```
Rem ===============================================================================
Rem       A_Reporting_BSPL preparation OOoBasic Program (Copyright:Shuichi Sunaga)
Rem      (2005/10/31:493)Adopted FindALL in AcctSum()(2006/05/14:549)
Rem      Segregated prior and post WsCalc procedures(2006/11/17:603)
Rem      Introduced PostBKChrg()(2007/12/10:783)
Rem      Inroduced RangeSum for posting to trend columns in PostXLn():(2008/12/5:849)
Rem      Concentration of Post2Owner and IndvBF into ND(2009/09/22:770)(2012/07/06:616)
Rem      Translation into English(2012/08/12:567)
Rem ===============================================================================
Rem Sub WsCalc()
Rem Sub ChkB4WsCalc()
Rem Sub PostXLn(ByVal WkNow As Object, ByVal adjClAct%, ByVal StrtLN%)
Rem Sub PostXFm()
Rem Sub PostBKChrg()
Rem Sub PostBFCF(ByVal WkNow As Object, ByVal StrtLN%)
Rem Function AcctSum&(ByVal WkWk as object, FkAcct$, fCol_Acct%, oSearchDesc, oSRange as object)
Rem Sub ChkAftWSCalc()
Rem Sub VeriDept()
Rem ===============================================================================
Option Explicit
Sub WsCalc()
    dim args(0) as new com.sun.star.beans.PropertyValue
    dim i%, mTopRow&
    oCurrYMD.setFocus()
    IntroTo3D()
    Rem (1)--- Quit WsCalc where current month transaction is non-exixtent ------
    If CellValue(WkBS, getFSRow%(WkBS, "ENDLN", ActCol) + 1, CBS_CM% -1) <> 0 Then
        MsgShow("InCompletePriorFS")
        End
    End If
    setCol4All()
    If ExamAllTopRow(cMonth%, "Top") then
        MsgTxt$ = "Will abort BS/PL Preparation because no transaction for the" & chr(13)
        MsgTxt$ = MsgTxt$ & "current month (" & CStr(cMonth%) & ") is found."
        MsgTtl$ = "Test of input transaction data"
        msgbox(MsgTxt$, 0, MsgTtl$)
        End
    End If
    Rem (2)---------- Change to manual calculation mode ----------------
    args(0).Name = "AutomaticCalculation"
    args(0).Value = False
    dispatcher.executeDispatch(aController, ".uno:AutomaticCalculation", "", 0, args())
    ThisComponent.calculateAll()
    Rem (3)----------Define Data Range subject to BS/PL preparation ----------------
    SetSrcRange "ValMo", cMonth%, cMonth%, "ActCD"
    Rem (4)---------- Verify Dr/Cr Balance for JV,AP,AR,MW -------------
    ChkB4WsCalc()
    Rem (5-1)-------- Make sum of posting source columns on the BS -----
    aController.ActiveSheet = WkBS
    WkBS.Unprotect("pwd")
    PostXLn WkBS, 6
    Rem (5-2)-------- Make sum of posting source columns on the PL -----
    aController.ActiveSheet = WkPL
    WkPL.Unprotect("pwd")
    If IsMultiDP then
        For i% = 1 To NumDpt
            WkDpt(i).Unprotect("pwd")
        Next i%
    End If
    PostXLn WkPL, 6
    Rem (5-3)------- Make sum of transactions for bank accounts --------
    aController.ActiveSheet = WkBS
    PostXFm
    Rem (5-4)------ Automatic JE for due from sole proprietorship------------
    If CorpOrIndv = "Indv" then
        Post2Owner() 'see INDIV
    End If
    Rem (5-5)------- Posting to trend analysis range（after PostXFm）--------
    PostBFCF WkBS, 6 '---Posting to BS trend
    BSPLShowCurrent("BP")
```

```
Rem ===============================================================================
Rem       A_Reporting_BSPL preparation OOoBasic Program (Copyright:Shuichi Sunaga)
Rem
Rem
Rem
Rem
Rem      (2012/04/24:573行)
Rem      Translation into English(2012/08/12:567)
Rem ===============================================================================
Rem Sub WsCalc()
Rem Sub ChkB4WsCalc()
Rem Sub PostXLn(ByVal WkNow As Object, ByVal adjClAct%, ByVal StrtLN%)
Rem Sub PostXFm()
Rem Sub PostBKChrg()
Rem Sub PostBFCF(ByVal WkNow As Object, ByVal StrtLN%)
Rem Function AcctSum&(ByVal WkWk as object, FkAcct$, fCol_Acct%, oSearchDesc, oSRange
Rem Sub ChkAftWSCalc()
Rem Sub VeriDept()
Rem ===============================================================================
Option Explicit
Sub WsCalc()

    Dim i%, mTopRow&, a%
    oCurrYMD.SetFocus
    IntroTo3D
    Rem (1)--- Quit WsCalc where current month transaction is non-exixtent ------
    If CellValue(WkBS, getFSRow%(WkBS, "ENDLN", ActCol) + 1, CBS_CM% - 1) <> 0 Then
        MsgShow ("InCompletePriorFS")
        End
    End If
    setCol4All
    If ExamAllTopRow(cMonth%, "Top") Then
        MsgTxt$ = "Will abort BS/PL Preparation because no transaction for the" & Chr(13)
        MsgTxt$ = MsgTxt$ & "current month (" & CStr(cMonth%) & ") is found."
        MsgTtl$ = "Test of input transaction data"
        resp% = MsgBox(MsgTxt$, 0, MsgTtl$)
        End
    End If
    Rem (2)---------- Change to manual calculation mode ----------------


    Application.Calculation = xlManual
    Calculate
    Rem (3)----------Define Data Range subject to BS/PL preparation ----------------
    SetSrcRange "ValMo", cMonth%, cMonth%, "ActCD"
    Rem (4)---------- Verify Dr/Cr Balance for JV,AP,AR,MW -------------
    ChkB4WsCalc
    Rem (5-1)-------- Make sum of posting source columns on the BS -----
    WkBS.Activate
    WkBS.Unprotect ("pwd")
    PostXLn WkBS, 6
    Rem (5-2)-------- Make sum of posting source columns on the PL -----
    WkPL.Activate
    WkPL.Unprotect ("pwd")
    If IsMultiDP Then
        For i% = 1 To NumDpt
            WkDpt(i).Unprotect ("pwd")
        Next i%
    End If
    PostXLn WkPL, 6
    Rem (5-3)------- Make sum of transactions for bank accounts --------
    WkBS.Activate
    PostXFm
    Rem (5-4)------ Automatic JE for due from sole proprietorship------------
    If CorpOrIndv = "Indv" Then
        Post2Owner 'See INDIV
    End If
    Rem (5-5)------- Posting to trend analysis range（after PostXFm）--------
    PostBFCF WkBS, 6 '---Posting to BS trend
    BSPLShowCurrent ("BP")
```

```
	Rem (6)---------- Change to Automatic calculation mode ----------
	args(0).Value = true
	dispatcher.executeDispatch(aController, ".uno:AutomaticCalculation", "", 0, args())
	Rem ⑦-------- Protect BS,PL, WkDpt(i) after FS Calculation --------
	WkBS.Protect("pwd")
	WkPL.Protect("pwd")
	If IsMultiDP then
		For i% = 1 To NumDpt
			WkDpt(i).Protect("pwd")
		Next i%
	End If
	Rem ⑧------- Handling where B/S Audit line shows other than zero --------
	ChkAftWSCalc()
	Rem ----------------------------------------------------------------------
	If IsAWDlgOpen() then
		oAWDlg.endExecute()
	End If
	GiveDTHeader()
	ThisComponent.calculateAll()
End Sub
Rem ---------- Verify Dr/Cr Balance for JV,AP,AR,MW ----------
Sub ChkB4WsCalc()
	setCol4All()
	Rem ----------------------------------------------------------
	If APUse Then
		getRowArray "AP", "ValMo", cMonth%, cMonth%
		Col_Gross% = Col_InArray("AP", "Gross")
		Col_Amount% = Col_InArray("AP", "Amount")
		if CellValue(WkAP, dbBtmBorder&, Col_Gross%) <> CellValue(WkAP, dbBtmBorder&, Col_Amount%) Then
			setOptState("AP")
			VeriByWS dbMonthTop&, 276
			End
		end if
	End If
	Rem ----------------------------------------------------------
	If ARUse Then
		getRowArray "AR", "ValMo", cMonth%, cMonth%
		Col_Gross% = Col_InArray("AR", "Gross")
		Col_Amount% = Col_InArray("AR", "Amount")
		if CellValue(WkAR, dbBtmBorder&, Col_Gross%) <> CellValue(WkAR, dbBtmBorder&, Col_Amount%) Then
			setOptState("AR")
			VeriByWS dbMonthTop&, 276
			End
		end If
	End If
	Rem ----------------------------------------------------------
	If MWUse Then
		getRowArray "MW", "ValMo", cMonth%, cMonth%
		Col_Gross% = Col_InArray("MW", "Gross")
		Col_Amount% = Col_InArray("MW", "Amount")
		if CellValue(WkMW, dbBtmBorder&, Col_Gross%) <> CellValue(WkMW, dbBtmBorder&, Col_Amount%) Then
			setOptState("MW")
			VeriByWS dbMonthTop&, 276
			End
		end if
	End If
	Rem ----------------------------------------------------------
	GetRowArray "JV", "ValMo", cMonth%, cMonth%
	Col_Gross% = Col_InArray("JV", "Gross")
	Col_Amount% = Col_InArray("JV", "Amount")
	If CellValue(WkJV, dbBtmBorder&, Col_Gross%) <> 0 or CellValue(WkJV, dbBtmBorder&, Col_Amount%) <> 0 Then
		setOptState("JV")
		VeriByWS dbMonthTop&, 276
		End
	End If
End Sub
Sub PostXLn(ByVal WkNow As Object, ByVal StrtLN%)
	dim i%, k%, mSigma%, sAName$, sAudWD$
	dim yCursor as object
	dim sEndCD$, WkAcct$
```

```
	Rem (6)---------- Change to Automatic calculation mode ----------

	Application.Calculation = xlAutomatic
	Rem (7)-------- Protect BS,PL, WkDpt(i) after FS Calculation --------
	WkBS.Protect ("pwd")
	WkPL.Protect ("pwd")
	If IsMultiDP Then
		For i% = 1 To NumDpt
			WkDpt(i).Protect ("pwd")
		Next i%
	End If
	Rem(8)------- Handling where B/S Audit line shows other than zero --------
	ChkAftWSCalc
	Rem ----------------------------------------------------------------------
	If IsAWDlgOpen() Then
		oAWDlg.Unload
	End If
	GiveDTHeader
	Calculate
End Sub
Rem ---------- Verify Dr/Cr Balance for JV,AP,AR,MW ----------
Sub ChkB4WsCalc()
	setCol4All
	Rem ----------------------------------------------------------
	If APUse Then
		getRowArray "AP", "ValMo", cMonth%, cMonth%
		Col_Gross% = Col_InArray("AP", "Gross")
		Col_Amount% = Col_InArray("AP", "Amount")
		If CellValue(WkAP, dbBtmBorder&, Col_Gross%) <> CellValue(WkAP, dbBtmBorder
			setOptState ("AP")
			VeriByWS dbMonthTop&, 276
			End
		End If
	End If
	Rem ----------------------------------------------------------
	If ARUse Then
		getRowArray "AR", "ValMo", cMonth%, cMonth%
		Col_Gross% = Col_InArray("AR", "Gross")
		Col_Amount% = Col_InArray("AR", "Amount")
		If CellValue(WkAR, dbBtmBorder&, Col_Gross%) <> CellValue(WkAR, dbBtmBorder
			setOptState ("AR")
			VeriByWS dbMonthTop&, 276
			End
		End If
	End If
	Rem ----------------------------------------------------------
	If MWUse Then
		getRowArray "MW", "ValMo", cMonth%, cMonth%
		Col_Gross% = Col_InArray("MW", "Gross")
		Col_Amount% = Col_InArray("MW", "Amount")
		If CellValue(WkMW, dbBtmBorder&, Col_Gross%) <> CellValue(WkMW, dbBtmBord
			setOptState ("MW")
			VeriByWS dbMonthTop&, 276
			End
		End If
	End If
	Rem ----------------------------------------------------------
	getRowArray "JV", "ValMo", cMonth%, cMonth%
	Col_Gross% = Col_InArray("JV", "Gross")
	Col_Amount% = Col_InArray("JV", "Amount")
	If CellValue(WkJV, dbBtmBorder&, Col_Gross%) <> 0 Or CellValue(WkJV, dbBtmBorder&
		setOptState ("JV")
		VeriByWS dbMonthTop&, 276
		End
	End If
End Sub
Sub PostXLn(ByVal WkNow As Object, ByVal StrtLN%)
	Dim i%, k%, mSigma%, sAName$, sAudWD$
	Dim sEndCD$, WkAcct$
	Dim bQuitKey As Boolean
```

```
dim bQuitKey as boolean
dim BFAmt&, CurrAmt&, ArrBP
BPRow& = StrtLN%
Do While True
    bQuitKey = False
    SelectCellFS(WkNow, BPRow&, BKCol)
    WkAcct$ = CellString(WkNow, BPRow&, ActCol)
    sAName$ = CellString(WkNow, BPRow&+1, CFCol-1)
    sEndCD$ = UCase(Trim(WkAcct$))
    sAudWD$ = myIIF(WkNow.Name = "BS", BSAudCD, PLAudCD)
    Rem ---------(Do not change the oreder of the following if statements)----------
    If sEndCD$ = "ENDLN" Then
        If sAName$ <> sAudWD$ then
            SetCellString WkNow, BPRow&+1, CFCol-1, sAudWD$
        End If
        Exit Sub
    ElseIf sAName$ = sAudWD$ Then
        If sEndCD$ <> "ENDLN" then
            SetCellString WkNow, BPRow&, ActCol, "ENDLN"
        End If
        Exit Sub
    ElseIf BPRow& > 300 Then
        Do While Cellstring(WkNow, BPRow&, CFCol-1) = ""
            BPRow& = BPRow& - 1
        Loop
        SetCellString WkNow, BPRow&, ActCol, "ENDLN"
        SetCellString WkNow, BPRow&+1, CFCol-1, sAudWD$
        Exit Sub
    ElseIf myIsEmpty(WkNow, BPRow&, 2)  Then
        bQuitKey = True
    Elseif Val(WkAcct$) = 0 then
        bQuitKey = True
    Elseif (Val(WkAcct$) = Val(REEndCD)) Then
        bQuitKey = True
    End If
    If bQuitKey = False then
        Rem --------------------(Get brought forward amounts)----------------
        if CorpOrIndv = "Indv" Then
            BFAmt& = IndvBF&(WkNow, WkAcct$, BPRow&) 'see INDIV
        else
            if A_VBSCD(WkAcct$) then  'BS Accounts
                If (Val(WkAcct$) = Val(REBegCD)) Then
                    If CBS_CM% = CBS_YE% +1 Or CBS_CM% = CBS_YB% Then
                        BFAmt& = CellValue(WkNow, getFSRow%(WkBS, REEndCD, ActCol), CBS_CM%-1)
                    Else
                        BFAmt& = CellValue(WkNow, getFSRow%(WkBS, REBegCD, ActCol), CBS_CM%-1)
                    End If
                Else
                    BFAmt& = CellValue(WkNow, BPRow&, CBS_CM%-1)
                End If
            elseif A_VPLCD(WkAcct$) then
                Rem Only when PL accounts also appear on BS
                Rem For each Dpt, posting will be done when dataArray is posted
                If CBS_CM% = CBS_YE% +1 Or CBS_CM% = CBS_YB% Then
                    BFAmt& = 0
                Else
                    If CBS_CM% <= CBS_YE% Then
                        BFAmt& = RangeSum(WkNow, CBS_YB%, BPRow&, CBS_CM%-1, BPRow&)
                    Else
                        BFAmt& = RangeSum(WkNow, CBS_YE%+1, BPRow&, CBS_CM%-1, BPRow&)
                    End If
                End If
            end if
        end if
        Rem ------------------- BK summation --------------------
        AmtArray&(0, 0) = AcctSum(WkBK, WkAcct$, oBKSDesc, BKRange)
        Rem ------------------- JV summation --------------------
        AmtArray&(0, 1) = AcctSum(WkJV, WkAcct$, oJVSDesc, JVRange)
        Rem ------------------- AP summation --------------------
        if APUse Then
```

```
Dim BFAmt&, CurrAmt&, ArrBP
BPRow& = StrtLN
Do While True
    bQuitKey = False
    SelectCellFS WkNow, BPRow&, BKCol
    WkAcct$ = CellString(WkNow, BPRow&, ActCol)
    sAName$ = CellString(WkNow, BPRow& + 1, CFCol - 1)
    sEndCD$ = UCase(Trim(WkAcct$))
    sAudWD$ = myIIF(WkNow.Name = "BS", BSAudCD, PLAudCD)
    Rem ---------(Do not change the oreder of the following if statements)----------
    If sEndCD$ = "ENDLN" Then
        If sAName$ <> sAudWD$ Then
            SetCellString WkNow, BPRow& + 1, CFCol - 1, sAudWD$
        End If
        Exit Sub
    ElseIf sAName$ = sAudWD$ Then
        If sEndCD$ <> "ENDLN" Then
            SetCellString WkNow, BPRow&, ActCol, "ENDLN"
        End If
        Exit Sub
    ElseIf BPRow& > 300 Then
        Do While CellString(WkNow, BPRow&, CFCol - 1) = ""
            BPRow& = BPRow& - 1
        Loop
        SetCellString WkNow, BPRow&, ActCol, "ENDLN"
        SetCellString WkNow, BPRow& + 1, CFCol - 1, sAudWD$
        Exit Sub
    ElseIf myIsEmpty(WkNow, BPRow&, 2) Then
        bQuitKey = True
    ElseIf Val(WkAcct$) = 0 Then
        bQuitKey = True
    ElseIf (Val(WkAcct$) = Val(REEndCD)) Then
        bQuitKey = True
    End If
    If bQuitKey = False Then
        Rem --------------------(Get brought forward amounts)----------------
        If CorpOrIndv = "Indv" Then
            BFAmt& = IndvBF&(WkNow, WkAcct$, BPRow&) 'see INDIV
        Else
            If A_VBSCD(WkAcct$) Then  'BS Accounts
                If (Val(WkAcct$) = Val(REBegCD)) Then
                    If CBS_CM% = CBS_YE% + 1 Or CBS_CM% = CBS_YB% Ther
                        BFAmt& = CellValue(WkNow, getFSRow%(WkBS, REEndC
                    Else
                        BFAmt& = CellValue(WkNow, getFSRow%(WkBS, REBegC
                    End If
                Else
                    BFAmt& = CellValue(WkNow, BPRow&, CBS_CM% - 1)
                End If
            ElseIf A_VPLCD(WkAcct$) Then
                Rem Only when PL accounts also appear on BS
                Rem For each Dpt, posting will be done when dataArray is posted
                If CBS_CM% = CBS_YE% + 1 Or CBS_CM% = CBS_YB% Then
                    BFAmt& = 0
                Else
                    If CBS_CM% <= CBS_YE% Then
                        BFAmt& = RangeSum(WkNow, CBS_YB%, BPRow&, CBS_
                    Else
                        BFAmt& = RangeSum(WkNow, CBS_YE% + 1, BPRow&, C
                    End If
                End If
            End If
        End If
        Rem ------------------- BK summation --------------------
        AmtArray&(0, 0) = AcctSum(WkBK, WkAcct$, BKRange)
        Rem ------------------- JV summation --------------------
        AmtArray&(0, 1) = AcctSum(WkJV, WkAcct$, JVRange)
        Rem ------------------- AP summation --------------------
        If APUse Then
            mSigma% = myIIF(CStr(WkAcct$) = CStr(APCD), -1, 1)
```

```
            mSigma%=myIIF(CStr(WkAcct$) = CStr(APCD), -1, 1)
            AmtArray&(0, 2) = AcctSum(WkAP, WkAcct$, oAPSDesc, APRange) * mSigma%
        end if
        Rem ------------------- AR summation --------------------
        if ARUse Then
            AmtArray&(0, 3) = AcctSum(WkAR, WkAcct$, oARSDesc, ARRange)
        end If
        Rem ------------------- MW summation --------------------
        if MWUse Then
            AmtArray&(0, 4) = AcctSum(WkMW, WkAcct$, oMWSDesc, MWRange)
        end If
        Rem -----------------------posting to BS/PL ----------------------
        Set yCursor = WkNow.getCellRangeByPosition(BFCol-1, BPRow&-1, RMCol-1, BPRow&-1)
        Rem --------------------setDataArray() corresponding to row array ----------
        if ARUse Then
            mSigma%=myIIF(CStr(WkAcct$) = CStr(ARCD), 1, -1)
            ArrBP=Array(BFAmt&, AmtArray&(0, 0), AmtArray&(0, 1), AmtArray&(0, 2), AmtArray&(0, 3)*mSigma%)
        elseif MWUse Then
            ArrBP=Array(BFAmt&, AmtArray&(0, 0), AmtArray&(0, 1), AmtArray&(0, 2), AmtArray&(0, 4))
        end if
        yCursor.setDataArray(Array(ArrBP))
        CurrAmt& = RangeSum(WkNow, BKCol, BPRow&, RMCol, BPRow&)
        if A_VPLCD(WkAcct$) then
            Rem ------------ posting of PL current month data to trend range -----------
            If Not WS_BS(WkAcct) then
                SetCellValue WkPL, BPRow&, CBS_CM%, CurrAmt&
            End If
        end if
        Rem ---------(compilation and posting for departmental accounting)------------------
        If IsMultiDP then
            if  A_VAllocCD(WkAcct$) then
                For i% = 1 to NumDpt
                    Set yCursor = WkDpt(i%).getCellRangeByPosition(BFCol-1, BPRow&-1, RMCol-1, BPRow&-1
                    Rem -----------------(get cumulative to prior month)--------------
                    If CBS_CM% = CBS_YE% +1 Or CBS_CM% = CBS_YB% Then
                        BFAmt& = 0
                    Else
                        If CBS_CM% <= CBS_YE% Then
                            BFAmt& = RangeSum(WkDpt(i), CBS_YB%, BPRow&, CBS_CM%-1, BPRow&)
                        Else
                            BFAmt& = RangeSum(WkDpt(i), CBS_YE%+1, BPRow&, CBS_CM%-1, BPRow&)
                        End If
                    End If
                    Rem --------------------------------------------------------------------



                    if ARUse Then
                        Rem note the minus sign in front of AmtArray&(i%, 3) for AR
                        ArrBP=Array(BFAmt&, AmtArray&(i%, 0), AmtArray&(i%, 1), AmtArray&(i%, 2), -AmtArray
                    elseif MWUse Then
                        ArrBP=Array(BFAmt&, AmtArray&(i%, 0), AmtArray&(i%, 1), AmtArray&(i%, 2), AmtArray&
                    end if
                    yCursor.setDataArray(Array(ArrBP))
                    CurrAmt& = RangeSum(WkDpt(i), BKCol, BPRow&, RMCol, BPRow&)
                    SetCellValue WkDpt(i), BPRow&, CBS_CM%, CurrAmt&
                    for k% = 0 to 4
                        AmtArray&(i%, k%) = 0
                    next k%
                Next i%
            end If
        End If
    End If
    BPRow& = BPRow& + 1
Loop
End Sub
Sub PostXFm()
    Rem ----- Current month summation of bank transaction and consumption tax ----
    dim aArgs(0) As Variant
    dim oFound as object, tBKRange As object
    Dim icCD%, WHTaxRow&, CTXRow&, CXPRow&, BKChgRow&
```

```
            AmtArray&(0, 2) = AcctSum(WkAP, WkAcct$, APRange) * mSigma%
        End If
        Rem ------------------- AR summation --------------------
        If ARUse Then
            AmtArray&(0, 3) = AcctSum(WkAR, WkAcct$, ARRange)
        End If
        Rem ------------------- MW summation --------------------
        If MWUse Then
            AmtArray&(0, 4) = AcctSum(WkMW, WkAcct$, MWRange)
        End If
        Rem -----------------------posting to BS/PL ----------------------
        SetCellValue WkNow, BPRow&, BFCol, BFAmt&
        SetCellValue WkNow, BPRow&, BKCol, AmtArray&(0, 0)
        SetCellValue WkNow, BPRow&, JVCol, AmtArray&(0, 1)
        SetCellValue WkNow, BPRow&, APCol, AmtArray&(0, 2)
        If ARUse Then
            mSigma% = myIIF(CStr(WkAcct$) = CStr(ARCD), 1, -1)
            SetCellValue WkNow, BPRow&, ARCol, AmtArray&(0, 3) * mSigma%
        ElseIf MWUse Then
            SetCellValue WkNow, BPRow&, MWCol, AmtArray&(0, 4)
        End If
        CurrAmt& = RangeSum(WkNow, BKCol, BPRow&, RMCol, BPRow&)
        If A_VPLCD(WkAcct$) Then
            Rem ------------ posting of PL current month data to trend range -----------
            If Not WS_BS(WkAcct) Then
                SetCellValue WkPL, BPRow&, CBS_CM%, CurrAmt&
            End If
        End If
        Rem ---------(compilation and posting for departmental accounting)-----------------
        If NumDpt > 1 Then
            If A_VAllocCD(WkAcct$) Then
                For i% = 1 To NumDpt
                    Rem -----------------(get cumulative to prior month)--------------
                    If CBS_CM% = CBS_YE% + 1 Or CBS_CM% = CBS_YB% Ther
                        BFAmt& = 0
                    Else
                        If CBS_CM% <= CBS_YE% Then
                            BFAmt& = RangeSum(WkDpt(i%), CBS_YB%, BPRo
                        Else
                            BFAmt& = RangeSum(WkDpt(i%), CBS_YE% + 1, BI
                        End If
                    End If
                    Rem --------------------------------------------------------------------
                    SetCellValue WkDpt(i%), BPRow&, BFCol, BFAmt&
                    SetCellValue WkDpt(i%), BPRow&, BKCol, AmtArray&(i%, 0)
                    SetCellValue WkDpt(i%), BPRow&, JVCol, AmtArray&(i%, 1)
                    SetCellValue WkDpt(i%), BPRow&, APCol, AmtArray&(i%, 2)
                    If ARUse Then
                        Rem note the minus sign in front of AmtArray&(i%, 3) for AR
                        SetCellValue WkDpt(i%), BPRow&, ARCol, -AmtArray&(i%,
                    ElseIf MWUse Then
                        SetCellValue WkDpt(i%), BPRow&, MWCol, AmtArray&(i%,
                    End If
                    CurrAmt& = RangeSum(WkDpt(i), BKCol, BPRow&, RMCol, BPF
                    SetCellValue WkDpt(i%), BPRow&, CBS_CM%, CurrAmt&
                    For k% = 0 To 4
                        AmtArray&(i%, k%) = 0
                    Next k%
                Next i%
            End If
        End If
    End If
    BPRow& = BPRow& + 1
Loop
End Sub
Sub PostXFm()
    Rem ----- Current month summation of bank transaction and consumption tax ----

    Dim oFound As Object, tBKRange As Range, oRange As Range
    Dim icCD%, WHTaxRow&, CTXRow&, CXPRow&, BKChgRow&
```

```
dim iStrRow&, iEndRow&
dim BKChgAmt&, tVal&, WHTax&, BKAmt&, CTXPAY&, CTXAmt&, nwVal&, CurrAmt&
dim ac%, MBKC%, ColName$
getPagePlan "BK", cMonth%
Rem If WkBK.AutoFilterMode Then
Rem    WkBK.Columns("A:M").AutoFilter
Rem End If
icCD% = Col_BKCD%
Rem ------------------------ Calculation by bank ------------------------
CTXRow& = getFSRow%(WkBS, CRTXCD, ActCol)
CXPRow& = getFSRow%(WkBS, PYTXCD, ActCol)
WHTaxRow& = getFSRow%(WkBS, WHTaxCD, ActCol)
BKChgRow& = getFSRow%(WkPL, BKChgCD, ActCol)
iEndRow& = dbMonthTop& -1
Do While True
    RNM$ = CAlpha(icCD%) & CStr(iEndRow& + 1) & cjc$
    RNM$ = RNM$ & CAlpha(icCD%) & Cstr(dbMonthBtm&)
    Set tBKRange = WkBK.getCellRangeByName(RNM$)
    MBKC% = CellValue(WkBK, iEndRow&+1, Col_BKCD%)
    BPRow& = getFSRow%(WkBS, CStr(MBKC%), ActCol)
    If iEndRow& = dbMonthTop& then
        SetCellValue WkCL, 1, 28, BPRow&
    End If
    SetCellValue WkCL, BPRow&, 28, CellValue(WkBS, BPRow&, 2)
    iStrRow& = getFirstRow&(tBKRange, Cstr(MBKC%))
    iEndRow& = getLastRow&(tBKRange, Cstr(MBKC%))
    SetCellValue WkCL, BPRow&, 30, CellValue(WkBK, iEndRow&, Col_Gross%+3)
    SelectCellFS WkBS, BPRow&, BKCol
    Rem ===Summation of increase and decrease of Bank amounts ================
    If CellValue(WkBK, tBKRange.RangeAddress.StartRow+1, Col_ValMo%) = cMonth% then
        BKAmt& = RangeSum&(WkBK, Col_Amount%, iStrRow&, Col_Amount%, iEndRow&)
    Else
        BKAmt& = 0
    End If
    Rem ===Summation of VAT received only when the column is setup ===========
    If Col_CXCr% < Col_Border% then
        CTXPAY& = RangeSum&(WkBK, Col_CXCr%, iStrRow&, Col_CXCr%, iEndRow&)
    Else
        CTXPAY& = 0
    End If
    Rem ==============(Summation of bank transfer charges)==================
    BKChgAmt& = RangeSum&(WkBK, Col_BKChg%, iStrRow&, Col_BKChg%, iEndRow&)
    CTXAmt& = Int((BKChgAmt& / (100 + CTXRate)) * CTXRate)
    tVal& = Val(CellValue(WkPL, BKChgRow&, BKCol))
    BKChgAmt& = BKChgAmt& - CTXAmt&
    SetCellValue WkPL, BKChgRow&, BKCol, tVal& + BKChgAmt&
    CurrAmt& = RangeSum(WkPL, BKCol, BKChgRow&, RMCol, BKChgRow&)
    SetCellValue WkPL, BKChgRow&, CBS_CM%, CurrAmt&
    If IsMultiDP Then
        tVal& = Val(CellValue(WkDpt(NumDpt), BKChgRow&, BKCol))
        SetCellValue WkDpt(NumDpt), BKChgRow&, BKCol, tVal& + BKChgAmt&
        CurrAmt& = RangeSum(WkDpt(NumDpt), BKCol, BKChgRow&, RMCol, BKChgRow&)
        SetCellValue WkDpt(NumDpt), BKChgRow&, CBS_CM%, CurrAmt&
    End If
    Rem ======================Summation of VAT paid=======================
    CTXAmt& = RangeSum&(WkBK, Col_CXDr%, iStrRow&, Col_CXDr%, iEndRow&) + CTXAmt&
    Rem ================Posting  of VAT paid and VAT received ===============
    tVal& = CellValue(WkBS, CTXRow&, BKCol)
    SetCellValue WkBS, CTXRow&, BKCol, tVal& + CTXAmt&
    If Col_CXCr% < Col_Border% then
        tVal& = CellValue(WkBS, CXPRow&, BKCol)
        SetCellValue WkBS, CXPRow&, BKCol, tVal& + CTXPAY&
    End If
    Rem ============ Summation of Withholding tax received and posting to BS ==========
    WHTax& = RangeSum&(WkBK, Col_WHT%, iStrRow&, Col_WHT%, iEndRow&)
    tVal& = CellValue(WkBS, WHTaxRow&, BKCol)
    SetCellValue WkBS, WHTaxRow&, BKCol, tVal& + WHTax&
    Rem ===Posting to BK acount on BS, and month end balance to the BK row on CL =====
    tVal& = CellValue(WkBS, BPRow&, BKCol)
    tVal& = tVal& - BKAmt& - CTXPAY& - CTXAmt& - BKChgAmt& - WHTax& 'note CTXAmt
```

```
Dim iStrRow&, iEndRow&
Dim BKChgAmt&, tVal&, WHTax&, BKAmt&, CTXPAY&, CTXAmt&, nwVal&, CurrAmt&
Dim AC%, MBKC%, tBKC%, ColName$
getPagePlan "BK", cMonth%
Rem If WkBK.AutoFilterMode Then
Rem    WkBK.Columns("A:M").AutoFilter
Rem End If
icCD% = Col_BKCD%
Rem ------------------------ Calculation by bank ------------------------
CTXRow& = getFSRow%(WkBS, CRTXCD, ActCol)
CXPRow& = getFSRow%(WkBS, PYTXCD, ActCol)
WHTaxRow& = getFSRow%(WkBS, WHTaxCD, ActCol)
BKChgRow& = getFSRow%(WkPL, BKChgCD, ActCol)
iEndRow& = dbMonthTop& - 1
Do While True
    RNM$ = CAlpha(icCD%) & CStr(iEndRow& + 1) & ":"
    RNM$ = RNM$ & CAlpha(icCD%) & CStr(dbMonthBtm&)
    Set tBKRange = WkBK.Range(RNM$)
    MBKC% = WkBK.Cells(iEndRow& + 1, Col_BKCD%).Value
    BPRow& = getFSRow%(WkBS, CStr(MBKC%), ActCol)
    If iEndRow& = dbMonthTop& Then
        SetCellValue WkCL, 1, 28, BPRow&
    End If
    SetCellValue WkCL, BPRow&, 28, CellValue(WkBS, BPRow&, 2)
    iStrRow& = getFirstRow&(tBKRange, CStr(MBKC%))
    iEndRow& = getLastRow&(tBKRange, CStr(MBKC%))
    SetCellValue WkCL, BPRow&, 30, CellValue(WkBK, iEndRow&, Col_Gross% + 3)
    SelectCell BPRow&, BKCol
    Rem ===Summation of increase and decrease of Bank amounts ================
    If CellValue(WkBK, iStrRow&, Col_ValMo%) = cMonth% Then
        BKAmt& = RangeSum&(WkBK, Col_Amount%, iStrRow&, Col_Amount%, iEndl
    Else
        BKAmt& = 0
    End If
    Rem ===Summation of VAT received only when the column is setup ===========
    If Col_CXCr% < Col_Border% Then
        CTXPAY& = RangeSum&(WkBK, Col_CXCr%, iStrRow&, Col_CXCr%, iEndRo
    Else
        CTXPAY& = 0
    End If
    Rem ==============(Summation of bank transfer charges)==================
    BKChgAmt& = RangeSum&(WkBK, Col_BKChg%, iStrRow&, Col_BKChg%, iEndRo
    CTXAmt& = Int((BKChgAmt& / (100 + CTXRate)) * CTXRate)
    tVal& = Val(CellValue(WkPL, BKChgRow&, BKCol))
    BKChgAmt& = BKChgAmt& - CTXAmt&
    SetCellValue WkPL, BKChgRow&, BKCol, tVal& + BKChgAmt&
    CurrAmt& = RangeSum(WkPL, BKCol, BKChgRow&, RMCol, BKChgRow&)
    SetCellValue WkPL, BKChgRow&, CBS_CM%, CurrAmt&
    If IsMultiDP Then
        tVal& = Val(CellValue(WkDpt(NumDpt), BKChgRow&, BKCol))
        SetCellValue WkDpt(NumDpt), BKChgRow&, BKCol, tVal& + BKChgAmt&
        CurrAmt& = RangeSum(WkDpt(NumDpt), BKCol, BKChgRow&, RMCol, BKChg
        SetCellValue WkDpt(NumDpt), BKChgRow&, CBS_CM%, CurrAmt&
    End If
    Rem ======================Summation of VAT paid=======================
    CTXAmt& = RangeSum&(WkBK, Col_CXDr%, iStrRow&, Col_CXDr%, iEndRow&) +
    Rem ================Posting  of VAT paid and VAT received ==============
    tVal& = CellValue(WkBS, CTXRow&, BKCol)
    SetCellValue WkBS, CTXRow&, BKCol, tVal& + CTXAmt&
    If Col_CXCr% < Col_Border% Then
        tVal& = CellValue(WkBS, CXPRow&, BKCol)
        SetCellValue WkBS, CXPRow&, BKCol, tVal& + CTXPAY&
    End If
    Rem ============ Summation of Withholding tax received and posting to BS ====
    WHTax& = RangeSum&(WkBK, Col_WHT%, iStrRow&, Col_WHT%, iEndRow&)
    tVal& = CellValue(WkBS, WHTaxRow&, BKCol)
    SetCellValue WkBS, WHTaxRow&, BKCol, tVal& + WHTax&
    Rem ===Posting to BK acount on BS, and month end balance to the BK row on CL =
    tVal& = CellValue(WkBS, BPRow&, BKCol)
    tVal& = tVal& - BKAmt& - CTXPAY& - CTXAmt& - BKChgAmt& - WHTax& 'CTXAmtI
```

```
            SetCellValue WkBS, BPRow&, BKCol, tVal&
            tVal& = CellValue(WkBS, BPRow&, BFCol) + tVal& + CellValue(WkBS, BPRow&, JVCol)
            SetCellValue WkCL, BPRow&, 29, tVal&
            Rem ============================================================
            If iEndRow&+1 > dbMonthBtm& then
                exit do
            End if
        Loop
        Rem ------------------- Amounts to AR bottom row ---------------------
        If ARUse Then
            BPRow& = getFSRow%(WkBS, ARCD, ActCol)
            dbBtmBorder&=RowArray(2, 3)
            Col_Amount%=Col_InArray("AR", "Amount")
            ac=myIIF(bAllocDpt, NumDpt, 0)
            tVal& = CellValue(WkBS, BPRow&, BKCol - 1) 'Balance at beginning
            SetCellValue WkAR, dbBtmBorder& + 1, Col_Amount% + ac + 1, tVal&
            tVal& = CellValue(WkBS, BPRow&, ARCol) 'Current month Increase
            SetCellValue WkAR, dbBtmBorder& + 1, Col_Amount% + ac + 2, tVal&
            tVal& = -CellValue(WkBS, BPRow&, BKCol) -CellValue(WkBS, BPRow&, JVCol)
            SetCellValue WkAR, dbBtmBorder& + 1, Col_Amount% + ac + 3, tVal&
            tVal& = RangeSum(WkBS, BFCol, BPRow&, ARCol, BPRow&) 'Balance at ending
            SetCellValue WkAR, dbBtmBorder& + 1, Col_Amount% + ac + 4, tVal&
        End If
        Rem ------------------- Amounts to MW bottom row ---------------------
        If MWUse Then
            BPRow& = getFSRow%(WkBS, MWCD, ActCol)
            dbBtmBorder&=RowArray(2, 4)
            Col_Amount%=Col_InArray("MW", "Amount")
            ac=myIIF(bAllocDpt, NumDpt, 0)
            tVal& = CellValue(WkBS, BPRow&, BKCol - 1) 'Balance at beginning
            SetCellValue WkMW, dbBtmBorder& + 1, Col_Amount% + ac + 1, tVal&
            tVal& = CellValue(WkBS, BPRow&, BKCol) + CellValue(WkBS, BPRow&, JVCol)
            SetCellValue WkMW, dbBtmBorder& + 1, Col_Amount% + ac + 2, tVal&
            tVal& = -CellValue(WkBS, BPRow&, MWCol) 'Current month payment
            SetCellValue WkMW, dbBtmBorder& + 1, Col_Amount% + ac + 3, tVal&
            tVal& = RangeSum(WkBS, BFCol, BPRow&, MWCol, BPRow&) 'Balance at ending
            SetCellValue WkMW, dbBtmBorder& + 1, Col_Amount% + ac + 4, tVal&
        End If
        Rem ------------------- Amounts to AP bottom row ---------------------
        If APUse Then
            BPRow& = getFSRow%(WkBS, APCD, ActCol)
            dbBtmBorder&=RowArray(2, 2)
            Col_Amount%=Col_InArray("AP", "Amount")
            ac=myIIF(bAllocDpt, NumDpt, 0)
            tVal& = -CellValue(WkBS, BPRow&, BKCol-1) 'Balance at beginning
            SetCellValue WkAP, dbBtmBorder& + 1, Col_Amount% + ac + 1, tVal&
            tVal& = -CellValue(WkBS, BPRow&, APCol) 'Increase
            SetCellValue WkAP, dbBtmBorder& + 1, Col_Amount% + ac + 2, tVal&
            tVal& = CellValue(WkBS, BPRow&, BKCol) + CellValue(WkBS, BPRow&, JVCol) 'Decrease
            SetCellValue WkAP, dbBtmBorder& + 1, Col_Amount% + ac + 3, tVal&
            tVal& = -RangeSum(WkBS, BFCol, BPRow&, APCol, BPRow&) 'Balance at ending
            SetCellValue WkAP, dbBtmBorder& + 1, Col_Amount% + ac + 4, tVal&
        End If
    End Sub
    Sub PostBFCF(ByVal WkNow As Object, ByVal StrtLN%)
        Dim CumAmount&, WkAcct$
        BPRow& = StrtLN%
        Do While True
            SelectCellFS(WkNow, BPRow&, BKCol)
            WkAcct$ = CellString(WkNow, BPRow&, 2)
            If Val(WkAcct$) > 0 And Len(WkAcct$) > 0 Then
                if A_VBSCD(WkAcct$) Then  Rem BS Accounts
                    If Val(WkAcct$) = Val(REEndCD) Then
                        Rem do nothing
                    Else
                        CumAmount& = RangeSum(WkNow, BFCol, BPRow&, RMCol, BPRow&)
                        SetCellValue WkNow, BPRow&, CBS_CM%, CumAmount&
                    End If
                elseif A_VPLCD(WkAcct$) Then Rem PL Accounts
                    CumAmount& = RangeSum(WkNow, BKCol, BPRow&, RMCol, BPRow&)
```

```
            SetCellValue WkBS, BPRow&, BKCol, tVal&
            tVal& = CellValue(WkBS, BPRow&, BFCol) + tVal& + CellValue(WkBS, BPRow&, JV
            SetCellValue WkCL, BPRow&, 29, tVal&
            Rem ============================================================
            If iEndRow& + 1 > dbMonthBtm& Then
                Exit Do
            End If
        Loop
        Rem ------------------- Amounts to AR bottom row ---------------------
        If ARUse Then
            BPRow& = getFSRow%(WkBS, ARCD, ActCol)
            dbBtmBorder& = RowArray(2, 3)
            Col_Amount% = Col_InArray("AR", "Amount")
            AC = myIIF(bAllocDpt, NumDpt, 0)
            tVal& = CellValue(WkBS, BPRow&, BKCol - 1) 'Balance at beginning
            SetCellValue WkAR, dbBtmBorder& + 1, Col_Amount% + AC + 1, tVal&
            tVal& = CellValue(WkBS, BPRow&, ARCol) 'Current month Increase
            SetCellValue WkAR, dbBtmBorder& + 1, Col_Amount% + AC + 2, tVal&
            tVal& = -CellValue(WkBS, BPRow&, BKCol) - CellValue(WkBS, BPRow&, JVCol)
            SetCellValue WkAR, dbBtmBorder& + 1, Col_Amount% + AC + 3, tVal&
            tVal& = RangeSum(WkBS, BFCol, BPRow&, ARCol, BPRow&) 'Balance at ending
            SetCellValue WkAR, dbBtmBorder& + 1, Col_Amount% + AC + 4, tVal&
        End If
        Rem ------------------- Amounts to MW bottom row ---------------------
        If MWUse Then
            BPRow& = getFSRow%(WkBS, MWCD, ActCol)
            dbBtmBorder& = RowArray(2, 4)
            Col_Amount% = Col_InArray("MW", "Amount")
            AC = myIIF(bAllocDpt, NumDpt, 0)
            tVal& = CellValue(WkBS, BPRow&, BKCol - 1) 'Balance at beginning
            SetCellValue WkMW, dbBtmBorder& + 1, Col_Amount% + AC + 1, tVal&
            tVal& = CellValue(WkBS, BPRow&, BKCol) + CellValue(WkBS, BPRow&, JVCol)
            SetCellValue WkMW, dbBtmBorder& + 1, Col_Amount% + AC + 2, tVal&
            tVal& = -CellValue(WkBS, BPRow&, MWCol) 'Current month payment
            SetCellValue WkMW, dbBtmBorder& + 1, Col_Amount% + AC + 3, tVal&
            tVal& = RangeSum(WkBS, BFCol, BPRow&, MWCol, BPRow&) 'Balance at ending
            SetCellValue WkMW, dbBtmBorder& + 1, Col_Amount% + AC + 4, tVal&
        End If
        Rem ------------------- Amounts to AP bottom row ---------------------
        If APUse Then
            BPRow& = getFSRow%(WkBS, APCD, ActCol)
            dbBtmBorder& = RowArray(2, 2)
            Col_Amount% = Col_InArray("AP", "Amount")
            AC = myIIF(bAllocDpt, NumDpt, 0)
            tVal& = -CellValue(WkBS, BPRow&, BKCol - 1) 'Balance at beginning
            SetCellValue WkAP, dbBtmBorder& + 1, Col_Amount% + AC + 1, tVal&
            tVal& = -CellValue(WkBS, BPRow&, APCol) 'Increase
            SetCellValue WkAP, dbBtmBorder& + 1, Col_Amount% + AC + 2, tVal&
            tVal& = CellValue(WkBS, BPRow&, BKCol) + CellValue(WkBS, BPRow&, JVCol)
            SetCellValue WkAP, dbBtmBorder& + 1, Col_Amount% + AC + 3, tVal&
            tVal& = -RangeSum(WkBS, BFCol, BPRow&, APCol, BPRow&) 'Balance at ending
            SetCellValue WkAP, dbBtmBorder& + 1, Col_Amount% + AC + 4, tVal&
        End If
    End Sub
    Sub PostBFCF(ByVal WkNow As Object, ByVal StrtLN%)
        Dim CumAmount&, WkAcct$, oRange As Range
        BPRow& = StrtLN%
        Do While True
            SelectCellFS WkNow, BPRow&, BKCol
            WkAcct$ = CellString(WkNow, BPRow&, 2)
            If Val(WkAcct$) > 0 And Len(WkAcct$) > 0 Then
                If A_VBSCD(WkAcct$) Then  'BS Accounts
                    If Val(WkAcct$) = Val(REEndCD) Then
                        Rem do nothing
                    Else
                        CumAmount& = RangeSum&(WkNow, BFCol, BPRow&, RMCol, BPR
                        SetCellValue WkNow, BPRow&, CBS_CM%, CumAmount&
                    End If
                ElseIf A_VPLCD(WkAcct$) Then 'PL Accounts
                    CumAmount& = RangeSum&(WkNow, BKCol, BPRow&, RMCol, BPRow&
```

```
                    SetCellValue WkNow, BPRow&, CBS_CM%, CumAmount&
                end if
            End If
            If WkAcct$ = "ENDLN" Then
                Exit Do
            Else
                BPRow& = BPRow& + 1
            End If
        Loop
End Sub
Function AcctSum&(ByVal WkWk as object, FkAcct$, oSearchDesc, oSRange as object)
        dim i%, j%, k%, xRow&, iCol%, oFLength%
        dim oFound as object, LnAcct$, LnMonth%, pSName$
        AcctSum& = 0
        pSName$ = WkWk.getName()
        iCol% = iWS(pSName$)
        If RowArray(3,iCol%) then
            Exit Function
        End If
        oSearchDesc.SearchWords = true
        oSearchDesc.SearchString = FKAcct$
        oSearchDesc.SearchBackwards = False
        Set oFound = oSRange.findAll(oSearchDesc)
        If IsNull(oFound) then
            Exit Function
        Endif
        getColArray(pSName$)
        oFLength% = oFound.getCount()
        For i% = 0 to oFLength%-1
            xRow& = oFound.getByIndex(i%).RangeAddress.StartRow + 1
            for j% = 1 to oFound.getByIndex(i%).Rows.Count
                LnMonth% = Val(CellString(WkWk, xRow&, Col_ValMo%))
                LnAcct$ = CellString(WkWk, xRow&, Col_ActCD%)
                If LnMonth% = cMonth% and LnAcct$ = FkAcct$ then
                    AcctSum& = AcctSum& + CellValue(WkWk, xRow&, Col_Amount%)
                    if IsMultiDP AND bAllocDpt Then
                        If A_VAllocCD(FkAcct$) Then
                            For k% = 1 To NumDpt
                                AmtArray&(k%, iCol%) =AmtArray&(k%, iCol%) +CellValue(WkWk, xRow&, Col_Amount%
                            Next k%
                        End If
                    end If
                End If
                xRow& = xRow& + 1
            next j%
        Next i%
End Function
Rem ------- Handling where B/S Audit line shows other than zero --------
Sub ChkAftWSCalc()
        dim i%, BSAudLN%, VeriV&
        dim iRow&, sBK$, bkRow&, cBKCD%
        Rem ----------Tie in BS and BK balances-------------------------
        iRow = CellValue(WkCL, 1, 28)
        Do While Len(CellString(WkCL, iRow, 28)) > 0
            If CellValue(WkCL, iRow, 29) <> CellValue(WkCL, iRow, 30) then
                sBK$ = CellString(WkBS, iRow, 2)
                MsgTxt$ = "There is an inbalance between BS and BK regarding" & chr(13)
                MsgTxt$ = MsgTxt$  & sBK$ & ">" & CellString(WkBS, iRow, 4)& "." & Chr(13)
                MsgTxt$ = MsgTxt$ & "Will proceed to Verification by row to identify the cause."
                cBKCD% = Col_InArray("BK", "BKCD")
                bkRow& = RowArray(0,0)+1
                Do While bkRow& <= RowArray(1,0)
                    If CellString(WkBK, bkRow&, cBKCD%) = sBK$ Then
                        Exit Do
                    End If
                    bkRow& = bkRow& + 1
                Loop
                setOptState("BK")
                SelectCell bkRow&, cBKCD%
                MsgTtl$ = "Test of agreement of BS and BK balances"
```

```
                    SetCellValue WkNow, BPRow&, CBS_CM%, CumAmount&
                End If
            End If
            If WkAcct$ = "ENDLN" Then
                Exit Do
            Else
                BPRow& = BPRow& + 1
            End If
        Loop
End Sub
Function AcctSum&(ByVal WkWk As Object, FkAcct$, oSRange As Object)
        Dim k%, xRow&, iCol%, firstAddress
        Dim oFound As Object, LnAcct$, LnMonth%, pSName$
        AcctSum& = 0
        pSName$ = WkWk.Name
        iCol% = iWS(pSName$)
        If RowArray(3, iCol%) Then
            Exit Function
        End If



        Set oFound = oSRange.Find(FkAcct$, Lookat:=xlWhole, SearchDirection:=xlNext)
        If oFound Is Nothing Then
            Exit Function
        End If
        getColArray (pSName$)
        firstAddress = oFound.Address
        Do
            xRow& = oFound.Row
            LnMonth% = Val(CellString(WkWk, xRow&, Col_ValMo%))
            LnAcct$ = CellString(WkWk, xRow&, Col_ActCD%)
            If LnMonth% = cMonth% And LnAcct$ = FkAcct$ Then
                AcctSum& = AcctSum& + CellValue(WkWk, xRow&, Col_Amount%)
                If IsMultiDP And bAllocDpt Then
                    If A_VAllocCD(FkAcct$) Then
                        For k% = 1 To NumDpt
                            AmtArray&(k%, iCol%) = AmtArray&(k%, iCol%) + CellValue(Wk
                        Next k%
                    End If
                End If
            End If



            Set oFound = oSRange.FindNext(oFound)
        Loop While Not oFound Is Nothing And oFound.Address <> firstAddress
End Function
Rem ------- Handling where B/S Audit line shows other than zero --------
Sub ChkAftWSCalc()
        Dim i%, BSAudLN%, VeriV&, a%
        Dim iRow&, sBK$, bkRow&, cBKCD%
        Rem ----------Tie in BS and BK balances-------------------------
        iRow& = CellValue(WkCL, 1, 28)
        Do While Len(CellString(WkCL, iRow&, 28)) > 0
            If CellValue(WkCL, iRow&, 29) <> CellValue(WkCL, iRow&, 30) Then
                sBK$ = CellString(WkBS, iRow&, 2)
                MsgTxt$ = "There is an inbalance between BS and BK regarding" & Chr(13)
                MsgTxt$ = MsgTxt$ & sBK$ & ">" & CellString(WkBS, iRow, 4) & "." & Chr(13)
                MsgTxt$ = MsgTxt$ & "Will proceed to Verification by row to identify the cause."
                cBKCD% = Col_InArray("BK", "BKCD")
                bkRow& = RowArray(0, 0) + 1
                Do While bkRow& <= RowArray(1, 0)
                    If CellString(WkBK, bkRow&, cBKCD%) = sBK$ Then
                        Exit Do
                    End If
                    bkRow& = bkRow& + 1
                Loop
                setOptState ("BK")
                SelectCell bkRow&, cBKCD%
                MsgTtl$ = "Test of agreement of BS and BK balances"
```

```
                resp% = MsgBox(MsgTxt$, 16, MsgTtl$)
                oDTDlg.endExecute()
                VeriByWS bkRow&, "WsCalc"
                End
            End If
            iRow = iRow + 1
        Loop
        BSAudLN%=getFSRow%(WkBS, "ENDLN", ActCol) + 1
        SelectCell  BSAudLN%, BKCol
        VeriV& = 0
        For i% = BKCol To RMCol
            VeriV& = VeriV& + CellValue(WkBS, BSAudLN%, i%)
        Next i%
        If VeriV& <> 0 then
            if CellValue(WkBS, BSAudLN%, BKCol) <> 0 Then
                setOptState("BK")
                VeriByWS dbMonthTop&, 276
            end if
            if CellValue(WkBS, BSAudLN%, JVCol) <> 0 Then
                setOptState("JV")
                VeriByWS dbMonthTop&, 276
            end if
            if CellValue(WkBS, BSAudLN%, APCol) <> 0 Then
                setOptState("AP")
                VeriByWS dbMonthTop&, 276
            end if
            if  ARUse Then
                If CellValue(WkBS, BSAudLN%, ARCol) <> 0 Then
                    setOptState("AR")
                    VeriByWS dbMonthTop&, 276
                End If
            end if
            if  MWUse Then
                If CellValue(WkBS, BSAudLN%, MWCol) <> 0 Then
                    setOptState("MW")
                    VeriByWS dbMonthTop&, 276
                End If
            end If
        End If
        Rem ----------------------------------------------------------------
        If IsMultiDP then
            VeriDept()
        End If
    End Sub
    Rem ==================(Verification after BS/PL Preparation)==================
    Sub VeriDept()
        Dim VD(BKCol To RMCol) As Variant
        Dim i%, j%, DPTAudLN%
        DPTAudLN%=getFSRow%(WkPL, UCase(DPTAudCD), CFCol-1)
        For i% = BKCol To RMCol
            for j% = 1 To NumDpt
                VD(i%) = VD(i%) + CellValue(WkDpt(j%), DPTAudLN%, i%)
            next j%
            VD(i%) = CellValue(WkPL, DPTAudLN%, i%) - VD(i%)
        Next i%
        If VD(BKCol) <> 0 Then
            setOptState("BK")
            VeriByWS dbMonthTop&, 276
        ElseIf VD(JVCol) <> 0 Then
            setOptState("JV")
            VeriByWS dbMonthTop&, 276
        ElseIf VD(APCol) <> 0 Then
            setOptState("AP")
            VeriByWS dbMonthTop&, 276
        End If
        If ARUse then
            if VD(ARCol) <> 0 Then
                setOptState("AR")
                VeriByWS dbMonthTop&, 276
            end if
```

```
491                 resp% = MsgBox(MsgTxt$, 16, MsgTtl$)
492                 oDTDlg.Hide
493                 VeriByWS bkRow&, "WsCalc"
494                 End
495             End If
496             iRow& = iRow& + 1
497         Loop
498         BSAudLN% = getFSRow%(WkBS, "ENDLN", ActCol) + 1
499         SelectCell BSAudLN%, BKCol
500         VeriV& = 0
501         For i% = BKCol To RMCol
502             VeriV& = VeriV& + CellValue(WkBS, BSAudLN%, i%)
503         Next i%
504         If VeriV& <> 0 Then
505             If CellValue(WkBS, BSAudLN%, BKCol) <> 0 Then
506                 setOptState ("BK")
507                 VeriByWS dbMonthTop&, 276
508             End If
509             If CellValue(WkBS, BSAudLN%, JVCol) <> 0 Then
510                 setOptState ("JV")
511                 VeriByWS dbMonthTop&, 276
512             End If
513             If CellValue(WkBS, BSAudLN%, APCol) <> 0 Then
514                 setOptState ("AP")
515                 VeriByWS dbMonthTop&, 276
516             End If
517             If ARUse Then
518                 If CellValue(WkBS, BSAudLN%, ARCol) <> 0 Then
519                     setOptState ("AR")
520                     VeriByWS dbMonthTop&, 276
521                 End If
522             End If
523             If MWUse Then
524                 If CellValue(WkBS, BSAudLN%, MWCol) <> 0 Then
525                     setOptState ("MW")
526                     VeriByWS dbMonthTop&, 276
527                 End If
528             End If
529         End If
530         Rem ----------------------------------------------------------------
531         If IsMultiDP Then
532             VeriDept
533         End If
534     End Sub
535     Rem ==================(Verification after BS/PL Preparation)================
536     Sub VeriDept()
537         Dim VD(BKCol To RMCol) As Variant
538         Dim i%, j%, DPTAudLN%
539         DPTAudLN% = getFSRow%(WkPL, UCase(DPTAudCD), CFCol - 1)
540         For i% = BKCol To RMCol
541             For j% = 1 To NumDpt
542                 VD(i%) = VD(i%) + CellValue(WkDpt(j%), DPTAudLN%, i%)
543             Next j%
544             VD(i%) = CellValue(WkPL, DPTAudLN%, i%) - VD(i%)
545         Next i%
546         If VD(BKCol) <> 0 Then
547             setOptState ("BK")
548             VeriByWS dbMonthTop&, 276
549         ElseIf VD(JVCol) <> 0 Then
550             setOptState ("JV")
551             VeriByWS dbMonthTop&, 276
552         ElseIf VD(APCol) <> 0 Then
553             setOptState ("AP")
554             VeriByWS dbMonthTop&, 276
555         End If
556         If ARUse Then
557             If VD(ARCol) <> 0 Then
558                 setOptState ("AR")
559                 VeriByWS dbMonthTop&, 276
560             End If
```

```
    End If
    If MWUse then
        if VD(MWCol) <> 0 Then
            setOptState("MW")
            VeriByWS dbMonthTop&, 276
        end if
    End If
End Sub
```

```
        End If
        If MWUse Then
            If VD(MWCol) <> 0 Then
                setOptState ("MW")
                VeriByWS dbMonthTop&, 276
            End If
        End If
End Sub
```