FEE    GROUP 2     0.05
!
   DEAL_CLOCK_INFO _
       ISSUE_CDU_DATE             20041001 _
       DEAL_FIRSTPAY_DATE         20041125
!
!
  DEFINE #FloorCollat1        = 0.5% * #OrigCollBal1
  DEFINE #SpecOCTarg1         = 3.50% * #OrigCollBal1
  DEFINE STANDARDIZE OC_ACTUAL_VAL       GROUP 1          #OC1            =
0.00
  DEFINE STANDARDIZE OCT_INITVAL         GROUP 1 CONSTANT #InitOCTarg1    =
3.50% * #OrigCollBal1
  DEFINE STANDARDIZE OCT_STEPDOWN_MONTH  GROUP 1 CONSTANT #StepDownDate1  = 37
  DEFINE STANDARDIZE OCT_STEPDOWN_FRAC   GROUP 1 CONSTANT #StepOCFrac1    =
0.07
  DEFINE STANDARDIZE EXCESS_INTEREST     GROUP 1          #XSSpread1      = 0
  DEFINE STANDARDIZE OCT_FLOOR           GROUP 1 CONSTANT #FloorOCTarg1   =
#FloorCollat1
  DEFINE STANDARDIZE OCT_VAL             GROUP 1 DYNAMIC  #Octval1        =
#SpecOCTarg1
!
  DEFINE #FloorCollat2        = 0.50% * #OrigCollBal2
  DEFINE #SpecOCTarg2         = 4.00% * #OrigCollBal2
  DEFINE STANDARDIZE OC_ACTUAL_VAL       GROUP 2          #OC2            =
0.00
  DEFINE STANDARDIZE OCT_INITVAL         GROUP 2 CONSTANT #InitOCTarg2    =
4.00% * #OrigCollBal2
  DEFINE STANDARDIZE OCT_STEPDOWN_MONTH  GROUP 2 CONSTANT #StepDownDate2  = 37
  DEFINE STANDARDIZE OCT_STEPDOWN_FRAC   GROUP 2 CONSTANT #StepOCFrac2    =
0.08
  DEFINE STANDARDIZE EXCESS_INTEREST     GROUP 2          #XSSpread2      = 0
  DEFINE STANDARDIZE OCT_FLOOR           GROUP 2 CONSTANT #FloorOCTarg2   =
#FloorCollat2
  DEFINE STANDARDIZE OCT_VAL             GROUP 2 DYNAMIC  #Octval2        =
#SpecOCTarg2
```

```
!
  DEFINE DYNAMIC STICKY #NetRate  = ( COLL_I_MISC("COUPON") ) / COLL_PREV_BAL
* 1200
  DEFINE DYNAMIC STICKY #NetRate1  = ( COLL_I_MISC("COUPON",1) ) /
COLL_PREV_BAL(1) * 1200
  DEFINE DYNAMIC STICKY #NetRate2  = ( COLL_I_MISC("COUPON",2) ) /
COLL_PREV_BAL(2) * 1200
!
  DEFINE DYNAMIC STICKY #NetRate2Actual360 = #Netrate2 * 30 /
DAYS_DIFF(CURDATE , MONTHS_ADD(CURDATE,-1))
!
  DEFINE DYNAMIC STICKY #NetRate1Actual360 = #Netrate1 * 30 /
DAYS_DIFF(CURDATE , MONTHS_ADD(CURDATE,-1))
!
  DEFINE TABLE "OC_CUMLOSS1" (49, 2) = "MONTH" "OC_CUMLOSS_FRAC1"
      37.1   0.0175
      38.1   0.01854166667
      39.1   0.01958333333
      40.1   0.020625
      41.1   0.02166666667
      42.1   0.02270833333
      43.1   0.02375
      44.1   0.02479166667
      45.1   0.02583333333
      46.1   0.026875
      47.1   0.02791666667
      48.1   0.02895833333
      49.1   0.03
      50.1   0.03083333333
      51.1   0.03166666667
      52.1   0.0325
      53.1   0.03333333333
      54.1   0.03416666667
      55.1   0.035
      56.1   0.03583333333
      57.1   0.03666666667
      58.1   0.0375
      59.1   0.03833333333
      60.1   0.03916666667
      61.1   0.04
      62.1   0.040625
      63.1   0.04125
      64.1   0.041875
      65.1   0.0425
      66.1   0.043125
      67.1   0.04375
      68.1   0.044375
      69.1   0.045
      70.1   0.045625
      71.1   0.04625
      72.1   0.046875
      73.1   0.0475
      74.1   0.04770833333
      75.1   0.04791666667
      76.1   0.048125
      77.1   0.04833333333
      78.1   0.04854166667
      79.1   0.04875
      80.1   0.04895833333
      81.1   0.04916666667
      82.1   0.049375
      83.1   0.04958333333
      84.1   0.04979166667
      360.1   0.05
!
  DEFINE TABLE "OC_CUMLOSS2" (37, 2) = "MONTH" "OC_CUMLOSS_FRAC2"
      37.1   0.04
      38.1   0.041875
```

```
      39.1   0.04375
      40.1   0.045625
      41.1   0.0475
      42.1   0.049375
      43.1   0.05125
      44.1   0.053125
      45.1   0.055
      46.1   0.056875
      47.1   0.05875
      48.1   0.060625
      49.1   0.0625
      50.1   0.06395833333
      51.1   0.06541666667
      52.1   0.066875
      53.1   0.06833333333
      54.1   0.06979166667
      55.1   0.07125
      56.1   0.07270833333
      57.1   0.07416666667
      58.1   0.075625
      59.1   0.07708333333
      60.1   0.07854166667
      61.1   0.08
      62.1   0.080625
      63.1   0.08125
      64.1   0.081875
      65.1   0.0825
      66.1   0.083125
      67.1   0.08375
      68.1   0.084375
      69.1   0.085
      70.1   0.085625
      71.1   0.08625
      72.1   0.086875
      360.1   0.0875
!
  DEFINE TABLE "NASAI" (5, 2) = "MONTH" "NAS_FRACAI"
      36.1   0%
      60.1   45%
      72.1   80%
      84.1   100%
      96.1   300%
!
  DEFINE #COUPON_AII4 = 0
  DEFINE #COUPON_AII4B = 0
  DEFINE #COUPON_AII4A = 0
!
TOLERANCE CLEANUP 0.00
!
TOLERANCE WRITEDOWN_0LOSS 9999999999999.00
TOLERANCE INTEREST        9999999999999.00
!
  INITIAL INDEX    LIBOR_6MO         2.211
  INITIAL INDEX    LIBOR_1YR         2.436
  INITIAL INDEX    CMT_1YR           2.205
  INITIAL INDEX    LIBOR_1MO         1.910
!
ifdef #CMOVER_3.0E3 _
DEFINE MACRO #TotalPaybackDue[1] = TOTAL_PAYBACK_DUE({#1})
!
ifndef #CMOVER_3.0E3 _
DEFINE MACRO #TotalPaybackDue[1] = COLL_BAL
!
!
Tranche "AI1" SEN_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AI1"} LT 0.01);
   Block 103200000.00 at 2.09 GROUP 1  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate1 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
```

```
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     1 * LIBOR_1MO + 0.18
     0     999
!
Tranche "AI2" SEN_FIX ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AI2"} LT 0.01);
   Block 27900000.00 at 3.73 GROUP 1  FREQ M _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
!
Tranche "AI3" SEN_FIX ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AI3"} LT 0.01);
   Block 36300000.00 at 4.09 GROUP 1  FREQ M _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
!
Tranche "AI4" SEN_FIX_CAP ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AI4"} LT
0.01);
   Block 39600000.00 at 4.78 GROUP 1  FREQ M FLOAT _
          COUPONCAP 30360 NONE ( #NetRate1 ); _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
    ( 4.78 )
    0    999
!
Tranche "AI5" SEN_FIX_CAP ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AI5"} LT
0.01);
   Block 27900000.00 at 5.68 GROUP 1  FREQ M FLOAT _
          COUPONCAP 30360 NONE ( #NetRate1 ); _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
    ( IF ((COLL_BAL("LAGMON_2",1) / #OrigCollBal1) < 10%) THEN 6.18 ELSE 5.68
)
    0    999
!
Tranche "AI6" SEN_NAS_FIX_CAP ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AI6"} LT
0.01);
   Block 26100000.00 at 4.69 GROUP 1  FREQ M FLOAT _
          COUPONCAP 30360 NONE ( #NetRate1 ); _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
    ( 4.69 )
    0    999
!
Tranche "MI1" MEZ_FIX_CAP ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"MI1"} LT
0.01);
   Block 21750000.00 at 5.31 GROUP 1  FREQ M FLOAT _
          COUPONCAP 30360 NONE ( #NetRate1 ); _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
    ( IF ((COLL_BAL("LAGMON_2",1) / #OrigCollBal1) < 10%) THEN 5.81 ELSE 5.31
)
    0    999
!
Tranche "MI2" MEZ_FIX_CAP ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"MI2"} LT
0.01);
   Block 9750000.00 at 5.71 GROUP 1  FREQ M FLOAT _
          COUPONCAP 30360 NONE ( #NetRate1 ); _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
    ( IF ((COLL_BAL("LAGMON_2",1) / #OrigCollBal1) < 10%) THEN 6.21 ELSE 5.71
)
    0    999
!
Tranche "MI3" MEZ_FIX_CAP ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"MI3"} LT
0.01);
   Block 7500000.00 at 5.9 GROUP 1  FREQ M FLOAT _
          COUPONCAP 30360 NONE ( #NetRate1 ); _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
```

```
          Delay 24  Dated 20041001  Next 20041125
    ( IF ((COLL_BAL("LAGMON_2",1) / #OrigCollBal1) < 10%) THEN 6.4 ELSE 5.9 )
    0    999
!
Tranche "SBI" JUN_OC_RES_NO ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"SBI"} LT
0.01);
   Block 0.00 at 0 GROUP 1  FREQ M _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
!
Tranche "AII2" SEN_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AII2"} LT 0.01);
   Block 150000000.00 at 2.15 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     1 * LIBOR_1MO + 0.24
     0     14
!
Tranche "AII1" SEN_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AII1"} LT 0.01);
   Block 351700000.00 at 2.09 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     1 * LIBOR_1MO + 0.18
     0     14
!
Tranche "AII3" SEN_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AII3"} LT 0.01);
   Block 235400000.00 at 2.25 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     (1 * LIBOR_1MO + ( IF ((COLL_BAL("LAGMON_2",2) / #OrigCollBal2) < 10%)
THEN 0.68 ELSE 0.34 ))
     0     14
!
Tranche "AII4" SEN_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"AII4"} LT 0.01);
   Block 30000000.00 at 2.4 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     (1 * LIBOR_1MO + ( IF ((COLL_BAL("LAGMON_2",2) / #OrigCollBal2) < 10%)
THEN 0.98 ELSE 0.49 ))
     0     14
!
Tranche "MII1" MEZ_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"MII1"} LT 0.01);
   Block 68900000.00 at 2.61 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     (1 * LIBOR_1MO + ( IF ((COLL_BAL("LAGMON_2",2) / #OrigCollBal2) < 10%)
THEN 1.05 ELSE 0.7 ))
     0     14
!
Tranche "MII2" MEZ_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"MII2"} LT 0.01);
   Block 54600000.00 at 3.16 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     (1 * LIBOR_1MO + ( IF ((COLL_BAL("LAGMON_2",2) / #OrigCollBal2) < 10%)
THEN 1.875 ELSE 1.25 ))
     0     14
!
```

```
Tranche "MII3" MEZ_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"MII3"} LT 0.01);
   Block 21400000.00 at 3.41 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     (1 * LIBOR_1MO + ( IF ((COLL_BAL("LAGMON_2",2) / #OrigCollBal2) < 10%)
THEN 2.25 ELSE 1.5 ))
     0     14
!
Tranche "MII4" MEZ_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"MII4"} LT 0.01);
   Block 21400000.00 at 3.76 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     (1 * LIBOR_1MO + ( IF ((COLL_BAL("LAGMON_2",2) / #OrigCollBal2) < 10%)
THEN 2.775 ELSE 1.85 ))
     0     14
!
Tranche "MII5" MEZ_FLT ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"MII5"} LT 0.01);
   Block 16600000.00 at 3.86 GROUP 2  FREQ M FLOAT RESET M _
          COUPONCAP 30360 NONE ( #NetRate2 * 30 / (IF CURDATE EQ
DEAL_FIRST_PAYDATE THEN 26 ELSE 30) ); _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE _
          Delay 0  Dated 20041029  Next 20041125
     (1 * LIBOR_1MO + ( IF ((COLL_BAL("LAGMON_2",2) / #OrigCollBal2) < 10%)
THEN 2.925 ELSE 1.95 ))
     0     14
!
Tranche "SBII" JUN_OC_RES_NO ! PAID_DOWN_WHEN ({#TotalPaybackDue}{"SBII"} LT
0.01);
   Block 0.00 at 0 GROUP 2  FREQ M _
          DAYCOUNT 30360 BUSINESS_DAY NONE _
          Delay 24  Dated 20041001  Next 20041125
!
Tranche "AII4B" MODELING EXCHANGE
   Block 17400000.00 FLOAT GROUP 2 _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE  FREQ M _
          Delay 0  Dated 20041029  Next 20041125
     ( #COUPON_AII4B)
    0    999
!
Tranche "AII4A" MODELING EXCHANGE
   Block 12600000.00 FLOAT GROUP 2 _
          DAYCOUNT ACTUAL360 BUSINESS_DAY NONE  FREQ M _
          Delay 0  Dated 20041029  Next 20041125
     ( #COUPON_AII4A)
    0    999
!
Tranche "R_1" JUN_RES
   Block 300000000.00 at 0 GROUP 1 NOTIONAL WITH GROUP 1 _
         DAYCOUNT 30360 BUSINESS_DAY NONE _
         FREQ M   Delay 24  Dated 20041001  Next 20041125
!
Tranche "R_2" JUN_RES
   Block 950000000.00 at 0 GROUP 2 NOTIONAL WITH GROUP 2 _
         DAYCOUNT 30360 BUSINESS_DAY NONE _
         FREQ M   Delay 24  Dated 20041001  Next 20041125
!
  Tranche "#OC1"              SYMVAR
  Tranche "#SpecOCTarg1"      SYMVAR
!
  Tranche "#OC2"              SYMVAR
  Tranche "#SpecOCTarg2"      SYMVAR
!
Tranche "COF2" PSEUDO
   Block    USE PCT 100.0 100.0 of MII1#1
```

```
   Block    USE PCT 100.0 100.0 of MII2#1
   Block    USE PCT 100.0 100.0 of MII3#1
   Block    USE PCT 100.0 100.0 of MII4#1
   Block    USE PCT 100.0 100.0 of MII5#1
   Block    USE PCT 100.0 100.0 of AII2#1
   Block    USE PCT 100.0 100.0 of AII1#1
   Block    USE PCT 100.0 100.0 of AII3#1
   Block    USE PCT 100.0 100.0 of AII4#1
!
Tranche "COF1" PSEUDO
   Block    USE PCT 100.0 100.0 of AI6#1
   Block    USE PCT 100.0 100.0 of AI1#1
   Block    USE PCT 100.0 100.0 of AI2#1
   Block    USE PCT 100.0 100.0 of AI3#1
   Block    USE PCT 100.0 100.0 of AI4#1
   Block    USE PCT 100.0 100.0 of AI5#1
   Block    USE PCT 100.0 100.0 of MI1#1
   Block    USE PCT 100.0 100.0 of MI2#1
   Block    USE PCT 100.0 100.0 of MI3#1
!
!
DEFINE PSEUDO_TRANCHE COLLAT _
   Delay 24 Dated 20041001 Next 20041125 Settle 20041029
DEFINE PSEUDO_TRANCHE COLLAT GROUP 1 _
   Delay 24 Dated 20041001 Next 20041125 Settle 20041029
DEFINE PSEUDO_TRANCHE COLLAT GROUP 2 _
   Delay 24 Dated 20041001 Next 20041125 Settle 20041029
!
  CREDIT_SUPPORT_BASIS GROUP_DEAL
!
ifdef #cmover_3.0g2 _
  DEFINE MACRO #BalanceRatio[1] = BALANCE_RATIO {#1}
!
ifndef #cmover_3.0g2 _
  DEFINE MACRO #BalanceRatio[1] =
!
  CLASS "RESID_1"  NO_BUILD_TRANCHE _
                    = "R_1#1"
  CLASS "RESID_2"  NO_BUILD_TRANCHE _
                    = "R_2#1"
  CLASS "AI6"       NO_BUILD_TRANCHE _
                    SHORTFALL_PAYBACK  COUPONCAP TRUE _
                    SHORTFALL_EARN_INT COUPONCAP TRUE _
                    = "AI6"
  CLASS "AI1"       NO_BUILD_TRANCHE _
                    SHORTFALL_PAYBACK  COUPONCAP TRUE _
                    SHORTFALL_EARN_INT COUPONCAP TRUE _
                    = "AI1"
  CLASS "AI2"       NO_BUILD_TRANCHE _
                    SHORTFALL_PAYBACK  COUPONCAP TRUE _
                    SHORTFALL_EARN_INT COUPONCAP TRUE _
                    = "AI2"
  CLASS "AI3"       NO_BUILD_TRANCHE _
                    SHORTFALL_PAYBACK  COUPONCAP TRUE _
                    SHORTFALL_EARN_INT COUPONCAP TRUE _
                    = "AI3"
  CLASS "AI4"       NO_BUILD_TRANCHE _
                    SHORTFALL_PAYBACK  COUPONCAP TRUE _
                    SHORTFALL_EARN_INT COUPONCAP TRUE _
                    = "AI4"
  CLASS "AI5"       NO_BUILD_TRANCHE _
                    SHORTFALL_PAYBACK  COUPONCAP TRUE _
                    SHORTFALL_EARN_INT COUPONCAP TRUE _
                    = "AI5"
  CLASS "MI1"       NO_BUILD_TRANCHE _
                    SHORTFALL_PAYBACK  COUPONCAP TRUE _
                    SHORTFALL_EARN_INT COUPONCAP TRUE _
                    = "MI1"
```

```
  CLASS "MI2"        NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "MI2"
  CLASS "MI3"        NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "MI3"
  CLASS "SBI"        NO_BUILD_TRANCHE _
                     = "SBI"
  CLASS "AII2"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "AII2"
  CLASS "AII1"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "AII1"
  CLASS "AII3"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "AII3"
  CLASS "AII4B"      DISTRIB_CLASS RULES _
                     = "AII4B"
  CLASS "AII4A"      DISTRIB_CLASS RULES _
                     = "AII4A"
  CLASS "MII1"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "MII1"
  CLASS "MII2"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "MII2"
  CLASS "MII3"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "MII3"
  CLASS "MII4"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "MII4"
  CLASS "MII5"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                     = "MII5"
  CLASS "SBII"       NO_BUILD_TRANCHE _
                     = "SBII"
  CLASS "AAANN" DISTRIB_CLASS PRORATA  WRITEDOWN_BAL PRORATA _
                     = "AI1" "AI2" "AI3" "AI4" "AI5"
  CLASS "AI"         WRITEDOWN_BAL PRORATA ALLOCATION _
                     = "AI6" "AAANN"
  CLASS "AIIA" DISTRIB_CLASS PRORATA  WRITEDOWN_BAL PRORATA _
                     = "AII2" "AII4B"
  CLASS "AII4"       NO_BUILD_TRANCHE _
                     SHORTFALL_PAYBACK  COUPONCAP TRUE _
                     SHORTFALL_EARN_INT COUPONCAP TRUE _
                    SHORTFALL_PAYBACK PRINCIPAL_LOSS TRUE  SHORTFALL_EARN_INT
INTEREST TRUE   = "AII4", _
    COMBINE_CLASSES {#BalanceRatio}{1.0} = "AII4B" "AII4A"
  CLASS "AIIB" DISTRIB_CLASS PRORATA  WRITEDOWN_BAL PRORATA _
                     = "AII1" "AII3" "AII4A"
  CLASS "AII"        WRITEDOWN_BAL PRORATA ALLOCATION _
                     = "AIIA" "AIIB"
!
!
  CLASS "GRP1" _
                          WRITEDOWN_BAL RULES _
```

```
                  DISTRIB_CLASS RULES _
                  SHORTFALL_PAYBACK PRINCIPAL_LOSS TRUE _
                  SHORTFALL_EARN_INT INTEREST TRUE _
                    = "AI" "MI1" "MI2" "MI3" "SBI"  "RESID_1"
  CLASS "GRP2" _
                          WRITEDOWN_BAL RULES _
                  DISTRIB_CLASS RULES _
                  SHORTFALL_PAYBACK PRINCIPAL_LOSS TRUE _
                  SHORTFALL_EARN_INT INTEREST TRUE _
                    = "AII" "MII1" "MII2" "MII3" "MII4" "MII5" "SBII"
"RESID_2"
!
  CLASS "ROOT"  ROOT_LIST = "GRP1" "GRP2"
!
  GROUP 0       ROOT      = 1 2
!

!
  DEFINE PSEUDO_TRANCHE CLASS "AII"          Delay 0  Dated 20041029  Next
20041125 DAYCOUNT ACTUAL360 BUSINESS_DAY NONE
!
  DEFINE PSEUDO_TRANCHE CLASS "AI"           Delay 24  Dated 20041001  Next
20041125 DAYCOUNT 30360 BUSINESS_DAY NONE
!
  DEFINE PSEUDO_TRANCHE CLASS "AAANN"        Delay 24  Dated 20041001  Next
20041125 DAYCOUNT 30360 BUSINESS_DAY NONE
!
  DEFINE PSEUDO_TRANCHE CLASS "AIIA"         Delay 0  Dated 20041029  Next
20041125 DAYCOUNT ACTUAL360 BUSINESS_DAY NONE
!
  DEFINE PSEUDO_TRANCHE CLASS "AIIB"         Delay 0  Dated 20041029  Next
20041125 DAYCOUNT ACTUAL360 BUSINESS_DAY NONE
!
  DEFINE PSEUDO_TRANCHE CLASS "AII4B"        Delay 0  Dated 20041029  Next
20041125 DAYCOUNT ACTUAL360 BUSINESS_DAY NONE
!
  DEFINE PSEUDO_TRANCHE CLASS "AII4A"        Delay 0  Dated 20041029  Next
20041125 DAYCOUNT ACTUAL360 BUSINESS_DAY NONE
!
!
  CROSSOVER When 0
!
TRIGGER "StepUp-CumLoss1" _
        FULL_NAME   "Group 1 Step Up Cumulative Loss Trigger" _
        ORIG_TESTVAL   0.000% _
        TESTVAL        ( #TrigCumLossFrac1); _
        ORIG_TARGETVAL  1.75% _
        TARGETVAL       (#CumLossShft1); _
        TRIGVAL          LODIFF
!
TRIGGER "StepUp-DlqEnh1" _
        FULL_NAME   "Group 1 Step Up Enhancement Delinquency Trigger" _
        ORIG_TESTVAL   0.000% _
        TESTVAL        ( #TrigEnhFrac1); _
        ORIG_TARGETVAL  6.24% _
        TARGETVAL       (#ReqPerc1); _
        TRIGVAL          LODIFF
!
TRIGGER "STEPUP_TRIGGER1" _
        FULL_NAME   "Group 1 Step Up Trigger" _
        DEFINITION "A Step Up Trigger exists, if_
;(1) a percentage calculated as the quotient of the amount of cumulative_
 realized losses divided by the original collateral balance exceeds the target
defined by a schedule;_
                      Month <=          %;_
                           37           1.75%;  _
                           38           1.854166667%;  _
                           39           1.958333333%;  _
```

```
                                40          2.0625%;  _
                                41          2.166666667%;  _
                                42          2.270833333%;  _
                                43          2.375%;  _
                                44          2.479166667%;  _
                                45          2.583333333%;  _
                                46          2.6875%;  _
                                47          2.791666667%;  _
                                48          2.895833333%;  _
                                49          3%;  _
                                50          3.083333333%;  _
                                51          3.166666667%;  _
                                52          3.25%;  _
                                53          3.333333333%;  _
                                54          3.416666667%;  _
                                55          3.5%;  _
                                56          3.583333333%;  _
                                57          3.666666667%;  _
                                58          3.75%;  _
                                59          3.833333333%;  _
                                60          3.916666667%;  _
                                61          4%;  _
                                62          4.0625%;  _
                                63          4.125%;  _
                                64          4.1875%;  _
                                65          4.25%;  _
                                66          4.3125%;  _
                                67          4.375%;  _
                                68          4.4375%;  _
                                69          4.5%;  _
                                70          4.5625%;  _
                                71          4.625%;  _
                                72          4.6875%;  _
                                73          4.75%;  _
                                74          4.770833333%;  _
                                75          4.791666667%;  _
                                76          4.8125%;  _
                                77          4.833333333%;  _
                                78          4.854166667%;  _
                                79          4.875%;  _
                                80          4.895833333%;  _
                                81          4.916666667%;  _
                                82          4.9375%;  _
                                83          4.958333333%;  _
                                84          4.979166667%;  _
                                360         5%;  _
_
 or;(2) the aggregate principal balance of all delinquent loans * 1_
 as a percentage of the respective collateral balance exceeds :_
 48% * the Senior Enhancement Percentage."_
        IMPACT     "If a Step Up Trigger is in effect the OC target will
change to_
 the last value before the trigger occurred if a stepdown has_
 occurred.  It has no effect if a stepdown has not occurred."  _
        EFFECTIVE_WHEN (If #StepDown1 Then TRIG_EFFECTIVE_YES Else
TRIG_EFFECTIVE_ALWAYSPASS); _
        TRIGVAL FORMULA ( min(TRIGGER("StepUp-CumLoss1","TRIGVAL"),
TRIGGER("StepUp-DlqEnh1","TRIGVAL")));
!
TRIGGER "StepUp-CumLoss2" _
        FULL_NAME   "Group 2 Step Up Cumulative Loss Trigger" _
        ORIG_TESTVAL   0.000% _
        TESTVAL        ( #TrigCumLossFrac2); _
        ORIG_TARGETVAL  4% _
        TARGETVAL       (#CumLossShft2); _
        TRIGVAL          LODIFF
!
TRIGGER "StepUp-DlqEnh2" _
```

```
        FULL_NAME   "Group 2 Step Up Enhancement Delinquency Trigger" _
        ORIG_TESTVAL   0.000% _
        TESTVAL        ( #TrigEnhFrac2); _
        ORIG_TARGETVAL  6.54589473684211% _
        TARGETVAL       (#ReqPerc2); _
        TRIGVAL          LODIFF
!
TRIGGER "STEPUP_TRIGGER2" _
        FULL_NAME   "Group 2 Step Up Trigger" _
        DEFINITION "A Step Up Trigger exists, if_
;(1) a percentage calculated as the quotient of the amount of cumulative_
 realized losses divided by the original collateral balance exceeds the target
defined by a schedule;_
                        Month <=          %;_
                             37           4%;  _
                             38           4.1875%;  _
                             39           4.375%;  _
                             40           4.5625%;  _
                             41           4.75%;  _
                             42           4.9375%;  _
                             43           5.125%;  _
                             44           5.3125%;  _
                             45           5.5%;  _
                             46           5.6875%;  _
                             47           5.875%;  _
                             48           6.0625%;  _
                             49           6.25%;  _
                             50           6.395833333%;  _
                             51           6.541666667%;  _
                             52           6.6875%;  _
                             53           6.833333333%;  _
                             54           6.979166667%;  _
                             55           7.125%;  _
                             56           7.270833333%;  _
                             57           7.416666667%;  _
                             58           7.5625%;  _
                             59           7.708333333%;  _
                             60           7.854166667%;  _
                             61           8%;  _
                             62           8.0625%;  _
                             63           8.125%;  _
                             64           8.1875%;  _
                             65           8.25%;  _
                             66           8.3125%;  _
                             67           8.375%;  _
                             68           8.4375%;  _
                             69           8.5%;  _
                             70           8.5625%;  _
                             71           8.625%;  _
                             72           8.6875%;  _
                             360          8.75%;  _
_
 or;(2) the aggregate principal balance of all delinquent loans * 1_
 as a percentage of the respective collateral balance exceeds :_
 34% * the Senior Enhancement Percentage."_
        IMPACT     "If a Step Up Trigger is in effect the OC target will
change to_
 the last value before the trigger occurred if a stepdown has_
 occurred.  It has no effect if a stepdown has not occurred."  _
        EFFECTIVE_WHEN (If #StepDown2 Then TRIG_EFFECTIVE_YES Else
TRIG_EFFECTIVE_ALWAYSPASS); _
        TRIGVAL FORMULA ( min(TRIGGER("StepUp-CumLoss2","TRIGVAL"),
TRIGGER("StepUp-DlqEnh2","TRIGVAL")));
!
  DEFINE DYNAMIC STICKY #COUPON_AII4 = (OPTIMAL_INTPMT("AII4") -
COUPONCAP_SHORTFALL("AII4")) / BBAL("AII4") * 36000 /
NDAYS_ACCRUE_INT("AII4#1")
  DEFINE DYNAMIC STICKY #COUPON_AII4B = (OPTIMAL_INTPMT("AII4") -
```

```
COUPONCAP_SHORTFALL("AII4")) / BBAL("AII4") * 36000 /
NDAYS_ACCRUE_INT("AII4B#1")
  DEFINE DYNAMIC STICKY #COUPON_AII4A = (OPTIMAL_INTPMT("AII4") -
COUPONCAP_SHORTFALL("AII4")) / BBAL("AII4") * 36000 /
NDAYS_ACCRUE_INT("AII4A#1")
!
  OPTIONAL REDEMPTION:    "Cleanup_Call_1" _
                          WHEN_EXPR ( COLL_BAL(1) / #OrigCollBal1 < 10% ); _
                          TARGET GROUP 1 _
                          PRICE_P ( COLL_BAL(1) ); _
                          DISTR_P RULES "OPTR_GROUP_1"
!
  OPTIONAL REDEMPTION:    "Cleanup_Call_2" _
                          WHEN_EXPR ( COLL_BAL(2) / #OrigCollBal2 < 10% ); _
                          TARGET GROUP 2 _
                          PRICE_P ( COLL_BAL(2) ); _
                          DISTR_P RULES "OPTR_GROUP_2"
!
!
 INTEREST_SHORTFALL GROUP 1 FULL_PREPAY    Compensate Pro_rata _
                            PARTIAL_PREPAY Compensate Pro_rata _
                            LOSS           NO_Compensate SUBORDINATED ACCUM
!
 INTEREST_SHORTFALL GROUP 2 FULL_PREPAY    Compensate Pro_rata _
                            PARTIAL_PREPAY Compensate Pro_rata _
                            LOSS           NO_Compensate SUBORDINATED ACCUM
!
  TRANCHE MISCINFO
  AI1           RATING SP "AAA"
  AI2           RATING SP "AAA"
  AI3           RATING SP "AAA"
  AI4           RATING SP "AAA"
  AI5           RATING SP "AAA"
  AI6           RATING SP "AAA"
  MI1           RATING SP "AA"
  MI2           RATING SP "A"
  MI3           RATING SP "BBB"
  SBI           RATING MD "NA"
  R_1           RATING MD "NA"
  AII1          RATING SP "AAA"
  AII2          RATING SP "AAA"
  AII3          RATING SP "AAA"
  AII4          RATING SP "AAA"
  MII1          RATING SP "AA"
  MII2          RATING SP "A"
  MII3          RATING SP "A-"
  MII4          RATING SP "BBB+"
  MII5          RATING SP "BBB"
  SBII          RATING MD "NA"
  R_2           RATING MD "NA"
  COF2          RATING MD "NA"
  COF1          RATING MD "NA"
!
 DEFINE MACRO BLOCK #AI_Int =
{
-----------------------------------
        from :  CLASS ( "AI" )
         pay :  CLASS INTEREST PRO_RATA  ( "AI6"; "AAANN" )
-----------------------------------
-----------------------------------
        from :  CLASS ( "AAANN" )
         pay :  CLASS INTEREST PRO_RATA  ( "AI1"; "AI2"; "AI3"; "AI4"; "AI5" )
-----------------------------------
}
 DEFINE MACRO BLOCK #AI_InS =
{
-----------------------------------
        from :  CLASS ( "AI" )
```

```
        pay :  CLASS INTSHORT PRO_RATA  ( "AI6"; "AAANN" )
------------------------------------
------------------------------------
       from :  CLASS ( "AAANN" )
        pay :  CLASS INTSHORT PRO_RATA  ( "AI1"; "AI2"; "AI3"; "AI4"; "AI5" )
------------------------------------
}
 DEFINE MACRO BLOCK #AI_Prn =
{
------------------------------------
   calculate :  #NasCeilAI = MIN ( BBAL("AI6"), CASH_ACCOUNT("AI"), #NasFracAI
* #NasShiftAI * CASH_ACCOUNT("AI") )
------------------------------------
  subject to :  CEILING ( #NasCeilAI )
       from :  CLASS ( "AI" )
        pay :  CLASS BALANCE SEQUENTIAL ("AI6")
------------------------------------
       from :  CLASS ( "AI" )
        pay :  CLASS BALANCE SEQUENTIAL ("AAANN", "AI6")
------------------------------------
!
       from :  CLASS ( "AAANN" )
        pay :  CLASS BALANCE SEQUENTIAL ( "AI1", "AI2", "AI3", "AI4", "AI5" )
------------------------------------
!
------------------------------------
       from :  CLASS ( "AI6" )
        pay :  SEQUENTIAL ( "AI6#1" )
------------------------------------
       from :  CLASS ( "AI1" )
        pay :  SEQUENTIAL ( "AI1#1" )
------------------------------------
       from :  CLASS ( "AI2" )
        pay :  SEQUENTIAL ( "AI2#1" )
------------------------------------
       from :  CLASS ( "AI3" )
        pay :  SEQUENTIAL ( "AI3#1" )
------------------------------------
       from :  CLASS ( "AI4" )
        pay :  SEQUENTIAL ( "AI4#1" )
------------------------------------
       from :  CLASS ( "AI5" )
        pay :  SEQUENTIAL ( "AI5#1" )
------------------------------------
}
 DEFINE MACRO BLOCK #MI1_Prn =
{
------------------------------------
       from :  CLASS ( "MI1" )
        pay :  SEQUENTIAL ( "MI1#1" )
------------------------------------
}
 DEFINE MACRO BLOCK #MI2_Prn =
{
------------------------------------
       from :  CLASS ( "MI2" )
        pay :  SEQUENTIAL ( "MI2#1" )
------------------------------------
}
 DEFINE MACRO BLOCK #MI3_Prn =
{
------------------------------------
       from :  CLASS ( "MI3" )
        pay :  SEQUENTIAL ( "MI3#1" )
------------------------------------
}
 DEFINE MACRO BLOCK #AII_Int =
{
```

```
-----------------------------------
        from :  CLASS ( "AII" )
         pay :  CLASS INTEREST PRO_RATA  ( "AIIA"; "AIIB" )
-----------------------------------
-----------------------------------
        from :  CLASS ( "AIIA" )
         pay :  CLASS INTEREST PRO_RATA  ( "AII2"; "AII4B" )
-----------------------------------
-----------------------------------
-----------------------------------
        from :  CLASS ( "AIIB" )
         pay :  CLASS INTEREST PRO_RATA  ( "AII1"; "AII3"; "AII4A" )
-----------------------------------
-----------------------------------
        from :  CLASS ( "AII4B" ; "AII4A" )
         pay :  CLASS INTEREST PRO_RATA  ( "AII4" )
-----------------------------------
}
 DEFINE MACRO BLOCK #AII_InS =
{
-----------------------------------
        from :  CLASS ( "AII" )
         pay :  CLASS INTSHORT PRO_RATA  ( "AIIA"; "AIIB" )
-----------------------------------
-----------------------------------
        from :  CLASS ( "AIIA" )
         pay :  CLASS INTSHORT PRO_RATA  ( "AII2"; "AII4B" )
-----------------------------------
-----------------------------------
-----------------------------------
        from :  CLASS ( "AIIB" )
         pay :  CLASS INTSHORT PRO_RATA  ( "AII1"; "AII3"; "AII4A" )
-----------------------------------
-----------------------------------
        from :  CLASS ( "AII4B" ; "AII4A" )
         pay :  CLASS INTSHORT PRO_RATA  ( "AII4" )
-----------------------------------
}
 DEFINE MACRO BLOCK #AII_Prn =
{
-----------------------------------
        from :  CLASS ( "AII" )
         pay :  CLASS BALANCE PRO_RATA ( "AIIA" ; "AIIB" )
-----------------------------------
!
        from :  CLASS ( "AIIA" )
         pay :  CLASS BALANCE SEQUENTIAL ( "AII2", "AII4B" )
-----------------------------------
!
  subject to :  CEILING ( (BBAL("AII4B#1")-BBAL("AII4B")) )
         pay :  SEQUENTIAL ( "AII4B#1")
-----------------------------------
!
        from :  CLASS ( "AIIB" )
         pay :  CLASS BALANCE SEQUENTIAL ( "AII1", "AII3", "AII4A" )
-----------------------------------
!
  subject to :  CEILING ( (BBAL("AII4A#1")-BBAL("AII4A")) )
         pay :  SEQUENTIAL ( "AII4A#1")
-----------------------------------
-----------------------------------
   calculate :  #P_AII4 = PRINCPMT("AII4B#1","AII4A#1")
-----------------------------------
        from :  SUBACCOUNT ( #P_AII4, CLASS "AII4B" )
         pay :  CLASS BALANCE  SEQUENTIAL ( "AII4" )
-----------------------------------
        from :  SUBACCOUNT ( #P_AII4, CLASS "AII4A" )
         pay :  CLASS BALANCE  SEQUENTIAL ( "AII4" )
```

```
-----------------------------------
   calculate :  #P_AII4 = PRINCPMT("AII4B#1","AII4A#1")
-----------------------------------
        from :  CLASS ( "AII4B" ; "AII4A" )
         pay :  AS_INTEREST ("R_1#1")
-----------------------------------
!
-----------------------------------
        from :  CLASS ( "AII2" )
         pay :  SEQUENTIAL ( "AII2#1" )
-----------------------------------
        from :  CLASS ( "AII4" )
         pay :  SEQUENTIAL ( "AII4#1" )
-----------------------------------
        from :  CLASS ( "AII1" )
         pay :  SEQUENTIAL ( "AII1#1" )
-----------------------------------
        from :  CLASS ( "AII3" )
         pay :  SEQUENTIAL ( "AII3#1" )
-----------------------------------
}
 DEFINE MACRO BLOCK #MII1_Prn =
{
-----------------------------------
        from :  CLASS ( "MII1" )
         pay :  SEQUENTIAL ( "MII1#1" )
-----------------------------------
}
 DEFINE MACRO BLOCK #MII2_Prn =
{
-----------------------------------
        from :  CLASS ( "MII2" )
         pay :  SEQUENTIAL ( "MII2#1" )
-----------------------------------
}
 DEFINE MACRO BLOCK #MII3_Prn =
{
-----------------------------------
        from :  CLASS ( "MII3" )
         pay :  SEQUENTIAL ( "MII3#1" )
-----------------------------------
}
 DEFINE MACRO BLOCK #MII4_Prn =
{
-----------------------------------
        from :  CLASS ( "MII4" )
         pay :  SEQUENTIAL ( "MII4#1" )
-----------------------------------
}
 DEFINE MACRO BLOCK #MII5_Prn =
{
-----------------------------------
        from :  CLASS ( "MII5" )
         pay :  SEQUENTIAL ( "MII5#1" )
-----------------------------------
}
!
!
 CMO Block Payment Rules
-----------------------------------
   calculate :  #BegBondBal1    =
BBAL("AI6#1","AI1#1","AI2#1","AI3#1","AI4#1","AI5#1","MI1#1","MI2#1","MI3#1")
   calculate :  #BegBondBal2    =
BBAL("AII2#1","AII4B#1","AII1#1","AII3#1","AII4A#1","MII1#1","MII2#1","MII3#1"
,"MII4#1","MII5#1")
   calculate :  #Princ1              = COLL_P(1)
!
   calculate :  #Interest1           = COLL_I(1)
```

```
!
   calculate :  #Princ2              = COLL_P(2)
!
   calculate :  #Interest2           = COLL_I(2)
!
   calculate :  #PrevSpecOC1         = #SpecOCTarg1
!
   calculate :  #CurrentOC1          = MAX( 0, COLL_PREV_BAL(1) -
BBAL("AI6#1", "AI1#1", "AI2#1", "AI3#1", "AI4#1", "AI5#1", "MI1#1", "MI2#1",
"MI3#1"))
!
   calculate :  #XSSpread1           = MAX( 0, #Interest1 -
OPTIMAL_INTPMT("GRP1") - INTSHORT_ACCUM("GRP1") + COUPONCAP_SHORTFALL("GRP1")
)
!
   calculate :  #FloorOCTotal1       = #FloorOCTarg1
!
   calculate :  #StepOCTarg1         = COLL_BAL(1) * #StepOCFrac1
!
   calculate :  #StepDownDatePass1   = CURMONTH GE #StepDownDate1
!
   calculate :  #StepDownBal1        = COLL_BAL(1) < #OrigCollBal1 * 50%
!
   calculate :  #StepDown1           = #StepDown1 OR ( #StepDownDatePass1 AND
#StepDownBal1 )
!
   calculate :  #ReqPerc1            = 48% * (COLL_PREV_BAL(1) - BBAL("AI") )
/ COLL_PREV_BAL(1)
!
   calculate :  #TrigEnhFrac1        = 1 * AVG_COLL("RATE",-1,2,3,1)
!
   calculate :  #CumLossShft1        = LOOKUP_TBL( "STEP",  CURMONTH     ,
"OC_CUMLOSS1", "MONTH", "OC_CUMLOSS_FRAC1" )
   calculate :  #TrigCumLossFrac1    = DELINQ_LOSS_ACCUM(1) / #OrigCollBal1
!
   calculate :  #TrigEvent1          = TRIGGER("STEPUP_TRIGGER1")
!
   calculate :  #TrigOCTargPost1     = #PrevSpecOC1
!
   calculate :  #SpecOCTarg1         = IF #StepDown1 _
                                     THEN IF #TrigEvent1 _
                                          THEN MAX( MIN( #InitOCTarg1,
#StepOCTarg1 ) , #TrigOCTargPost1, #FloorOCTotal1 ) _
                                          ELSE MAX( MIN( #InitOCTarg1,
#StepOCTarg1 ) , #FloorOCTotal1 )  _
                                     ELSE MAX ( #InitOCTarg1, #FloorOCTotal1 )

!
   calculate :  #SpecOCTarg1         = MIN( #SpecOCTarg1, COLL_BAL(1) )
!
   calculate :  #SpecOCTarg1         = #Octval1
!
   calculate :  #OCDeficiency1       = MAX(0, #SpecOCTarg1 - #CurrentOC1)
!
   calculate :  #OCSurplus1          = MINMAX(0, #CurrentOC1 - #SpecOCTarg1,
COLL_P(1))
!
   calculate :  #PrincPmt1           = MAX(0, COLL_P(1) - #OCSurplus1)
!
   calculate :  #PrevSpecOC2         = #SpecOCTarg2
!
   calculate :  #CurrentOC2          = MAX( 0, COLL_PREV_BAL(2) -
BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1", "MII1#1", "MII2#1",
"MII3#1", "MII4#1", "MII5#1"))
!
   calculate :  #XSSpread2           = MAX( 0, #Interest2 -
OPTIMAL_INTPMT("GRP2") - INTSHORT_ACCUM("GRP2") + COUPONCAP_SHORTFALL("GRP2")
)
```

```
!
   calculate :  #FloorOCTotal2       = #FloorOCTarg2
!
   calculate :  #StepOCTarg2         = COLL_BAL(2) * #StepOCFrac2
!
   calculate :  #StepDownDatePass2   = CURMONTH GE #StepDownDate2
!
   calculate :  #StepDownBal2        = COLL_BAL(2) < #OrigCollBal2 * 50%
!
   calculate :  #StepDown2           = #StepDown2 OR ( #StepDownDatePass2 AND
#StepDownBal2 )
!
   calculate :  #ReqPerc2            = 38% * (COLL_PREV_BAL(2) - BBAL("AII") )
/ COLL_PREV_BAL(2)
!
   calculate :  #TrigEnhFrac2        = 1 * AVG_COLL("RATE",-1,2,3,2)
!
   calculate :  #CumLossShft2        = LOOKUP_TBL( "STEP",  CURMONTH     ,
"OC_CUMLOSS2", "MONTH", "OC_CUMLOSS_FRAC2" )
   calculate :  #TrigCumLossFrac2    = DELINQ_LOSS_ACCUM(2) / #OrigCollBal2
!
   calculate :  #TrigEvent2          = TRIGGER("STEPUP_TRIGGER2")
!
   calculate :  #TrigOCTargPost2     = #PrevSpecOC2
!
   calculate :  #SpecOCTarg2         = IF #StepDown2 _
                                       THEN IF #TrigEvent2 _
                                            THEN MAX( MIN( #InitOCTarg2,
#StepOCTarg2 ) , #TrigOCTargPost2, #FloorOCTotal2 ) _
                                            ELSE MAX( MIN( #InitOCTarg2,
#StepOCTarg2 ) , #FloorOCTotal2 )  _
                                       ELSE MAX ( #InitOCTarg2, #FloorOCTotal2 )

!
   calculate :  #SpecOCTarg2         = MIN( #SpecOCTarg2, COLL_BAL(2) )
!
   calculate :  #SpecOCTarg2         = #Octval2
!
   calculate :  #OCDeficiency2       = MAX(0, #SpecOCTarg2 - #CurrentOC2)
!
   calculate :  #OCSurplus2          = MINMAX(0, #CurrentOC2 - #SpecOCTarg2,
COLL_P(2))
!
   calculate :  #PrincPmt2           = MAX(0, COLL_P(2) - #OCSurplus2)
!
!
   calculate :  #NasShiftAI          = LOOKUP_TBL( "STEP", CURMONTH , "NASAI",
"MONTH", "NAS_FRACAI" )
!
   calculate :  #NasFracAI           = BBAL("AI6")/ BBAL("AI6", "AAANN")
!
   calculate :  #XSIntRem1           = MAX( 0, #Interest1 -
OPTIMAL_INTPMT("GRP1") - INTSHORT_ACCUM("GRP1") + #OCSurplus1 +
COUPONCAP_SHORTFALL("GRP1")+ IF BBAL("AII2", "AII4B", "AII1", "AII3", "AII4A",
"MII1", "MII2", "MII3", "MII4", "MII5") EQ 0 THEN #Princ2 ELSE 0 )
!
   calculate :  #XSIntRem2           = MAX( 0, #Interest2 -
OPTIMAL_INTPMT("GRP2") - INTSHORT_ACCUM("GRP2") + #OCSurplus2 +
COUPONCAP_SHORTFALL("GRP2") + IF BBAL("AI6", "AI1", "AI2", "AI3", "AI4",
"AI5", "MI1", "MI2", "MI3") EQ 0 THEN #Princ1 ELSE 0 )
!
   calculate :  #SubDefic1           = DELINQ_NET_LOSS(1)
!
   calculate :  #AddPrinc1           = MIN( #XSIntRem1, #SubDefic1 )
   calculate :  #XSIntRem1           = MAX( 0, #XSIntRem1 - #AddPrinc1 )
!
   calculate :  #SubDefic2           = DELINQ_NET_LOSS(2)
!
```

```
   calculate :  #AddPrinc2           = MIN( #XSIntRem2, #SubDefic2 )
   calculate :  #XSIntRem2           = MAX( 0, #XSIntRem2 - #AddPrinc2 )
!
   calculate :  #MoreDefic1          = MAX( 0, #SubDefic1 - #AddPrinc1 )
!
   calculate :  #MoreDefic2          = MAX( 0, #SubDefic2 - #AddPrinc2 )
!
   calculate :  #AddPrinc1C          = MIN(( #XSIntRem2 ) * ( #MoreDefic1 / (
#MoreDefic1 + #MoreDefic2)), #MoreDefic1 )
   calculate :  #AddPrinc2C          = MIN(( #XSIntRem1 ) * ( #MoreDefic2 / (
#MoreDefic1 + #MoreDefic2)), #MoreDefic2 )
!
   calculate :  #XSIntOut1           = MIN(( #AddPrinc2C ) * ( #XSIntRem1 / (
#XSIntRem1 + #XSIntRem2)), #XSIntRem1 )
   calculate :  #XSIntOut2           = MIN(( #AddPrinc1C ) * ( #XSIntRem2 / (
#XSIntRem1 + #XSIntRem2)), #XSIntRem2 )
!
   calculate :  #XSIntRem1           = MAX( 0, #XSIntRem1 - #XSIntOut1 )
   calculate :  #XSIntRem2           = MAX( 0, #XSIntRem2 - #XSIntOut2 )
!
   calculate :  #XtraPDA1            = MIN( #OCDeficiency1, #XSIntRem1 )
   calculate :  #XSIntRem1           = MAX( 0, #XSIntRem1 - #XtraPDA1 )
!
   calculate :  #XtraPDA2            = MIN( #OCDeficiency2, #XSIntRem2 )
   calculate :  #XSIntRem2           = MAX( 0, #XSIntRem2 - #XtraPDA2 )
!
   calculate :  #NeedPDA1C           = MAX( 0, #OCDeficiency1- #XtraPDA1 )
   calculate :  #NeedPDA2C           = MAX( 0, #OCDeficiency2- #XtraPDA2 )
!
   calculate :  #XtraPDA1C           = MIN( #NeedPDA1C , (#XSIntRem2 ) * (
#NeedPDA1C / ( #NeedPDA1C + #NeedPDA2C)) )
   calculate :  #XtraPDA2C           = MIN( #NeedPDA2C , (#XSIntRem1 ) * (
#NeedPDA2C / ( #NeedPDA2C + #NeedPDA1C)) )
!
   calculate :  #XSIntOut1           = MIN(( #XtraPDA2C ) * ( #XSIntRem1 / (
#XSIntRem1 + #XSIntRem2)), #XSIntRem1 )
   calculate :  #XSIntOut2           = MIN(( #XtraPDA1C ) * ( #XSIntRem2 / (
#XSIntRem1 + #XSIntRem2)), #XSIntRem2 )
!
   calculate :  #XSIntRem1           = MAX( 0, #XSIntRem1 - #XSIntOut1 )
   calculate :  #XSIntRem2           = MAX( 0, #XSIntRem2 - #XSIntOut2 )
!
!
   calculate :  #DistribAmt1         = #PrincPmt1 + #AddPrinc1 + #XtraPDA1 +
#AddPrinc1C + #XtraPDA1C
   calculate :  #DistribAmt2         = #PrincPmt2 + #AddPrinc2 + #XtraPDA2 +
#AddPrinc2C + #XtraPDA2C
!
   calculate :  #ClassAIPDA          = IF (#TrigEvent1 OR (#StepDown1 EQ 0.0))
_
                                       THEN #DistribAmt1 _
                                       ELSE BBAL("AI6", "AI1", "AI2", "AI3",
"AI4", "AI5") _
                                       - MIN(COLL_BAL(1) - #FloorOCTotal1,
#AITargPct * COLL_BAL(1))
   calculate :  #ClassAIPDA          = MAX( 0.0, MIN(BBAL("AI6", "AI1", "AI2",
"AI3", "AI4", "AI5"), #ClassAIPDA ))
   calculate :  #ClassAIPDA          = MAX( 0, MIN( #ClassAIPDA, #DistribAmt1
) )
!
!
   calculate :  #ClassMI1PDA         = IF (#TrigEvent1 OR (#StepDown1 EQ 0.0))
_
                                       THEN #DistribAmt1 - #ClassAIPDA _
                                       ELSE BBAL("AI6", "AI1", "AI2", "AI3",
"AI4", "AI5", "MI1") - #ClassAIPDA _
                                       - MIN(COLL_BAL(1) - #FloorOCTotal1,
#MI1TargPct * COLL_BAL(1))
```

```
   calculate :  #ClassMI1PDA         = MAX( 0.0, MIN(BBAL("MI1"), #ClassMI1PDA
))
   calculate :  #ClassMI1PDA         = MAX( 0, MIN( #ClassMI1PDA, #DistribAmt1
- #ClassAIPDA ) )
!
!
   calculate :  #ClassMI2PDA         = IF (#TrigEvent1 OR (#StepDown1 EQ 0.0))
_
                                        THEN #DistribAmt1 - #ClassAIPDA -
#ClassMI1PDA _
                                        ELSE BBAL("AI6", "AI1", "AI2", "AI3",
"AI4", "AI5", "MI1", "MI2") - #ClassAIPDA - #ClassMI1PDA _
                                        - MIN(COLL_BAL(1) - #FloorOCTotal1,
#MI2TargPct * COLL_BAL(1))
   calculate :  #ClassMI2PDA         = MAX( 0.0, MIN(BBAL("MI2"), #ClassMI2PDA
))
   calculate :  #ClassMI2PDA         = MAX( 0, MIN( #ClassMI2PDA, #DistribAmt1
- #ClassAIPDA - #ClassMI1PDA ) )
!
!
   calculate :  #ClassMI3PDA         = IF (#TrigEvent1 OR (#StepDown1 EQ 0.0))
_
                                        THEN #DistribAmt1 - #ClassAIPDA -
#ClassMI1PDA - #ClassMI2PDA _
                                        ELSE BBAL("AI6", "AI1", "AI2", "AI3",
"AI4", "AI5", "MI1", "MI2", "MI3") - #ClassAIPDA - #ClassMI1PDA - #ClassMI2PDA
_
                                        - MIN(COLL_BAL(1) - #FloorOCTotal1,
#MI3TargPct * COLL_BAL(1))
   calculate :  #ClassMI3PDA         = MAX( 0.0, MIN(BBAL("MI3"), #ClassMI3PDA
))
   calculate :  #ClassMI3PDA         = MAX( 0, MIN( #ClassMI3PDA, #DistribAmt1
- #ClassAIPDA - #ClassMI1PDA - #ClassMI2PDA ) )
!
!
   calculate :  #ClassAIIPDA         = IF (#TrigEvent2 OR (#StepDown2 EQ 0.0))
_
                                        THEN #DistribAmt2 _
                                        ELSE BBAL("AII2", "AII4B", "AII1",
"AII3", "AII4A") _
                                        - MIN(COLL_BAL(2) - #FloorOCTotal2,
#AIITargPct * COLL_BAL(2))
   calculate :  #ClassAIIPDA         = MAX( 0.0, MIN(BBAL("AII2", "AII4B",
"AII1", "AII3", "AII4A"), #ClassAIIPDA ))
   calculate :  #ClassAIIPDA         = MAX( 0, MIN( #ClassAIIPDA, #DistribAmt2
) )
!
!
   calculate :  #ClassMII1PDA        = IF (#TrigEvent2 OR (#StepDown2 EQ 0.0))
_
                                        THEN #DistribAmt2 - #ClassAIIPDA _
                                        ELSE BBAL("AII2", "AII4B", "AII1",
"AII3", "AII4A", "MII1") - #ClassAIIPDA _
                                        - MIN(COLL_BAL(2) - #FloorOCTotal2,
#MII1TargPct * COLL_BAL(2))
   calculate :  #ClassMII1PDA        = MAX( 0.0, MIN(BBAL("MII1"),
#ClassMII1PDA ))
   calculate :  #ClassMII1PDA        = MAX( 0, MIN( #ClassMII1PDA,
#DistribAmt2 - #ClassAIIPDA ) )
!
!
   calculate :  #ClassMII2PDA        = IF (#TrigEvent2 OR (#StepDown2 EQ 0.0))
_
                                        THEN #DistribAmt2 - #ClassAIIPDA -
#ClassMII1PDA _
                                        ELSE BBAL("AII2", "AII4B", "AII1",
"AII3", "AII4A", "MII1", "MII2") - #ClassAIIPDA - #ClassMII1PDA _
                                        - MIN(COLL_BAL(2) - #FloorOCTotal2,
```

```
#MII2TargPct * COLL_BAL(2))
   calculate :  #ClassMII2PDA        = MAX( 0.0, MIN(BBAL("MII2"),
#ClassMII2PDA ))
   calculate :  #ClassMII2PDA        = MAX( 0, MIN( #ClassMII2PDA,
#DistribAmt2 - #ClassAIIPDA - #ClassMII1PDA ) )
!
!
   calculate :  #ClassMII3PDA        = IF (#TrigEvent2 OR (#StepDown2 EQ 0.0))
_
                                        THEN #DistribAmt2 - #ClassAIIPDA -
#ClassMII1PDA - #ClassMII2PDA _
                                        ELSE BBAL("AII2", "AII4B", "AII1",
"AII3", "AII4A", "MII1", "MII2", "MII3") - #ClassAIIPDA - #ClassMII1PDA -
#ClassMII2PDA _
                                        - MIN(COLL_BAL(2) - #FloorOCTotal2,
#MII3TargPct * COLL_BAL(2))
   calculate :  #ClassMII3PDA        = MAX( 0.0, MIN(BBAL("MII3"),
#ClassMII3PDA ))
   calculate :  #ClassMII3PDA        = MAX( 0, MIN( #ClassMII3PDA,
#DistribAmt2 - #ClassAIIPDA - #ClassMII1PDA - #ClassMII2PDA ) )
!
!
   calculate :  #ClassMII4PDA        = IF (#TrigEvent2 OR (#StepDown2 EQ 0.0))
_
                                        THEN #DistribAmt2 - #ClassAIIPDA -
#ClassMII1PDA - #ClassMII2PDA - #ClassMII3PDA _
                                        ELSE BBAL("AII2", "AII4B", "AII1",
"AII3", "AII4A", "MII1", "MII2", "MII3", "MII4") - #ClassAIIPDA -
#ClassMII1PDA - #ClassMII2PDA - #ClassMII3PDA _
                                        - MIN(COLL_BAL(2) - #FloorOCTotal2,
#MII4TargPct * COLL_BAL(2))
   calculate :  #ClassMII4PDA        = MAX( 0.0, MIN(BBAL("MII4"),
#ClassMII4PDA ))
   calculate :  #ClassMII4PDA        = MAX( 0, MIN( #ClassMII4PDA,
#DistribAmt2 - #ClassAIIPDA - #ClassMII1PDA - #ClassMII2PDA - #ClassMII3PDA )
)
!
!
   calculate :  #ClassMII5PDA        = IF (#TrigEvent2 OR (#StepDown2 EQ 0.0))
_
                                        THEN #DistribAmt2 - #ClassAIIPDA -
#ClassMII1PDA - #ClassMII2PDA - #ClassMII3PDA - #ClassMII4PDA _
                                        ELSE BBAL("AII2", "AII4B", "AII1",
"AII3", "AII4A", "MII1", "MII2", "MII3", "MII4", "MII5") - #ClassAIIPDA -
#ClassMII1PDA - #ClassMII2PDA - #ClassMII3PDA - #ClassMII4PDA _
                                        - MIN(COLL_BAL(2) - #FloorOCTotal2,
#MII5TargPct * COLL_BAL(2))
   calculate :  #ClassMII5PDA        = MAX( 0.0, MIN(BBAL("MII5"),
#ClassMII5PDA ))
   calculate :  #ClassMII5PDA        = MAX( 0, MIN( #ClassMII5PDA,
#DistribAmt2 - #ClassAIIPDA - #ClassMII1PDA - #ClassMII2PDA - #ClassMII3PDA -
#ClassMII4PDA ) )
!
!
  calculate :  "AI" _
 NO_CHECK  CUSTOM GROUP 1   AMOUNT        = #ClassAIPDA
!
  calculate :  "MI1" _
 NO_CHECK  CUSTOM GROUP 1   AMOUNT        = #ClassMI1PDA
!
  calculate :  "MI2" _
 NO_CHECK  CUSTOM GROUP 1   AMOUNT        = #ClassMI2PDA
!
  calculate :  "MI3" _
 NO_CHECK  CUSTOM GROUP 1   AMOUNT        = #ClassMI3PDA
!
  calculate :  "AII" _
 NO_CHECK  CUSTOM GROUP 2   AMOUNT        = #ClassAIIPDA
```

```
!
  calculate :  "MII1" _
 NO_CHECK  CUSTOM GROUP 2   AMOUNT        = #ClassMII1PDA
!
  calculate :  "MII2" _
 NO_CHECK  CUSTOM GROUP 2   AMOUNT        = #ClassMII2PDA
!
  calculate :  "MII3" _
 NO_CHECK  CUSTOM GROUP 2   AMOUNT        = #ClassMII3PDA
!
  calculate :  "MII4" _
 NO_CHECK  CUSTOM GROUP 2   AMOUNT        = #ClassMII4PDA
!
  calculate :  "MII5" _
 NO_CHECK  CUSTOM GROUP 2   AMOUNT        = #ClassMII5PDA
!
-----------------------------------
!
-----------------------------------  Pay Interest to AI  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "AI" )
-----------------------------------
  {#AI_Int}
-----------------------------------
!
-----------------------------------  Pay Interest Shortfall to AI  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "AI" )
-----------------------------------
  {#AI_InS}
-----------------------------------
!
-----------------------------------  Pay Interest to MI1  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MI1" )
-----------------------------------
!
-----------------------------------  Pay Interest Shortfall to MI1  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MI1" )
-----------------------------------
!
-----------------------------------  Pay Interest to MI2  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MI2" )
-----------------------------------
!
-----------------------------------  Pay Interest Shortfall to MI2  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MI2" )
-----------------------------------
!
-----------------------------------  Pay Interest to MI3  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MI3" )
-----------------------------------
!
-----------------------------------  Pay Interest Shortfall to MI3  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MI3" )
-----------------------------------
!
-----------------------------------  Pay Principal to AI  From Cash Account
        from :  CLASS ( "GRP1" )
```

```
        pay :  CLASS PRINCIPAL SEQUENTIAL  ( "AI" )
-----------------------------------
  {#AI_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MI1  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MI1" )
-----------------------------------
  {#MI1_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MI2  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MI2" )
-----------------------------------
  {#MI2_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MI3  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MI3" )
-----------------------------------
  {#MI3_Prn}
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to AI  From
Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "AI6"; "AI1"; "AI4"; "AI5" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MI1
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MI1" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MI2
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MI2" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MI3
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MI3" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to AI  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "AI" )
-----------------------------------
        from :  CLASS ( "AI" )
         pay :  CLASS PRINCSHORT_LOSS PRO_RATA    (
"AI6";"AI1";"AI2";"AI3";"AI4";"AI5" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MI1  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MI1" )
```

```
-----------------------------------
!
----------------------------------- Pay Writedown Loss to MI2  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MI2" )
-----------------------------------
!
----------------------------------- Pay Writedown Loss to MI3  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MI3" )
-----------------------------------
!
----------------------------------- Pay Interest to AII  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "AII" )
-----------------------------------
  {#AII_Int}
-----------------------------------
!
----------------------------------- Pay Interest Shortfall to AII  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "AII" )
-----------------------------------
  {#AII_InS}
-----------------------------------
!
----------------------------------- Pay Interest to MII1  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII1" )
-----------------------------------
!
----------------------------------- Pay Interest Shortfall to MII1  From
Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII1" )
-----------------------------------
!
----------------------------------- Pay Interest to MII2  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII2" )
-----------------------------------
!
----------------------------------- Pay Interest Shortfall to MII2  From
Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII2" )
-----------------------------------
!
----------------------------------- Pay Interest to MII3  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII3" )
-----------------------------------
!
----------------------------------- Pay Interest Shortfall to MII3  From
Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII3" )
-----------------------------------
!
----------------------------------- Pay Interest to MII4  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII4" )
-----------------------------------
!
----------------------------------- Pay Interest Shortfall to MII4  From
```

```
Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII4" )
-----------------------------------
!
-----------------------------------  Pay Interest to MII5  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII5" )
-----------------------------------
!
-----------------------------------  Pay Interest Shortfall to MII5  From
Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII5" )
-----------------------------------
!
-----------------------------------  Pay Principal to AII  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "AII" )
-----------------------------------
  {#AII_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII1  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII1" )
-----------------------------------
  {#MII1_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII2  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII2" )
-----------------------------------
  {#MII2_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII3  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII3" )
-----------------------------------
  {#MII3_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII4  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII4" )
-----------------------------------
  {#MII4_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII5  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII5" )
-----------------------------------
  {#MII5_Prn}
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to AII
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "AII2"; "AII4B"; "AII1";
"AII3"; "AII4A" )
-----------------------------------
        from :  CLASS ( "AII4B" )
  subject to :  CEILING (( COUPONCAP_SHORTFALL("AII4B") ))
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "AII4" )
-----------------------------------
```

```
        from :  CLASS ( "AII4A" )
  subject to :  CEILING (( COUPONCAP_SHORTFALL("AII4A") ))
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "AII4" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII1
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII1" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII2
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII2" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII3
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII3" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII4
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII4" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII5
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII5" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to AII  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "AII" )
-----------------------------------
        from :  CLASS ( "AII" )
         pay :  CLASS PRINCSHORT_LOSS PRO_RATA    (
"AII2";"AII4";"AII1";"AII3" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MII1  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII1" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MII2  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII2" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MII3  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII3" )
-----------------------------------
```

```
!
------------------------------------  Pay Writedown Loss to MII4  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII4" )
------------------------------------
!
------------------------------------  Pay Writedown Loss to MII5  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII5" )
------------------------------------
!
------------------------------------  Pay Interest to AII  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "AII" )
------------------------------------
  {#AII_Int}
------------------------------------
!
------------------------------------  Pay Interest Shortfall to AII  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "AII" )
------------------------------------
  {#AII_InS}
------------------------------------
!
------------------------------------  Pay Interest to MII1  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII1" )
------------------------------------
!
------------------------------------  Pay Interest Shortfall to MII1  From
Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII1" )
------------------------------------
!
------------------------------------  Pay Interest to MII2  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII2" )
------------------------------------
!
------------------------------------  Pay Interest Shortfall to MII2  From
Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII2" )
------------------------------------
!
------------------------------------  Pay Interest to MII3  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII3" )
------------------------------------
!
------------------------------------  Pay Interest Shortfall to MII3  From
Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII3" )
------------------------------------
!
------------------------------------  Pay Interest to MII4  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII4" )
------------------------------------
!
------------------------------------  Pay Interest Shortfall to MII4  From
Cash Account
```

```
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII4" )
-----------------------------------
!
-----------------------------------  Pay Interest to MII5  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MII5" )
-----------------------------------
!
-----------------------------------  Pay Interest Shortfall to MII5  From
Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MII5" )
-----------------------------------
!
-----------------------------------  Pay Principal to AII  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "AII" )
-----------------------------------
  {#AII_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII1  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII1" )
-----------------------------------
  {#MII1_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII2  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII2" )
-----------------------------------
  {#MII2_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII3  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII3" )
-----------------------------------
  {#MII3_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII4  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII4" )
-----------------------------------
  {#MII4_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MII5  From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MII5" )
-----------------------------------
  {#MII5_Prn}
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to AII
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "AII2"; "AII4B"; "AII1";
"AII3"; "AII4A" )
-----------------------------------
        from :  CLASS ( "AII4B" )
  subject to :  CEILING (( COUPONCAP_SHORTFALL("AII4B") ))
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "AII4" )
-----------------------------------
        from :  CLASS ( "AII4A" )
```

```
  subject to :  CEILING (( COUPONCAP_SHORTFALL("AII4A") ))
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "AII4" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII1
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII1" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII2
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII2" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII3
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII3" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII4
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII4" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MII5
From Cash Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MII5" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to AII  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "AII" )
-----------------------------------
        from :  CLASS ( "AII" )
         pay :  CLASS PRINCSHORT_LOSS PRO_RATA    (
"AII2";"AII4";"AII1";"AII3" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MII1  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII1" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MII2  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII2" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MII3  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII3" )
-----------------------------------
!
```

```
------------------------------------  Pay Writedown Loss to MII4  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII4" )
------------------------------------
!
------------------------------------  Pay Writedown Loss to MII5  From Cash
Account
        from :  CLASS ( "GRP1" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MII5" )
------------------------------------
!
------------------------------------  Pay Interest to AI  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "AI" )
------------------------------------
  {#AI_Int}
------------------------------------
!
------------------------------------  Pay Interest Shortfall to AI  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "AI" )
------------------------------------
  {#AI_InS}
------------------------------------
!
------------------------------------  Pay Interest to MI1  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MI1" )
------------------------------------
!
------------------------------------  Pay Interest Shortfall to MI1  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MI1" )
------------------------------------
!
------------------------------------  Pay Interest to MI2  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MI2" )
------------------------------------
!
------------------------------------  Pay Interest Shortfall to MI2  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MI2" )
------------------------------------
!
------------------------------------  Pay Interest to MI3  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTEREST  SEQUENTIAL  ( "MI3" )
------------------------------------
!
------------------------------------  Pay Interest Shortfall to MI3  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS INTSHORT  SEQUENTIAL  ( "MI3" )
------------------------------------
!
------------------------------------  Pay Principal to AI  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "AI" )
------------------------------------
  {#AI_Prn}
------------------------------------
!
------------------------------------  Pay Principal to MI1  From Cash Account
```

```
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MI1" )
-----------------------------------
  {#MI1_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MI2  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MI2" )
-----------------------------------
  {#MI2_Prn}
-----------------------------------
!
-----------------------------------  Pay Principal to MI3  From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCIPAL SEQUENTIAL  ( "MI3" )
-----------------------------------
  {#MI3_Prn}
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to AI  From
Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "AI6"; "AI1"; "AI4"; "AI5" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MI1
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MI1" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MI2
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MI2" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Payback Basis Risk Shortfall to MI3
From Cash Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS COUPONCAP_SHORT PRO_RATA ( "MI3" )
-----------------------------------
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to AI  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "AI" )
-----------------------------------
        from :  CLASS ( "AI" )
         pay :  CLASS PRINCSHORT_LOSS PRO_RATA    (
"AI6";"AI1";"AI2";"AI3";"AI4";"AI5" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MI1  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MI1" )
-----------------------------------
!
-----------------------------------  Pay Writedown Loss to MI2  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MI2" )
```

```
------------------------------------
!
------------------------------------  Pay Writedown Loss to MI3  From Cash
Account
        from :  CLASS ( "GRP2" )
         pay :  CLASS PRINCSHORT_LOSS SEQUENTIAL  ( "MI3" )
------------------------------------
!
        from :  CLASS ( "GRP1" )
         pay :  AS_INTEREST ("R_1#1")
------------------------------------
        from :  CLASS ( "GRP2" )
         pay :  AS_INTEREST ("R_2#1")
------------------------------------
   calculate :  #CurrBondBal1    =
BBAL("AI6#1","AI1#1","AI2#1","AI3#1","AI4#1","AI5#1","MI1#1","MI2#1","MI3#1")
   calculate :  #CurrBondBal2    =
BBAL("AII2#1","AII4B#1","AII1#1","AII3#1","AII4A#1","MII1#1","MII2#1","MII3#1"
,"MII4#1","MII5#1")
------------------------------------
   calculate :  #PrinPaid1 = #BegBondBal1 - #CurrBondBal1
   calculate :  #PrinPaid2 = #BegBondBal1 - #CurrBondBal1
------------------------------------
!!        when :  IS_TRUE (CURMONTH GT 1)
         pay :  INCREMENT ( BALANCE "SBI#1"  , BY #PrinPaid1 )
         pay :  INCREMENT ( BALANCE "SBII#1" , BY #PrinPaid2 )
------------------------------------
   calculate :  #WriteDown1 = MAX(0.0,
BBAL("AI6#1","AI1#1","AI2#1","AI3#1","AI4#1","AI5#1","MI1#1","MI2#1","MI3#1","
SBI#1") - COLL_BAL(1))
   calculate :  #WriteDown2 = MAX(0.0,
BBAL("AII2#1","AII4B#1","AII1#1","AII3#1","AII4A#1","MII1#1","MII2#1","MII3#1"
,"MII4#1","MII5#1","SBII#1") - COLL_BAL(2))
------------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "SBI#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "SBII#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "SBI#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "SBII#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "MI3#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "MI2#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "MI1#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN PRO_RATA ( "AI6#1"; "AI1#1"; "AI2#1"; "AI3#1";
"AI4#1"; "AI5#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII5#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII4#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII3#1" )
------------------------------------
```

```
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII2#1" )
------------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII1#1" )
------------------------------------
   calculate :  #WD_AII2 = MIN(BBAL("AII2#1"), #Writedown2 * BBAL("AII2#1") /
BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #WD_AII4B = MIN(BBAL("AII4B#1"), #Writedown2 * BBAL("AII4B#1")
/ BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #WD_AII1 = MIN(BBAL("AII1#1"), #Writedown2 * BBAL("AII1#1") /
BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #WD_AII3 = MIN(BBAL("AII3#1"), #Writedown2 * BBAL("AII3#1") /
BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #WD_AII4A = MIN(BBAL("AII4A#1"), #Writedown2 * BBAL("AII4A#1")
/ BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #Writedown2 = #Writedown2 - #WD_AII2 - #WD_AII4B - #WD_AII1 -
#WD_AII3 - #WD_AII4A
------------------------------------
        from :  SUBACCOUNT ( #WD_AII2 )
         pay :  WRITEDOWN SEQUENTIAL ( "AII2#1" )
------------------------------------
  subject to :  CEILING ( #WD_AII4B )
         pay :  DECREMENT( BALANCE "AII4B#1", BY #WD_AII4B )
------------------------------------
        from :  SUBACCOUNT ( #WD_AII1 )
         pay :  WRITEDOWN SEQUENTIAL ( "AII1#1" )
------------------------------------
        from :  SUBACCOUNT ( #WD_AII3 )
         pay :  WRITEDOWN SEQUENTIAL ( "AII3#1" )
------------------------------------
  subject to :  CEILING ( #WD_AII4A )
         pay :  DECREMENT( BALANCE "AII4A#1", BY #WD_AII4A )
------------------------------------
   calculate : #BondBal1    =
BBAL("AI6#1","AI1#1","AI2#1","AI3#1","AI4#1","AI5#1","MI1#1","MI2#1","MI3#1","
SBI#1")
   calculate : #OC1         = MAX( 0, COLL_BAL(1) - #BondBal1 )
   calculate : #IncrOC1     = MAX( 0, #OC1 - BBAL( "SBI#1" ) )
   calculate : #DecrOC1     = MAX( 0, BBAL( "SBI#1" ) - #OC1)
------------------------------------
   calculate : #BondBal2    =
BBAL("AII2#1","AII4B#1","AII1#1","AII3#1","AII4A#1","MII1#1","MII2#1","MII3#1"
,"MII4#1","MII5#1","SBII#1")
   calculate : #OC2         = MAX( 0, COLL_BAL(2) - #BondBal2 )
   calculate : #IncrOC2     = MAX( 0, #OC2 - BBAL( "SBII#1" ) )
   calculate : #DecrOC2     = MAX( 0, BBAL( "SBII#1" ) - #OC2)
------------------------------------
         pay :  INCREMENT ( BALANCE "SBI#1" , BY #IncrOC1 )
!!         pay :  INCREMENT ( BALANCE "SBI#1" , BY #DecrOC1 )
------------------------------------
         pay :  INCREMENT ( BALANCE "SBII#1" , BY #IncrOC2 )
!!         pay :  INCREMENT ( BALANCE "SBII#1" , BY #DecrOC2 )
------------------------------------
   calculate : #BondBal1    =
BBAL("AI6#1","AI1#1","AI2#1","AI3#1","AI4#1","AI5#1","MI1#1","MI2#1","MI3#1")
   calculate : #OC1         = MAX( 0, COLL_BAL(1) - #BondBal1 )
   calculate : #BondBal2    =
BBAL("AII2#1","AII4B#1","AII1#1","AII3#1","AII4A#1","MII1#1","MII2#1","MII3#1"
,"MII4#1","MII5#1")
   calculate : #OC2         = MAX( 0, COLL_BAL(2) - #BondBal2 )
   calculate : #BondBal     = #BondBal1 + #BondBal2
------------------------------------
------------------------------------
 calculate: #CallBalGrp1 = COLL_BAL(1)
 calculate: #CallBalGrp2 = COLL_BAL(2)
------------------------------------
---------------------- SECTION: "OPTR_GROUP_1"
```

```
-----------------------------------
        from :  CASH_ACCOUNT (100)
  subject to :  CEILING ( #CallBalGrp1)
         pay :  CLASS BALANCE SEQUENTIAL ( "GRP1" )
         pay :  CLASS MORE_INTEREST SEQUENTIAL ( "GRP1" )
-----------------------------------
        from :  CLASS ( "GRP1" )
         pay :  PRO_RATA ( "AI6#1"; "AI1#1"; "AI2#1"; "AI3#1"; "AI4#1";
"AI5#1" )
-----------------------------------
        from :  CLASS ( "GRP1" )
         pay :  PRO_RATA ( "MI1#1" )
-----------------------------------
        from :  CLASS ( "GRP1" )
         pay :  PRO_RATA ( "MI2#1" )
-----------------------------------
        from :  CLASS ( "GRP1" )
         pay :  PRO_RATA ( "MI3#1" )
-----------------------------------
        from :  CLASS ( "GRP1" )
         pay :  PRO_RATA ( "SBI#1" )
-----------------------------------
        from :  CLASS ( "GRP1" )
         pay :  AS_INTEREST ( "R_1#1" )
-----------------------------------
   calculate : #WriteDown1 = BBAL("GRP1")
-----------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "SBI#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "MI3#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "MI2#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN SEQUENTIAL ( "MI1#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown1 )
         pay :  WRITEDOWN PRO_RATA ( "AI6#1"; "AI1#1"; "AI2#1"; "AI3#1";
"AI4#1"; "AI5#1" )
-----------------------------------
---------------------- SECTION: "OPTR_GROUP_2"
-----------------------------------
        from :  CASH_ACCOUNT (100)
  subject to :  CEILING ( #CallBalGrp2)
         pay :  CLASS BALANCE SEQUENTIAL ( "GRP2" )
         pay :  CLASS MORE_INTEREST SEQUENTIAL ( "GRP2" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  PRO_RATA ( "AII2#1"; "AII1#1"; "AII3#1" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  CLASS BALANCE SEQUENTIAL ( "AII4B" )
-----------------------------------
  subject to :  CEILING ( (BBAL("AII4B#1")-BBAL("AII4B")) )
         pay :  SEQUENTIAL ( "AII4B#1" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  CLASS BALANCE SEQUENTIAL ( "AII4A" )
-----------------------------------
  subject to :  CEILING ( (BBAL("AII4A#1")-BBAL("AII4A")) )
         pay :  SEQUENTIAL ( "AII4A#1" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  PRO_RATA ( "MII1#1" )
-----------------------------------
```

```
        from :  CLASS ( "GRP2" )
         pay :  PRO_RATA ( "MII2#1" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  PRO_RATA ( "MII3#1" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  PRO_RATA ( "MII4#1" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  PRO_RATA ( "MII5#1" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  PRO_RATA ( "SBII#1" )
-----------------------------------
        from :  CLASS ( "GRP2" )
         pay :  AS_INTEREST ( "R_2#1" )
-----------------------------------
        from :  CLASS ( "AII4B" )
         pay :  CLASS BALANCE SEQUENTIAL ("AII4")
-----------------------------------
        from :  CLASS ( "AII4" )
         pay :  PRO_RATA ( "AII4#1" )
-----------------------------------
        from :  CLASS ( "AII4A" )
         pay :  CLASS BALANCE SEQUENTIAL ("AII4")
-----------------------------------
        from :  CLASS ( "AII4" )
         pay :  PRO_RATA ( "AII4#1" )
-----------------------------------
   calculate : #WriteDown2 = BBAL("GRP2")
-----------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "SBII#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII5#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII4#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII3#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII2#1" )
-----------------------------------
        from :  SUBACCOUNT ( #Writedown2 )
         pay :  WRITEDOWN SEQUENTIAL ( "MII1#1" )
-----------------------------------
   calculate :  #WD_AII2 = MIN(BBAL("AII2#1"), #Writedown2 * BBAL("AII2#1") /
BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #WD_AII4B = MIN(BBAL("AII4B#1"), #Writedown2 * BBAL("AII4B#1")
/ BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #WD_AII1 = MIN(BBAL("AII1#1"), #Writedown2 * BBAL("AII1#1") /
BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #WD_AII3 = MIN(BBAL("AII3#1"), #Writedown2 * BBAL("AII3#1") /
BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #WD_AII4A = MIN(BBAL("AII4A#1"), #Writedown2 * BBAL("AII4A#1")
/ BBAL("AII2#1", "AII4B#1", "AII1#1", "AII3#1", "AII4A#1"))
   calculate :  #Writedown2 = #Writedown2 - #WD_AII2 - #WD_AII4B - #WD_AII1 -
#WD_AII3 - #WD_AII4A
-----------------------------------
        from :  SUBACCOUNT ( #WD_AII2 )
         pay :  WRITEDOWN SEQUENTIAL ( "AII2#1" )
-----------------------------------
  subject to :  CEILING ( #WD_AII4B )
         pay :  DECREMENT( BALANCE "AII4B#1", BY #WD_AII4B )
```

```
------------------------------------
        from :  SUBACCOUNT ( #WD_AII1 )
         pay :  WRITEDOWN SEQUENTIAL ( "AII1#1" )
------------------------------------
        from :  SUBACCOUNT ( #WD_AII3 )
         pay :  WRITEDOWN SEQUENTIAL ( "AII3#1" )
------------------------------------
  subject to :  CEILING ( #WD_AII4A )
         pay :  DECREMENT( BALANCE "AII4A#1", BY #WD_AII4A )
------------------------------------
!
 Collateral OVER
!
!       Factor      --Delay--
! Type   Date       P/Y    BV   Use BV for 0
  WL  20041001    9999 9999   FALSE
!
! Pool#  Type     Gross     Current     Original   --Fee--  Maturity Orig  ARM
          Gross  #mos  #mos   P#mos  P#mos  Life   Reset Life   Max   Look
!                Coupon    Factor      Balance    P/Y  BV  P/Y   BV Term
Index          Margin ToRst RstPer ToRst  RstPer Cap    Cap   Floor  Negam
Back
!! BEGINNING OF COLLATERAL
M        1               WL    00    WAC              6.069 (      9718262.92
/      9718262.92 );      9718262.92                      0.25          0.25
         178:2     178:2       180 NO_CHECK


                                                GROUP 1
M        2               WL    00    WAC              8.038 (      2941983.41
/      2941983.41 );      2941983.41                      0.389         0.389
         179:1     179:1       180 NO_CHECK


                                                GROUP 1
M        3               WL    00    WAC              6.377 (      1043338.82
/      1043338.82 );      1043338.82                      0.29          0.29
         239:1     239:1       240 NO_CHECK


                                                GROUP 1
M        4               WL    00    WAC              8.325 (      1937906.33
/      1937906.33 );      1937906.33                      0.406         0.406
         238:3     238:3       241 NO_CHECK


                                                GROUP 1
M        5               WL    00    WAC              6.576 (    121089487.41
/    121089487.41 );    121089487.41                      0.263         0.263
         359:1     359:1       360 NO_CHECK


                                                GROUP 1
M        6               WL    00    WAC              7.967 (    159008140.49
/    159008140.49 );    159008140.49                      0.31          0.31
         359:1     359:1       360 NO_CHECK


                                                GROUP 1
M        7               WL    00    WAC              6.515 (      2502769.75
/      2502769.75 );      2502769.75                      0.25          0.25
         359:1     359:1       360 NO_CHECK


                     AMORT NONE FOR         113 GROUP 1
M        8               WL    00    WAC              7.589 (      1269703.63
/      1269703.63 );      1269703.63                      0.351         0.351
         359:1     359:1       360 NO_CHECK
```

```
                    AMORT NONE FOR          83 GROUP 1
M        9              WL    00    WAC                8.8 (       488407.24
/       488407.24 );       488407.24                    0.329        0.329
         355:4     355:4       359 NO_CHECK

                                                                   BALLOON
SCHED_BOTH           180                            GROUP 1
M        11             WL    00    WAC              3.918 (      1747282.01
/      1747282.01 );      1747282.01                   0.25         0.25
         360:0     360:0       360 NO_CHECK ARM LIBOR_1YR
2.25    13   12 SYNC_INT            9.918                2            2.25
    0      0                              ORIG_GROSSRATE        3.918
                                                   GROUP 2       TEASER
M        12             WL    00    WAC              7.234 (      1346023.30
/      1346023.30 );      1346023.30                   0.25         0.25
         357:3     357:3       360 NO_CHECK ARM LIBOR_1YR
4.759    22   10 SYNC_INT           13.234             1.835          5.125
     0      0  INIT_PERCAP         1.933  ORIG_GROSSRATE       7.234
                                                   GROUP 2       TEASER
M        13             WL    00    WAC              5.473 (     10964323.48
/     10964323.48 );     10964323.48                   0.25         0.25
         359:1     359:1       360 NO_CHECK ARM LIBOR_1YR
2.266    36   12 SYNC_INT           11.473                2            2.266
     0      0                              ORIG_GROSSRATE       5.473
                                                   GROUP 2       TEASER
M        14             WL    00    WAC              5.528 (     15080914.72
/     15080914.72 );     15080914.72                   0.25         0.25
         359:1     359:1       360 NO_CHECK ARM LIBOR_1YR
2.251    60   12 SYNC_INT           10.528                2            2.251
     0      0  INIT_PERCAP             5  ORIG_GROSSRATE       5.528
                                                   GROUP 2       TEASER
M        15             WL    00    WAC              5.375 (       867386.81
/       867386.81 );       867386.81                   0.25         0.25
         360:0     360:0       360 NO_CHECK ARM LIBOR_1YR
2.25    85   12 SYNC_INT           11.375                2            2.25
    0      0  INIT_PERCAP             6  ORIG_GROSSRATE       5.375
                                                   GROUP 2       TEASER
M        16             WL    00    WAC               7.75 (        74851.78
/        74851.78 );        74851.78                   0.375        0.375
         355:5     355:5       360 NO_CHECK ARM CMT_1YR
9.875     8   12 SYNC_INT            13.75                2            9.75
     0      0                              ORIG_GROSSRATE        7.75
                                                   GROUP 2       TEASER
M        17             WL    00    WAC              6.658 (      1095408.42
/      1095408.42 );      1095408.42                   0.278        0.278
         358:2     358:2       360 NO_CHECK ARM LIBOR_6MO
4.284     5    6 SYNC_INT           11.496                1            4.284
     0      0  INIT_PERCAP         0.794  ORIG_GROSSRATE       6.658
                                                   GROUP 2       TEASER
M        18             WL    00    WAC               7.94 (    720931665.36
/    720931665.36 );    720931665.36                   0.462        0.462
         359:1     359:1       360 NO_CHECK ARM LIBOR_6MO
7.51    24    6 SYNC_INT           14.129             1.05          7.947
    0      0  INIT_PERCAP         2.846  ORIG_GROSSRATE        7.94
                                                   GROUP 2       TEASER
M        19             WL    00    WAC              7.797 (     97996617.40
/     97996617.40 );     97996617.40                   0.446        0.446
         359:1     359:1       360 NO_CHECK ARM LIBOR_6MO
7.385    36    6 SYNC_INT           13.865             1.05          7.708
     0      0  INIT_PERCAP         2.979  ORIG_GROSSRATE       7.797
                                                   GROUP 2       TEASER
M        20             WL    00    WAC              6.446 (     11161330.49
/     11161330.49 );     11161330.49                   0.25         0.25
         359:1     359:1       360 NO_CHECK ARM LIBOR_6MO
3.195    60    6 SYNC_INT           12.423             1.855          3.229
     0      0  INIT_PERCAP         5.613  ORIG_GROSSRATE       6.446
```

```
                                                    GROUP 2        TEASER
M        21              WL    00    WAC                5.601 (      6837447.94
/      6837447.94 );      6837447.94                      0.25          0.25
         358:2     358:2       360 NO_CHECK ARM LIBOR_1YR
2.265    35   12 SYNC_INT            11.601                2            2.265
     0      0  INIT_PERCAP          2.371  ORIG_GROSSRATE        5.601
                         AMORT NONE FOR          44 GROUP 2        TEASER
M        22              WL    00    WAC                5.566 (     13666297.53
/     13666297.53 );     13666297.53                      0.25          0.25
         358:2     358:2       360 NO_CHECK ARM LIBOR_1YR
2.256    59   12 SYNC_INT            10.595                2            2.256
     0      0  INIT_PERCAP          4.913  ORIG_GROSSRATE        5.566
                         AMORT NONE FOR          60 GROUP 2        TEASER
M        23              WL    00    WAC                 5.5 (       118425.40
/       118425.40 );       118425.40                      0.25          0.25
         359:1     359:1       360 NO_CHECK ARM LIBOR_1YR
2.25     84   12 SYNC_INT              10.5                2            2.25
     0      0  INIT_PERCAP              5  ORIG_GROSSRATE         5.5
                         AMORT NONE FOR          84 GROUP 2        TEASER
M        24              WL    00    WAC                4.951 (      1242112.37
/      1242112.37 );      1242112.37                      0.25          0.25
         358:2     358:2       360 NO_CHECK ARM CMT_1YR
2.25     35   12 SYNC_INT            10.951                2            2.25
     0      0                                ORIG_GROSSRATE        4.951
                         AMORT NONE FOR          55 GROUP 2        TEASER
M        25              WL    00    WAC                6.469 (       366851.03
/       366851.03 );       366851.03                      0.25          0.25
         359:1     359:1       360 NO_CHECK ARM CMT_1YR
2.545    60   12 SYNC_INT            12.469                2            2.545
     0      0                                 ORIG_GROSSRATE        6.469
                         AMORT NONE FOR          92 GROUP 2        TEASER
M        26              WL    00    WAC                5.897 (      3731638.73
/      3731638.73 );      3731638.73                      0.253         0.253
         359:1     359:1       360 NO_CHECK ARM LIBOR_6MO
3.655     6    9 SYNC_INT            12.194            3.227           3.655
     0      0  INIT_PERCAP          3.885  ORIG_GROSSRATE        5.897
                         AMORT NONE FOR         115 GROUP 2        TEASER
M        27              WL    00    WAC                6.736 (     37340131.94
/     37340131.94 );     37340131.94                      0.452         0.452
         359:1     359:1       360 NO_CHECK ARM LIBOR_6MO
6.401    24    6 SYNC_INT            12.709                1            6.601
     0      0  INIT_PERCAP              3  ORIG_GROSSRATE        6.736
                         AMORT NONE FOR          59 GROUP 2        TEASER
M        28              WL    00    WAC                6.493 (      9468109.51
/      9468109.51 );      9468109.51                      0.348         0.348
         358:2     358:2       360 NO_CHECK ARM LIBOR_6MO
4.861    35    7 SYNC_INT            12.493            1.383           4.948
     0      0  INIT_PERCAP           4.15  ORIG_GROSSRATE        6.493
                         AMORT NONE FOR          58 GROUP 2        TEASER
M        29              WL    00    WAC                6.361 (     15963181.78
/     15963181.78 );     15963181.78                      0.25          0.25
         359:1     359:1       360 NO_CHECK ARM LIBOR_6MO
2.89     60    6 SYNC_INT           12.304            1.929           2.89
     0      0  INIT_PERCAP           5.9  ORIG_GROSSRATE        6.361
                         AMORT NONE FOR          62 GROUP 2        TEASER
```