*'The main TestEvent Macro to call. It's used to call one of three*

*'separate Test Macros based on the measurement mode of the PMM.*

*'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

**Function** **FlukeCalSledCalibrationAdjust**(iT, iL, iC, iP, cTest, cConfig)

*' Determine which macro to run. There are five Meas Mode codes:*

*'0 = Absolute*

*'1 = Gauge*

*'2 = Abs by Atms*

*'3 = Differential (Bi-Directional)*

*'4 = ???*

*'5 = Negative Gauge*

**Select Case** cTest.TestPrsMeasMode

**Case** 0:

**Call** **FlukeCalSledAbsolute**(iT, iL, iC, iP, cTest, cConfig)

**Case** 1:

**Call** **FlukeCalSledGauge**(iT, iL, iC, iP, cTest, cConfig)

**Case** 3:

**Call** **FlukeCalSledDifferential**(iT, iL, iC, iP, cTest, cConfig)

**End Select**

**End Function**

**Function** **FlukeCalSled\_Dwell**(dwell)

cDebug.LogStatus "Dwell for " & dwell & "s"

tStart = timer

**Do**

cCOMPASS.**TimeDelay** 1

td = **CInt**(dwell-**time\_Difference**(tStart))

cCOMPASS.StatusDisplay "Dwell..." & td

**If** cCOMPASS.SystemAbort **Then** **Exit** **Function**

**If** td <= 0 **Then** **Exit** **Do**

**Loop** **Until** **False**

**End Function**

**Function** FlukeCalSled\_WaitForReady(isZero, timeout)

cDebug.LogStatus "Wait for Ready: timeout=" & timeout

tStart = Timer

*'The Do Loop tells COMPASS to look at the logic output of the UseReady method. The logic True / False*

*'can be queried by using the .Ready parameter. The return from cConfig.SetPres(1).Ready is either TRUE or FALSE.*

**Do**

cCOMPASS.**TimeDelay** 2 *'delay controller*

cCOMPASS.**StatusDisplay** process & ": Waiting for Ready.........."

**If** cCOMPASS.SystemAbort **Then** **Exit** **Function**

**If** **Time\_Difference**(tStart) > timeout **Then**

cDebug.LogStatus "\*\*\*\*\*\*\*TIMEOUT WAITING for Ready\*\*\*\*"

cCOMPASS.**StatusDisplay** process & ": ............TIMEOUT Waiting for Ready.........."

cCOMPASS.**TimeDelay** 5 *'delay to show above message*

**If** (cCOMPASS.cConfig.DUTPrs(1).RangeMain.MeasMode = 0) And (isZero = 1) **Then**

cDebug.LogStatus "Timeout for absolute 0, as good as it will get"

**Exit** **Do** *'absolute 0, as good as possible*

**End If**

**Exit** **Function**

**End If**

*'If the SetPrs device supports the UseReady concept then loop until the .Ready parameter becomes TRUE or,*

*'If the SetPrs device doesn't support UseReady then look at if the RefPrs device does. The presumption*

*'is that either the controller or the reference device will support the UseReady method. Exit the loop*

*'once one of these two devices indicates Ready.*

**If** cConfig.SetPrs(1).RangeMain.Useready **Then**

cDebug.LogStatus "Wait for ready...SetPrs.Ready=" & cConfig.SetPrs(1).Ready

**If** cConfig.SetPrs(1).Ready **Then** **Exit** **Do**

**Else**

cDebug.LogStatus "Wait for ready...RefPrs.Ready=" & cConfig.RefPrs(1).Ready

**If** cConfig.RefPrs(1).Ready **Then** **Exit** **Do**

**End If**

**Loop**

FlukeCalSled\_WaitForReady = **True**

**End Function**

**Function** **FlukeCalSled\_Date**()

yy = right(Year(**Date**),2)

dd = Right(**String**(2, "0") & Day(**Date**), 2)

mm = Right(**String**(2, "0") & Month(**Date**), 2)

**FlukeCalSled\_Date** = mm & "/" & dd & "/" & yy

**End Function**