Generated by: calypso_user, DataServer: localhost Time: Tue Jul 09 07:41:19 EDT 2024
Name: EquityReverseConvertiblev1
Calculator: ForwardPayoff
Description: Reverse Convertible with KO

Script: Variables

```
Constant Start As ReferenceDate From Product.StartDate
Constant CouponPeriod As AccrualPeriod[]
Constant Maturity As PaymentDate From Product.Maturity
Constant Settlement As Enum 'Physical', 'Cash'
Constant KI As ReferenceDate[]
Constant KO As AccrualPeriod[]
Constant PrincipalProtection As Double From 1
Constant ITMParticipation As Double From 1
Constant StrikePct As Double From 1
Constant KI_BarrierPct As Double From 0.5
Constant KO_BarrierPct As Double From 1.5
Constant KO_Curr As Currency
Constant KO_FX As Double From 1
Constant KO_Rebate As Double From 0.1
Constant KO_PeriodDivisor As Double From 4
Constant KO_RebateType As Enum 'Absolute', 'Period'
Constant CouponRateStrikePct As Double From 1
Constant FinalCouponRate As Double From 0.0
Constant PayRec As Integer From Product.BuySell
Constant Curr As Currency From Product.Currency
Constant Notional As Double From Product.Notional
Constant Basket As Quotable[] From Product.Basket
Constant N As Integer From Product.BasketSize
Constant CouponRateHigh As Double From 0.1
Constant CouponRateLow As Double From 0
ValuationDate As ReferenceDate From Context.ValuationDate
KO_RebateAccrued As Double
AboveKO  As Boolean
AboveCP  As Boolean
DelivQty  As Double
RealizedCoupon  As Double
FlowValue  As Double
i  As Integer
KNOCKED_IN  As Boolean
KNOCKED_OUT  As Boolean
WorstLevel  As Double
WorstLevelIndex  As Integer
WorstLevelBarrierKO As Double
WorstLevelBarrierKI As Double
Strike  As Double[] From Basket
CouponRateStrike As Double[] From Basket
KI_Barrier  As Double[] From Basket
KO_Barrier  As Double[] From Basket
Performance  As Double[] From Basket
Option As Measure to NPV
Prob_KO As Measure
Prob_KI As Measure
m_Prob_KI As ScheduleMeasure From KI
m_WorstLevelIndex as Measure
```

Script: Forward

```
Start:
  For i = 1 To N
    Strike[i] = (StrikePct * Basket[i])
    KO_Barrier[i] = (KO_BarrierPct * Basket[i])
    KI_Barrier[i] = (KI_BarrierPct * Basket[i])
    CouponRateStrike[i] = (CouponRateStrikePct * Basket[i])
  Next
ValuationDate:
  WorstLevel = Infinity
  WorstLevelIndex = 1
  For i = 1 To N
    Performance[i] = (Basket[i] / Strike[i])
    If (Performance[i] < WorstLevel) Then
      WorstLevelIndex = i
      WorstLevel = Performance[i]
      WorstLevelBarrierKO = KO_Barrier[i]
      WorstLevelBarrierKI = KI_Barrier[i]
    EndIf
  Next
  m_WorstLevelIndex = WorstLevelIndex
KI:
  m_Prob_KI = 0
  If Not(KNOCKED_IN) Then
    For i = 1 To N
      If (Basket[i] <= KI_Barrier[i]) Then
        m_Prob_KI = 1
        KNOCKED_IN = True
      EndIf
    Next
  EndIf
CouponPeriod:
  If Not(KNOCKED_OUT) Then
    AboveCP = True
    For i = 1 To N
      If (Basket[i] < CouponRateStrike[i]) Then
        AboveCP = False
      EndIf
    Next
    RealizedCoupon = If(AboveCP, CouponRateHigh, CouponRateLow)
    FlowValue = Interest(Notional, (PayRec * RealizedCoupon), Curr, 'DGT_COUPON')
    Option += FlowValue
  EndIf
KO:
  KO_RebateAccrued += (KO_Rebate / KO_PeriodDivisor)
  If Not(KNOCKED_OUT) Then
    AboveKO = True
    For i = 1 To N
      If (Basket[i] < KO_Barrier[i]) Then
        AboveKO = False
      EndIf
    Next
    If AboveKO Then
      KNOCKED_OUT = True
      Prob_KO = 1
      Select Case KO_RebateType
        Case 'Absolute'
          FlowValue = Principal((((PayRec * Notional) * KO_FX) * (PrincipalProtection +
 KO_Rebate)), KO_Curr, 'KNOCK_OUT')
        Case 'Period'
```

```
          FlowValue = Principal((((PayRec * Notional) * KO_FX) * (PrincipalProtection +
 KO_RebateAccrued)), KO_Curr, 'KNOCK_OUT')

       EndSelect
      Option += FlowValue
    EndIf
  EndIf
Maturity:
  If Not(KNOCKED_OUT) Then
    WorstLevel = Infinity
    WorstLevelIndex = 1
    For i = 1 To N
      Performance[i] = (Basket[i] / Strike[i])
      If (Performance[i] < WorstLevel) Then
        WorstLevelIndex = i
        WorstLevel = Performance[i]
      EndIf
    Next
    If Not(KNOCKED_IN) Then
      FlowValue = Principal(((PayRec * Notional) * ((PrincipalProtection +
 (ITMParticipation * Max((Performance[WorstLevelIndex] - 1), 0))) + FinalCouponRate)),
 Curr, 'REDEMPTION')
      Option += FlowValue
    Else
      Prob_KI = 1.0
      If (Basket[WorstLevelIndex] >= Strike[WorstLevelIndex]) Then
        FlowValue = Principal(((PayRec * Notional) * PrincipalProtection), Curr,
 'REDEMPTION')
        Option += FlowValue
      Else
        Select Case Settlement
          Case 'Physical'
            DelivQty = (Notional / Strike[WorstLevelIndex])
            FlowValue = Physical((PayRec * DelivQty), Basket[WorstLevelIndex], 0.0, 1,
 'REDEMPTION', 1)
            Option += FlowValue
          Case 'Cash'
            FlowValue = Principal(((PayRec * Notional) * Performance[WorstLevelIndex]),
 Curr, 'REDEMPTION')
            Option += FlowValue

         EndSelect
      EndIf
    EndIf
  EndIf
```

Script: BOEvents

```
KNOCK_OUT|KNOCKED_OUT
KNOCK_IN|KNOCKED_IN
DGT_COUPON|EquityRate
```

Script: BarrierDescriptors

KO|WorstLevelBarrierKO|Basket|Up|Out|Closing|False
KI|WorstLevelBarrierKI|Basket|Down|In|Closing|False