



# dataTaker Automatic Weather Station Code Examples



# Table of Contents

Introduction	2
<hr/>	
<b>9 am reporting of data</b>	<b>3</b>
<hr/>	
DT80 V9.08 Firmware and latter	3
<hr/>	
Wind gust calculation	3
<hr/>	
<b>Wind directions calculations</b>	<b>3</b>
<hr/>	
Bureau of Meteorology method	3
<hr/>	
Wind speed and direction using Vector functions	5
<hr/>	
Wind speed, direction and Yamartino method of standard deviation of wind speed	6
<hr/>	
Calculating RH when measuring wet bulb and dry bulb	7
<hr/>	
Calculating rain fall rate with a tipping bucket rain gauge	8
<hr/>	
Calculating rain fall rate with a tipping bucket rain gauge and allowing sleep mode	8
<hr/>	

# Introduction

This example demonstrates how to use DT 80 series of logger for Automatic weather Stations (AWS) applications. These examples demonstrate various functions and provide code to use as a starting point for your application. This code uses the methods described in the Bureau of Meteorology standards.

Program for monitoring Automatic weather Stations (AWS); Wind gust, Wind Speed, Wind Direction, RH, Rain Gauge calculation using DT80 Series of Loggers.

Hardware: DT80 Series 1,2,3 & 4 loggers and CEM 20

For more information visit [thermofisher.com/datataker](https://www.thermofisher.com/datataker)

# 9 am reporting of data

## DT80 V9.08 Firmware and latter

Version 9 firmware introduce CRON like schedule triggering which greatly simplifies this task. The schedule can now be triggered at any time of the day or month, etc. with a simple CRON like expression. For more details, please refer to the section on Schedule Triggers in the DT80 range user manual.

## Wind gust calculation

This program calculates the 3 second wind gust by using a 3 second moving average and reports the maximum wind gust every 10 minutes. The sensor used in this example was a R. M. Young Wind Monitor, but it is basically applicable to other pulse or frequency output wind sensors.

```
BEGIN"Gust"
S1=0,0.1904,0,1"Kts"

RA10M          'Set the reporting speed for the calculated statistical data
5CV("Gust~Kts")      'Reprot the 3 second maximum gust speed in last 10 minutes
10CV("Ave~Kts")=6CV/7CV 'Report the average wind speed for the last 10 minutes
5..7CV(W)=0          'Clear the variables for the above calculations.

RB1S          'Set the sample speed for the calculated statistical data
3CV(W)=2CV      'Wind speed at T-3 seconds
2CV(W)=1CV      'Wind speed at T-2 seconds
1HSC(R,RS,LT,=1CV,S1,W) 'Read the wind speed for the last second
4CV(W)=(1CV+2CV+3CV)/3 'Calculate the average 3 second wind speed.
5CV(W)=5CV*(4CV<=5CV)+4CV*(4CV>5CV) 'Keep the Peak wind speed till 5CV is reset (From schedule A)
6CV(W)=6CV+1CV  'Calculate sum of the wind speed.
7CV(W)=7CV+1    'Calculate the number of samples

END
```

## Wind directions calculations

### Bureau of Meteorology method

This code uses the methods described in the Bureau of Meteorology standards.

Program for monitoring a wind set and calculating;

- mean wind speed
- mean wind direction
- standard deviation of wind speed
- standard deviation of wind direction

The program assumes that the wind speed and wind direction sensors are; anemometer with output of 0–1000 mV for 0–50 m/s windspeed wind vane with output of 0–1000 mV for 0–360 deg wind direction. Other types of wind speed and wind direction sensors are easily supported by changing the sensor calibration polynomials and the sensor signal types below.

```

BEGIN
/D /T 'Turn on automatic time and date stamp of data.

'Define the wind sensor calibrations (Polynomials)
Y1=0,0.050"m/s"           ' Wind speed      0-1000mV = 0-50 m/s
Y2=0,0.360"Deg"           ' Wind direction 0-1000mV = 0-360 deg

'Define units text for calculations (Polynomials with unity scaling)
Y3=0,1"m/s"               ' Units text for wind speed
Y4=0,1"Deg"               ' Units text for wind direction

'Clear accumulators used to sum intermediate data (Channel variables)
10..16CV(W)=0

'Schedule to scan wind set and sum intermediate data every 5 seconds
RA5S
  1V(Y1,=1CV,W)           ' Scan wind speed and save
  2V(Y2,=2CV,W)           ' Scan wind direction and save
  2CV(W)=2CV/57.29        ' Convert wind direction to radians for Dataloader trig functions
  10CV(W)=10CV+COS(2CV)   ' Sum of COS(wd)
  11CV(W)=11CV+SIN(2CV)   ' Sum of SIN(wd)
  12CV(W)=12CV+1CV        ' Sum of ws
  13CV(W)=13CV+(1CV*1CV)  ' Sum of ws * ws
  14CV(W)=14CV+(1CV*COS(2CV)) ' Sum of ws * COS(wd)
  15CV(W)=15CV+(1CV*SIN(2CV)) ' Sum of ws * SIN(wd)
  16CV(W)=16CV+1.0        ' Number of scans
  'Schedule to calculate wind data, and log and return the data every minute

RB1M
  20CV(W)=14CV/16CV       ' Mean ws * COS(wd)
  21CV(W)=15CV/16CV       ' Mean ws * SIN(wd)
  'Calculate and log/display/return mean wind speed
  22CV(W)=SQRT((20CV*20CV)+(21CV*21CV))
  22CV("Mean Wind Speed",Y3,FF2)
  'Calculate and log/display/return mean wind direction
  23CV(W)=ATAN(21CV/20CV)*57.29
  'Find quadrant for mean wind direction
  23CV(W)=23CV+((20CV>0)AND(21CV<0))*360   ' 4th quadrant : +Cos -Sin
  23CV(W)=23CV+((20CV<0)AND(21CV<0))*180   ' 3rd quadrant : -Cos -Sin
  23CV(W)=23CV+((20CV<0)AND(21CV>0))*180   ' 2nd quadrant : -Cos +Sin
  23CV(W)=23CV+((20CV>0)AND(21CV>0))*0     ' 1st quadrant : +Cos +Sin
  23CV(W)=23CV-(12CV=0)*(23CV+1)           ' No wind speed, return -1.0
  23CV("Mean Wind Dirn",Y4,FF2)
  'Calculate and log/display/return standard deviation of wind speed
  24CV(W)=13CV-((12CV*12CV)/16CV)
  24CV(W)=SQRT(24CV/16CV)
  24CV("SD Wind Speed",Y3,FF2)
  'Calculate and log/display/return standard deviation of wind direction
  25CV(W)=10CV/16CV
  26CV(W)=11CV/16CV
  27CV(W)=SQRT((25CV*25CV)+(26CV*26CV))
  27CV(W)=SQRT(-2.0*LN(27CV))*57.29
  27CV("SD Wind Dirn",Y4,FF2)
  'Clear accumulators
  10..16CV(W)=0
LOGON
END

```

## Wind speed and direction using Vector functions

Version 9.08 firmware for the DT80 range introduces vector maths that simplified the above calculations. This code demonstrates the use of vector maths and the new in line C type If Then Else statement.

This code is for wind speed and wind direction only.

```
BEGIN "WS&WD"
'=====
'
' Wind speed and direction calculation using Vector functions
'
' For DT8x range of DataTaker data loggers
'
' Notes:
' Requires 9.08 firmware or above.
' Version 9.08 firmware introduces vector mathematics which
' greatly simplifies the calculation of wind speed and direction.
'
' This code also make use of the new C type inline If Then Else statement
' That was also added in V9.08 firmware
'
' For assistance in using this code please contact
' DataTaker technical support or your DataTaker distributor.
'=====

'=====
'Change the spans to suit your sensor calibrations
'=====
'Define the wind sensor calibrations (Polynomials)
S1=0,50,0,1000"m/s"          ' Wind speed      0-1000mV = 0-50 m/s
S2=0,360,0,1000"deg"        ' Wind direction 0-1000mV = 0-360 deg

'Clear accumulators used to sum statistical samples
1..3CV(W)=0

'=====
'Schedule A scans wind set and sum intermediate data every 5 seconds
'
' Note: All channels are working channels so no data is being logged.
' If you want data to be logged for diagnostic purposes remove the
' W channel option.
'=====

RA"Sample"5S
    'Change channel type to mach out put of Wind Speed sensor
    1V("Raw_WS~m/s",S1,W)                                     ' Read
instantaneous wind speed
    Calc("Inst_WS~m/s",W)=(&Raw_WS<0)?0:&Raw_WS              ' If WS < 0 then WS = 0 else WS
= WS

    'Change channel type to mach out put of Wind Direction sensor
    2V("Inst_WD~deg",S2,W)                                     ' Read
instantaneous wind direction

    ' Vector components for wind speed and wind direction.
    1CV("sumX",W)=1CV+MAGDIR2X(&Inst_WS,D2R(&Inst_WD))        ' Sum X component of vector
    2CV("sumY",W)=2CV+MAGDIR2Y(&Inst_WS,D2R(&Inst_WD))        ' Sum Y component of vector
```

```

3CV("Num_Samp",W)=3CV+1                                     ' Count number
of samples

'Schedule B calculates wind data, and log and return the data every 6 minutes
RB"Report"("b:",DATA:OV:62D)6M
    'Calculate Wind speed and wind direction from vector components.
    Calc("Ave_WS~m/s")=XY2MAG(&sumX,&sumY)/&Num_Samp           ' Calculate average wind speed
    Calc("Ave_WD~deg")=(&Ave_WS>0)?R2D(XY2DIR(&sumX,&sumY)):-1 ' Calculate average wind direction. If WS = 0
then WD = -1
    1..3CV(W)=0

LogOnB
END

```

## Wind speed, direction and Yamartino method of standard deviation of wind speed

This example builds on the code above to include Standard deviation of wind direction by the Yamartino method.

```

BEGIN "WSWSD"
'=====
'
'    Wind speed and direction calculation using Vector functions
'    an Yamartino method to calculate the standard deviation of wind speed.
'
'    For DT8x range of DataTaker data loggers
'
'    Notes:
'    Requires 9.08 firmware or above.
'
'    For assistance in using this code please contact
'    DataTaker technical support or your DataTaker distributor.
'=====

'=====
'Change the spans to suit your sensor calibrations
'=====
'Define the wind sensor calibrations (Polynomials)
S1=0,50,0,1000"m/s"           ' Wind speed      0-1000mV = 0-50 m/s
S2=0,360,0,1000"deg"         ' Wind direction 0-1000mV = 0-360 deg

'Clear accumulators used to sum statistical samples
1..3CV(W)=0

'=====
'Schedule A scans wind set and sum intermediate data every 5 seconds
'
'    Note: All channels are working channels so no data is being logged.
'    If you want data to be logged for diagnostic purposes remove the
'    W channel option.
'=====

RA"Sample"5S
    'Change channel type to mach out put of Wind Speed sensor
    1V("Raw_WS~m/s",S1,W)                                     ' Read instantaneous wind speed
    Calc("Inst_WS~m/s",W)=(&Raw_WS<0)?0:&Raw_WS             ' If WS < 0 then WS = 0 else WS
= WS

```

```

        'Change channel type to match out put of Wind Direction sensor
2V("Inst_WD~deg",S2,W)                                ' Read instantaneous wind
direction

        ' Vector components for wind speed and wind direction.
1CV("sumX",W)=1CV+MAGDIR2X(&Inst_WS,D2R(&Inst_WD))      ' Sum X component of vector
2CV("sumY",W)=2CV+MAGDIR2Y(&Inst_WS,D2R(&Inst_WD))      ' Sum Y component of vector
3CV("Num_Samp",W)=3CV+1                                  ' Count number of samples

        ' Components for standard deviation of wind direction (Yamartino method)
4CV("Sum_Cos_Theta",W)=4CV+Cos(D2R(&Inst_WD))           ' Sum of COS(wd)
5CV("Sum_Sin_Theta",W)=5CV+Sin(D2R(&Inst_WD))           ' Sum of COS(wd)

'Schedule B calculates wind data, and log and return the data every 6 minutes
RB"Report"("b:",DATA:0V:62D)6M
        'Calculate Wind speed and wind direction from vector components.
Calc("Ave_WS~m/s")=XY2MAG(&sumX,&sumY)/&Num_Samp          ' Calculate average wind speed
Calc("Ave_WD~deg")=(&Ave_WS>0)?R2D(XY2DIR(&sumX,&sumY)):-1 ' Calculate average wind direction. If WS = 0
then WD = -1

        ' Calculate standard deviation of Wind speed. (Yamartino method)
Calc("E",W)=Sqrt(1-((&Sum_Sin_Theta/&Num_Samp)^2+(&Sum_Cos_Theta/&Num_Samp)^2)) ' Calculate E component
Calc("SD_WD",W)=R2D(ASin(&E)*(1+0.1547*&E^3))             ' Calculate standard deviation
of wind direction.
Calc("Std_Dev_Wnd_Dir")=(&Num_Samp<10)?999:&SD_WD        ' If number of samples is less than Then 10 SD
Wind = 999

        1..5CV(W)=0      'Clear counters

LogOnB
END

```

## Calculating RH when measuring wet bulb and dry bulb

This calculation could be used on ANY of the DataTaker loggers.

```

BEGIN
RA1S
1TK(=1CV,"DryT")
2TK(=2CV,"WetT")
3CV("Dryabs",W)=1CV+273.16
4CV("Wetabs",W)=2CV+273.16
5CV(W)=-2892.3693/3CV-2.892736*LOG(3CV)/LOG(10)-4.9369728/1000*3CV+5.606905/((10^6)*(3CV^2))-4.645869/((10^9)*(3CV^3))+19.4260451+3.7874/((10^12)*(3CV^4))
6CV(W)=-2892.3693/4CV-2.892736*LOG(4CV)/LOG(10)-4.9369728/1000*4CV+5.606905/((10^6)*(4CV^2))-4.645869/((10^9)*(4CV^3))+19.4260451+3.7874/((10^12)*(4CV^4))
7CV("e",W)=10^6CV-0.66*(1CV-2CV)*(1+1.15/1000*2CV)
8CV(W)=(100*7CV/(10^5CV))*(2CV>0)+(2CV<=0)
8CV("RH")=((1CV>=0)AND(1CV<99))*8CV+((1CV<0)OR(1CV>=99))
END

```

## Calculating rain fall rate with a tipping bucket rain gauge

To calculate the rain fall rate, we need to know the time when the bucket tipped. The digital inputs can be used as counters and to trigger a schedule to execute.

This code uses digital input 1 to trigger the schedule and count the number of tips of the bucket. The data is date and time stamped with the /D/T switches.

```
BEGIN"RainRate"  
Y1=0,0.1"mm" 'Rain gauge calibration (0.1 mm per tip)  
/D 'Date stamp the data  
/T 'Time stamp the data  
RA1-E 'Trigger schedule A when the bucket tips  
    1C(Y1) 'Read the counter  
END
```

## Calculating rain fall rate with a tipping bucket rain gauge and allowing sleep mode

The DT500 and DT80 both have a wake terminal. Then this terminal is pulled low the DataTaker will wake up from sleep mode. In this example, the WK terminal is linked to 1 High Speed Counter (1HSC). When the High Speed Counter changes from 0 to 1, the schedule is triggered and the bucket volume is accumulated in 1 CV.

```
BEGIN  
/D  
/T  
1CV=0  
RA1HSC(1)  
    1CV("Rain rate")=1CV+0.1  
END
```



---

**In Australia:**

For customer service, call 1300-735-292  
To email an order, [ordersau@thermofisher.com](mailto:ordersau@thermofisher.com)

**In New Zealand:**

For customer service, call 0800-933-966  
To email an order, [ordersnz@thermofisher.com](mailto:ordersnz@thermofisher.com)

Find out more at [thermofisher.com/datataker](https://thermofisher.com/datataker)

**ThermoFisher**  
S C I E N T I F I C