Baker Hall Eddy Covariance Flux Tower

Construction and User Manual

B.J. Baule, Chase Brooke, Cheyenne Lei, Gabriela Shirkey

GEO 892: Micrometeorological Instrumentation
Instructor: Dr. Jiquan Chen | Fall 2017

Outline

Project Overview

In Fall 2017, as part of a class focused on Micrometeorological Instrumentation taught by Dr. Jiquan Chen, an eddy covariance flux tower was constructed. This document serves as a basic outline to the design, construction, siting, and the datalogger program pertaining to the flux tower as of Fall 2017.

The station is design to be a semi-permanent installation, on the roof of Baker Hall, which is located on on the Michigan State University campus (Figure 1). The station sits above all surrounding structures and obstructions on the second tier of the roof. The surface of the roof is gravel over a tarred roof surface. The area surrounding Baker Hall is a suburban mosaic consisting of the Michigan State University Campus and surrounding East Lansing neighborhoods. The Red Cedar River is located approximately 100 meters to the south of Baker Hall. The station is at the highest point (approximately 30 meters above the ground) in the vicinity, so obstruction from taller objects is not a major concern with station citing.

Figure 1. Baker Hall on the Michigan State University campus. The approximate location of the station is marked with a red ‘X’.

Figure 2. The completed EC Flux Tower installed on the roof of Baker Hall.

Station Construction

The frame of the station consists of a triangular base constructed of three eight-foot lengths of two-inch galvanized steel pipe (Figure 2). The base is attached to a ten-foot triangular galvanized steel tower, which makes up the primary tower structure, by steel u-bolts through two holes that were drilled in each base pipe (Figure 3). The tower is secured to the triangular base by three lengths of guide wire, extending from the top of the ten foot tower to the end of each side of the triangular base. Two sandbags, weighing 50 pounds each, are placed at each end of the feet, with one additional sand bag placed on the center plate of the ten foot tower (Figure 2).

The structure is protected by a lightning rod that is grounded through copper conduit to the structure of Baker Hall. The station is powered by two deep cycle 12 volt marine batteries, wired in parallel, which are attached to a battery charger attached to the 120 volt alternating current supplied by Baker Hall (Figure 4). All instrumentation is attached to the tower itself or a 10 ft galvanized steel cross member attached to the body of the tower. A figure showing all instruments attached to the station is shown in Figure 5. A complete parts list is given in the Appendix of the document.

All instruments attached to the station are shown in Figure 5. A table of a subset of variables along with these associated instruments and their frequency of outputs is shown in Table 1 below. A complete part list for tower infrastructure is found in Appendix E. The full CR5000 program for this flux tower is available in Appendix F.

Figure 3. The final base construction of the tower. Note that each length of galvanized pipe that makes up the base is attached via u-bolt through two holes in each pipe that were drilled by team members.

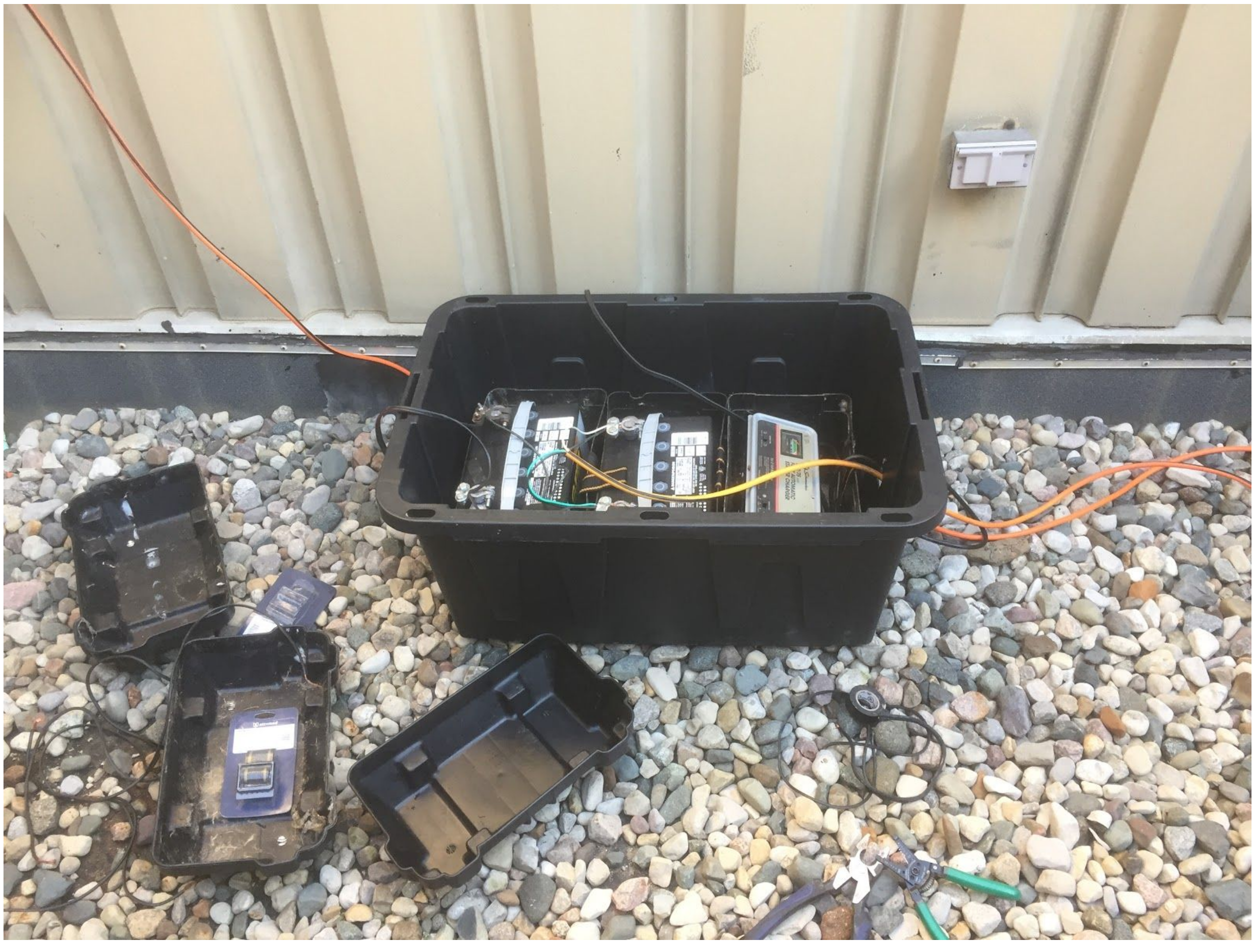


Figure 4. 12V deep cycle marine batteries, wired in parallel, along with battery charger and 120V power supply. Batteries and charger are enclosed in battery cases, inside of a weatherproof tote.

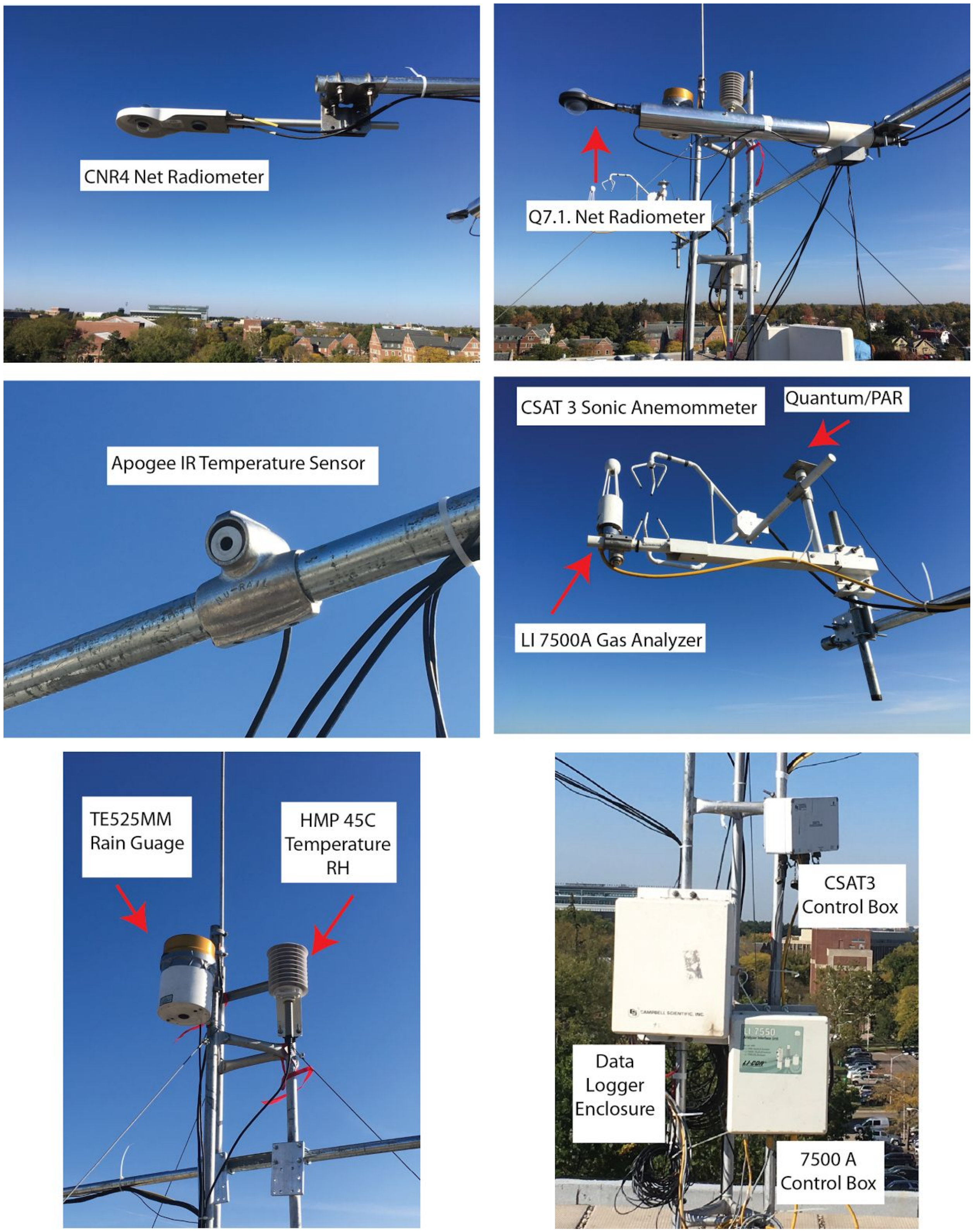


Figure 5. Instrumentation attached to the the Baker Hall EC Flux Tower. Enclosed in the data logger enclosure is a Campbell Scientific CR5000 Data Logger.

Table 1. Table of variables by instrument and observational frequency output by tower program.

Variable	**Instrument**	**Frequency**
Air Temperature	HMP45C	30 Minute Average
Relative Humidity	HMP45C	30 Minutes Average
Liquid Precipitation	TE525MM Rain Gauge	30 Minute Total
Net Radiation	CNR4 Net Radiometer Q7.1 Net Radiometer	30 Minute Average
Shortwave Radiation Downward	CNR4 Net Radiometer	30 Minute Average
Shortwave Radiation Upward	CNR4 Net Radiometer	30 Minute Average
Longwave Radiation Downward	CNR4 Net Radiometer	30 Minute Average
Longwave Radiation Upward	CNR4 Net Radiometer	30 Minute Average
Net Shortwave Radiation	Q7.1 Net Radiometer	30 Minute Average
Net Longwave Radiation	Q7.1 Net Radiometer	30 Minute Average
Wind Velocities (zonal, meridional, and vertical)	CSAT3 Sonic Anemometer	1/10 Second 30 Minute Average
Wind Speed	CSAT3 Sonic Anemometer	30 Minute Average
Wind Direction	CSAT3 Sonic Anemometer	30 Minute Average/Snapshot
Water Vapor Flux	LI5000A Gas Analyzer	1/10 Second 30 Minute Average
Carbon Dioxide Flux	LI5000A Gas Analyzer	1/10 Second 30 Minute Average
Surface Temperature	Apogee IR Temperature	30 Minute Average

Issues and Obstacles

After gathering the sensors and equipment for the tower, we conducted a test assembly in an easily accessible area. The purpose of this was twofold: to identify any issues or problems in the assembly process, as well as to check the functionality of sensors and the datalogger. In regards to the physical assembly of the tower, we ran into multiple small issues. First, we were short on mounting plates which fit the brackets of the sensors, and while fabricating replacement ones on-site was difficult, it was doable. Second, we came across several sensor issues where the sensor was malfunctioning or was wired incorrectly into the datalogger.

From this experience in assembling a flux tower, we highly recommend a trial setup of a station prior to final installation and assembly in its terminal location. In doing so, we can readily identify sensor issues and programming malfunctions in a readily-accessible location for diagnosis and repair. If not possible, we highly suggest some sort of testing over a set period of time to identify potential issues and make the final installation proceed smoothly.

LoggerNet and Data Collection

LoggerNet is the main support software package used to process flux data from Campbell Scientific dataloggers. It allows a customizable interface with the datalogger which offers many benefits to the user including:

- Custom datalogger programs using Short Cut, Edlog, or CRBasic Editor.
- Access to multiple types of dataloggers through different types of setups.
- Conversion of loggernet data files into ASCII, CSV, and XML for reading into Excel and R Statistical Package.
- Collection and observation of data through real-time display, or scheduled manual retrieval.
- Supports communication, and data retrieval between dataloggers, as well as a PC.
- Preliminary analysis of data through integrated programs to create and view graphs.

To view and analyze Baker Tower's flux data, please download LoggerNet software from https://www.campbellsci.com/loggernet. Examples of

Connect to a Datalogger

To initially connect to a datalogger, the user must first setup the datalogger to interact with LoggerNet (Fig 1). This can be completed by accessing the 'Setup' program from the 'Main' menu. Clicking 'Add' in the 'setup' screen will allow you to select a datalogger from the drop-down list. This setup screen will also allow you to customize your datalogger by selecting your type of connection (Direct Connect via cable to the data logger, networked computer, phone model, etc), schedule a collection, send a completed program, and test the communication to confirm functionality. More information can be viewed at: https://www.campbellsci.com/videos/overview-and-software-setup-quickstart-part-1.

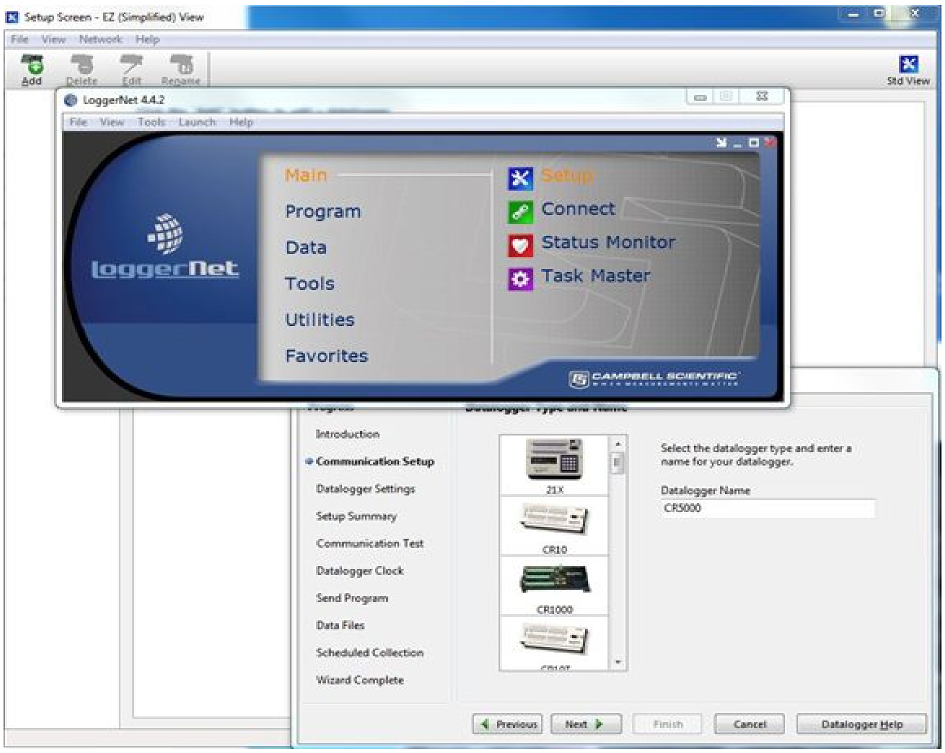


Figure 6. Setting up a datalogger in LoggerNet.

Create and Input a Program

Creating a program is made relatively through LoggerNet. This can be completed by accessing multiple options within the 'Program' menu, based on your preferences (Fig 2). Short Cut creates simple programs using preconfigured sensor wiring and diagrams. Programs are created in four steps: 1) select the datalogger model, 2) select the sensors and devices to be connected to the datalogger, 3) select the data to be stored in the datalogger, and 4) save the datalogger program. Short Cut also generates a wiring diagram for connecting your sensors and devices to the datalogger. This is especially useful for other dataloggers and for users with limited experience with writing code. Edlog and CRBasics are both more advanced programming languages for the creation of a datalogger program. It is intended for use by more experienced programmers who needs more flexibility than what is found in the Short Cut Program generator. These programs allows the creation of extensive code for specific sensors not found in Short Cut, as well as the more dedicated creation of tables for data analysis.

More information can be retrieved here for:

- Short Cut: https://www.campbellsci.com/videos/programming-with-short-cut-quickstart-part-2
- Edlog: https://www.campbellsci.com/blog/edit-input-locations-in-edlog
- CRBasics: https://www.campbellsci.com/crbasiceditor

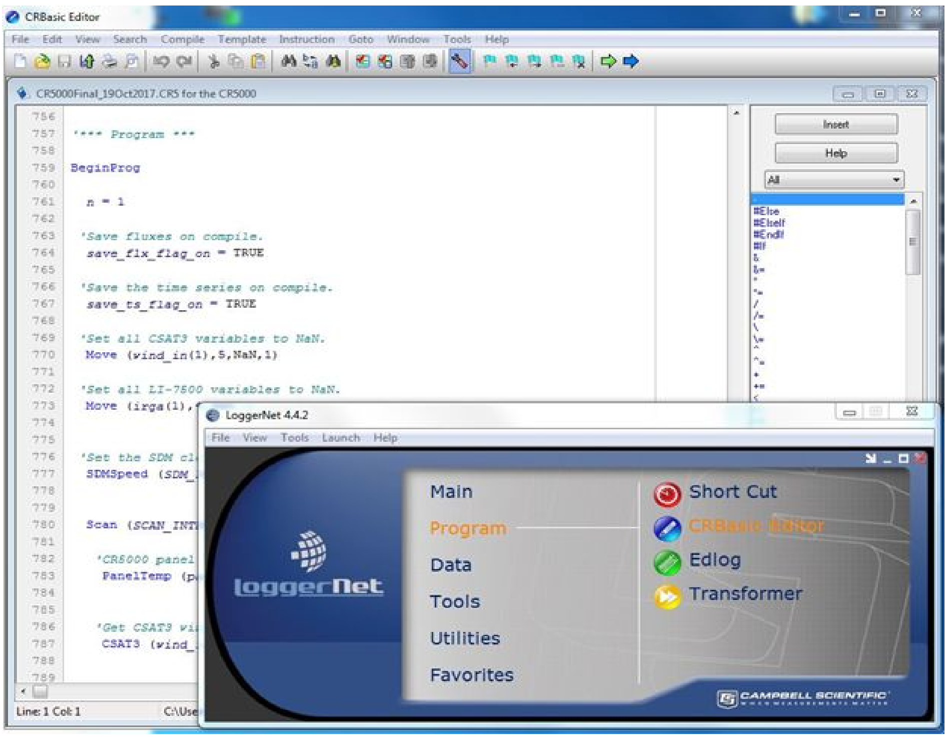


Figure 7. Creating a program in CRBasics.

Access, Retrieve, Review and Analyze Data

To access data from a datalogger requires the user to access the ‘Connect’ program from the ‘Main’ menu (See Appendix E, Figure 3). In the ‘Stations’ screen, you will be able to select which datalogger you will want to access data from. Once selected, the user can choose a variety of options, including connect in real-time to view data either by a number display or in a graphed format, collect all data on the datalogger at that current point in time, and analyze the station’s status for troubleshooting purposes. All options allow the user to view all currently measured variables, or select a specific variable to view in greater detail.

Data collected from the datalogger can be viewed within LoggerNet through the ‘View Pro’ program in the ‘Data’ menu (Fig 4). This software allows the user to input a file from the datalogger to view data retrieved. Multiple options to view the data includes graphs, XY plots, and histograms.

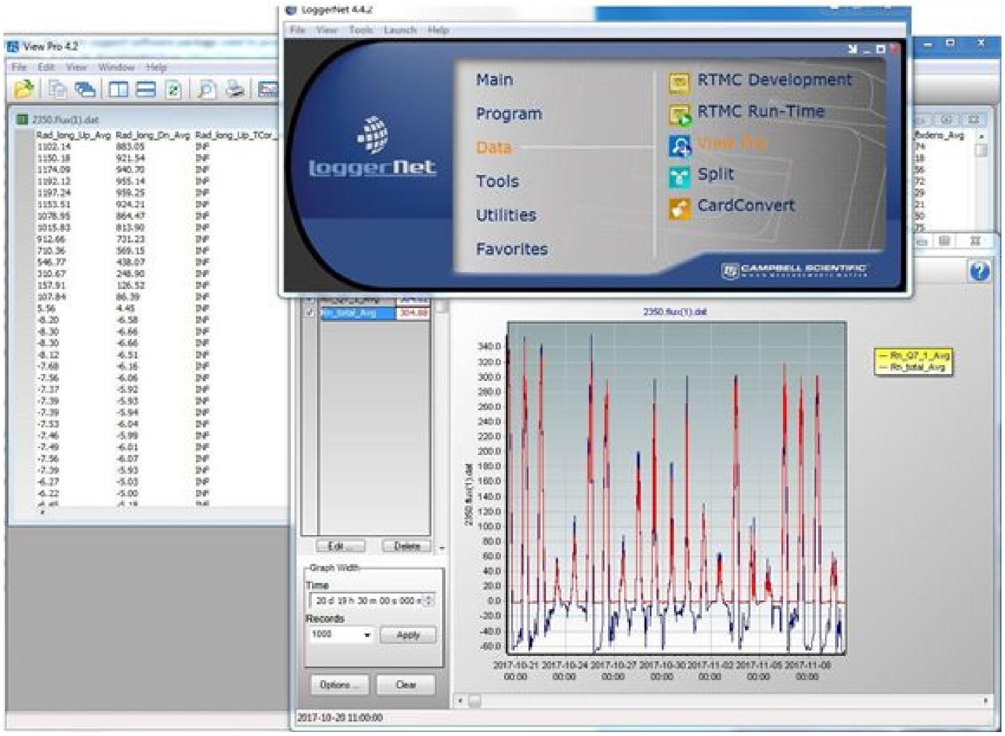


Figure 8. Using View Pro in LoggerNet to access, graph, and analyze data

To export the datalogger file to a third party software (e.g. Excel, R) for further analysis, the ‘Card Convert’ option can be accessed from the ‘Data’ menu (Fig 5). After selecting the data from the ‘Card Drive’, the user can then modify the file into multiple formats including ASCII data tables and CSVs from the ‘Destination Option’ menu. Importing it into third party programs is then possible after the successful conversion. More information can also be viewed at [https://www.campbellsci.com/videos/use-loggernet-to-send-a-program-and-collect-data-quick start-part-3](https://www.campbellsci.com/videos/use-loggernet-to-send-a-program-and-collect-data-quickstart-part-3).

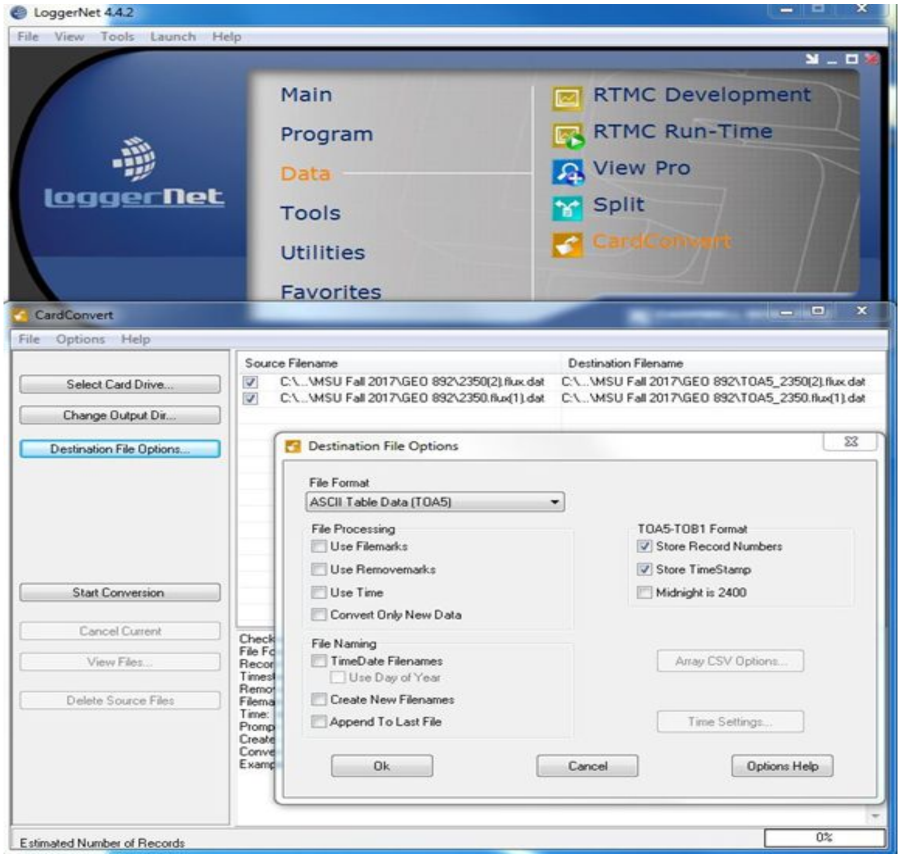


Figure 9. Converting LoggerNet data files for use in third-party programs.

EdiRe

EdiRe is a software package created by the University of Edinburg. EdiRe can be used as a processing software to analyze and correct raw eddy covariance data and other micrometeorological data. This software can be downloaded from https://www.geos.ed.ac.uk/homes/jbm/micromet/EdiRe/Downloads.html. Data from Campbell Scientific dataloggers can be inputted as a raw data file, or processed into a binary file from CardConvert in LoggerNet. It is compatible in Windows and allows users greater access in editing their data including computing variances, adding calibrations, and removing errors etc. More information on EdiRe is available at https://s.campbellsci.com/documents/ca/technical-papers/edire.pdf

Common Errors and Troubleshooting

LoggerNet Errors

Connecting to dataloggers can fail if the connection through the computer/internet is faulty. Failed connections will result in a window indicating the error problem. Most connection problems can be solved by disconnecting from the datalogger and reconnecting. Checking the correct cable attachment from the datalogger to the PC can also fix many connection problems. Checking the power on the datalogger while connecting can also aid in connecting to the datalogger; most dataloggers need between 10V to 12V to operate.
Coding errors can occur through incorrect calibration numbers, specifying inaccurate variables, data tables, subroutines, constants, and instructions within the program. Care should always be made to correctly specify your code to minimize compiling errors. Errors from compiling are usually made available at the bottom of the code, with specific lines indicating where the error is. If your program is still not working after compiling, or your sensor is not compatible with Short Cut, many sensors from multiple companies come with a manuals with recommended codes which can be copy and pasted into CRBasics programs.

Sensor Errors

Sensor errors can occur due to incorrect wiring of sensors, incorrect contact with the datalogger, or incorrect calibration due to program constant or out-of-date calibration coefficients. Make sure your sensors are wired in the correct channel as stated by your program diagram. Check sensor wires for good connection with the datalogger by firmly but gently pulling on the wire after installation. Always make sure that wires are not touching each other or exposed to the elements, as they can become shorted. Confirm correct sensor power, voltage, or excitation channel to make sure the sensor is interacting with the datalogger, or has sufficient power.

Other Errors

Sometimes, LoggerNet, or your program may not be the problem. Make sure all sensors are calibrated often to reduce data errors. Always make sure your power supply is sufficient enough to power and sustain your sensors for correct measurements. In the case of battery drain or power outages, make sure alternate power sources are in place to reduce incorrect measurements or data loss. Updating your datalogger to the newest operating system is always vital for functioning code and efficient communication with the datalogger. To find out your current operating system, access the ‘Station Status’ menu within the ‘Connect’ program of LoggerNet. To upgrade your operating system, download the newest firmware from Campbell Scientific and install following the instructions in the ‘Device Config Utility’ from the ‘Utilities’ menu of LoggerNet. Please note that to use Device Config Utility you must be connected directly to the datalogger. Also note that all settings will be reset to their factory defaults when a new operating system is sent to the datalogger, so remember to collect and/or backup data. Downloads can be accessed from https://www.campbellsci.com/downloads?b=3.

Finally, Campbell Scientific offers support for identifying and troubleshooting errors not limited to LoggerNet coding, wiring of sensors, datalogger errors and inaccurate data collection. Campbell Scientific offers manuals, FAQs, documents, videos and tutorials for almost all problems. They can be accessed by email, fax and phone (435.227.9000). More information can be accessed at https://www.campbellsci.com/support

Data Analysis

Preliminary data from Oct 20, 2017 to Nov 14, 2017 was collected on a flash drive and analyzed for discussion. Each sensor was inspected and is discussed below.

HMP45C Temperature, Relative Humidity, and Vapor Pressure Deficit

Upon initial inspection of the data from the Baker Hall EC Tower, it was noted that the values reported from the HMP45C for air temperature, relative humidity, and vapor pressure deficit did not look correct (Figure 6a,c). The variation looked somewhat as expected, however the values do not vary much with time. This potentially could indicate a bad sensor or an issue with the coefficients assigned in the program that runs the data logger. The range suggested by panel temperature (Figure 6b) appears more reasonable and might be useful in diagnosing the problem with the HMP45C.

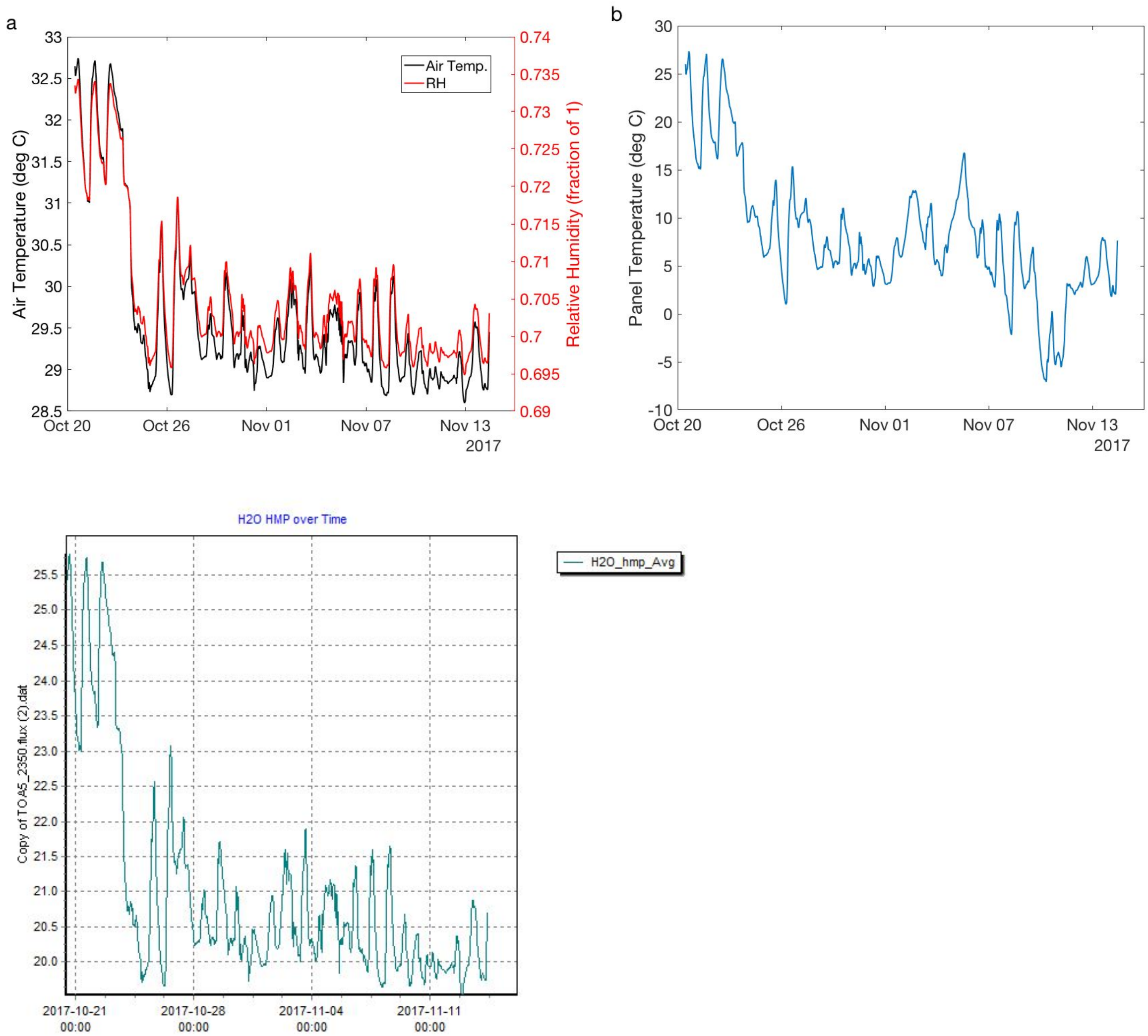


Figure 6. a) Air temperature and relative humidity as reported by the HMP45C; b) Panel temperature as reported by the CR5000 datalogger; c) Vapor pressure deficit as reported by the HMP45C.

Precipitation

After reviewing the precipitation data over the test period (Figure 7), the datalogger showed several significant rain events. The values for each rain measurement appeared within the expected amounts, and exhibited no issue.

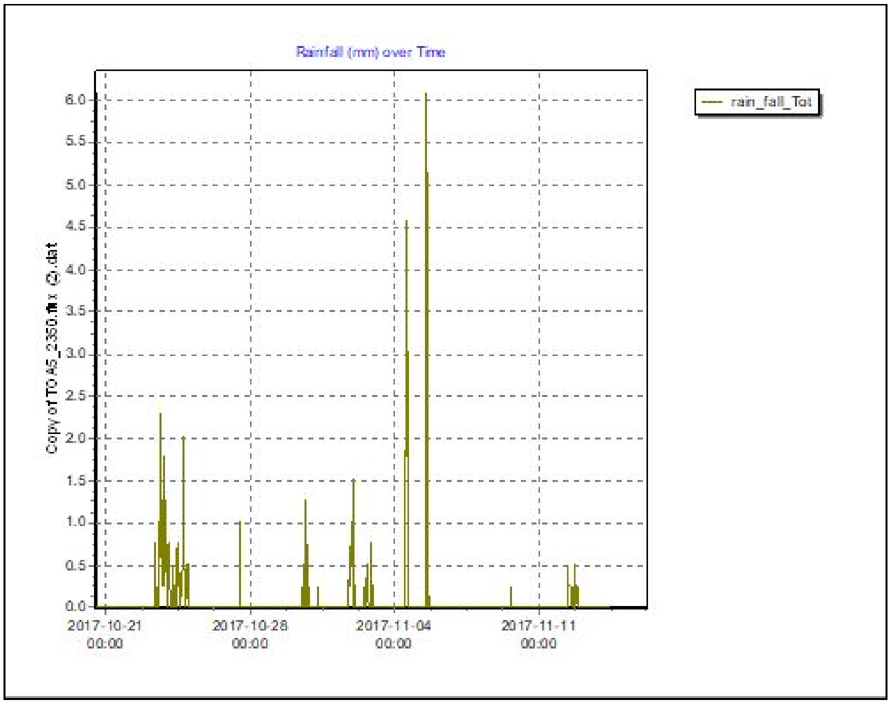
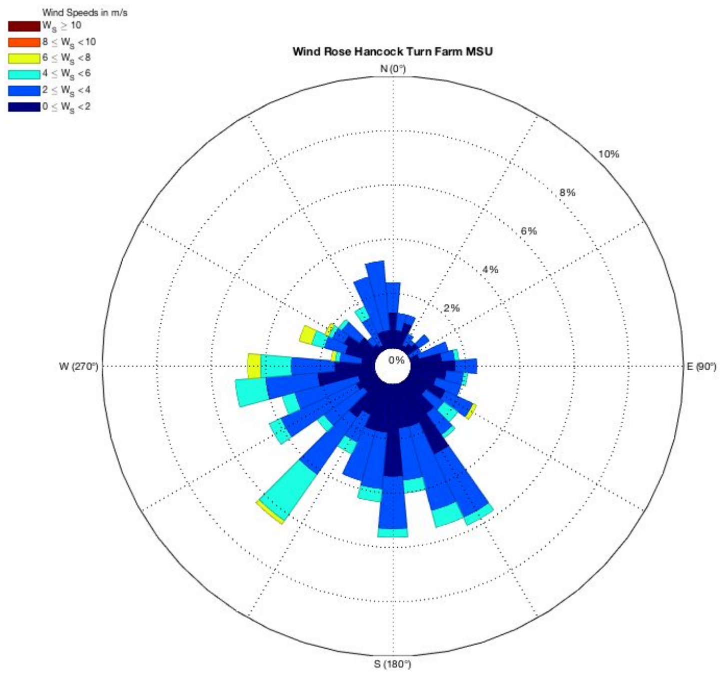


Figure 7. Precipitation events during the study period exhibited no issue.

Comparison to EnviroWx Agricultural Weather Station at MSU Hancock Turf Farm

Several analyses were performed to compare the data collected from the Baker Hall EC Tower to the EnviroWx station located approximately two miles to the south of Baker Hall at the corner of Farm Lane and Mt. Hope Highway. These included comparisons of wind speed and wind direction (Figure 7), downward shortwave solar radiation (Figure 8), and panel temperature (in lieu of HMP45C air temperature) to the EnviroWx air temperature (Figure 9). Measurements from Hancock Turf Farm were only available at an hourly time step; as a result where appropriate, averaging was performed on each variable from the Baker Hall Tower to make values comparable. Wind speeds and directions were similar at the two stations, even given the substantial height difference between the wind measurements at both stations. The most interesting difference between the two stations is the large prevalence of south easterly winds at the Baker Hall Tower that were not present at Hancock. Wind speeds were slightly faster on average at Baker, which is not surprising given the higher anemometer height. Solar radiation was more divergent between the two sites, though they follow one another (r = 0.9), at higher radiation values there is a divergence with the Hancock site reading somewhat higher. This is likely due to the difference in observation times between the two sites (0.5 hour vs. one hour). Panel temperature from the Baker Tower and and 1.5 meter air temperature from Hancock were very similar, with the panel temperature being somewhat warmer due to it not being shielded and enclosed in a shelter.



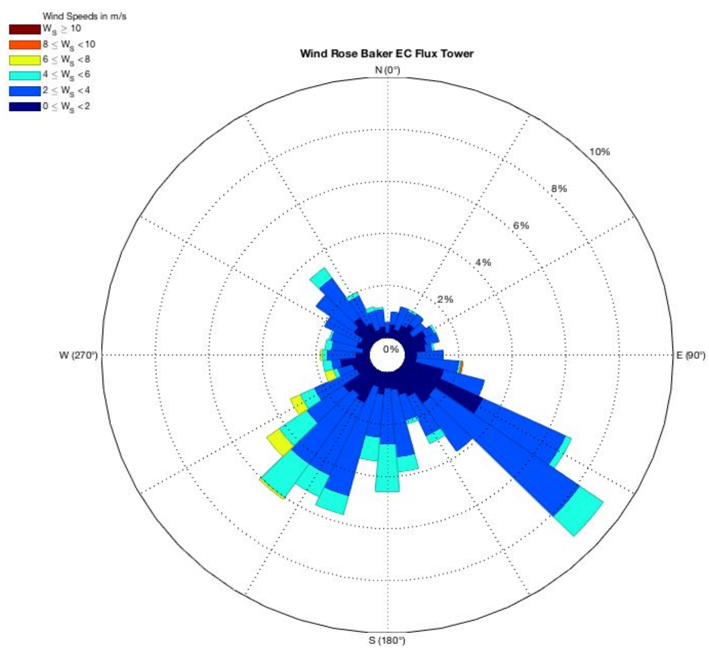


Figure 8. Wind roses showing meteorological wind direction and wind intensity for Hancock Turf Farm (Top) and Baker Hall EC Tower (Bottom) for the period from October 20th to November 14th.

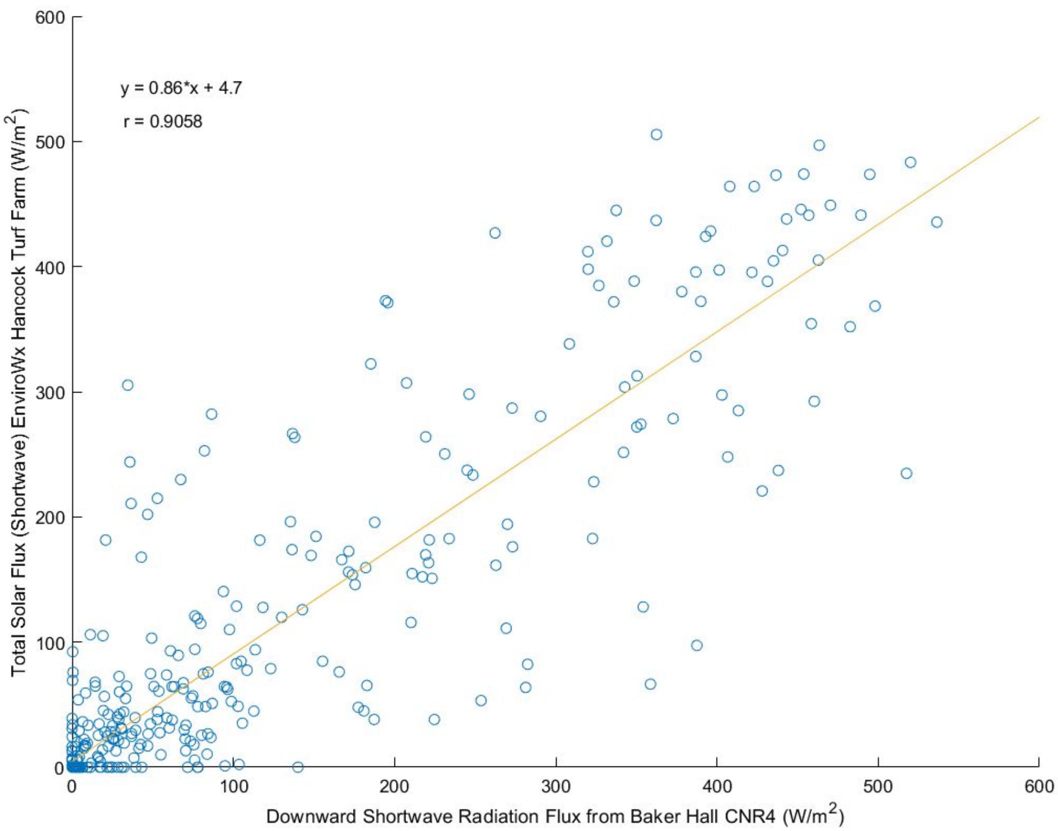


Figure 9. Downward shortwave radiation flux (W/m^2) from Hancock Turf Farm and Baker Hall EC Tower from October 20th 2017 to November 14th 2017. Pearson correlation coefficient (r) and least-squares regression line is given associating between the two variables

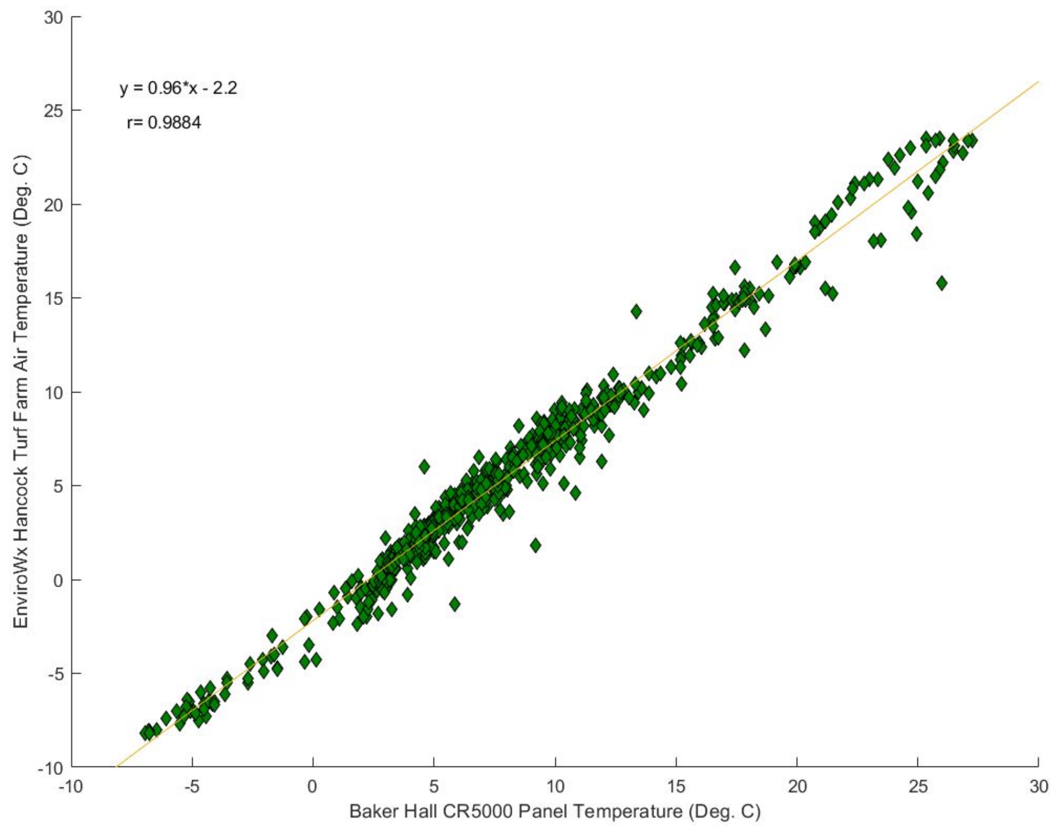


Figure 10. 1.5 meter air temperature from Hancock Turf Farm and CR5000 panel temperature from Baker Hall EC Tower. Pearson correlation coefficient (r) and least-squares regression line is given associating between the two variables

3D Wind Speed CSAT3

The CSAT3 caught 3D wind speed during the study period. Observations were made in LoggerNet's ViewPro, which can quickly graph flux data. Typically, wind speeds will not exceed 1 m/s, yet in Figure 11 the Uz m/s is seen reaching 1.6. This large jump in Uz m/s between October 30, 2017 and November 2, 2017 is most likely an error that would need processing in EdiRe to resolve.

CSAT3 Uz average wind speed

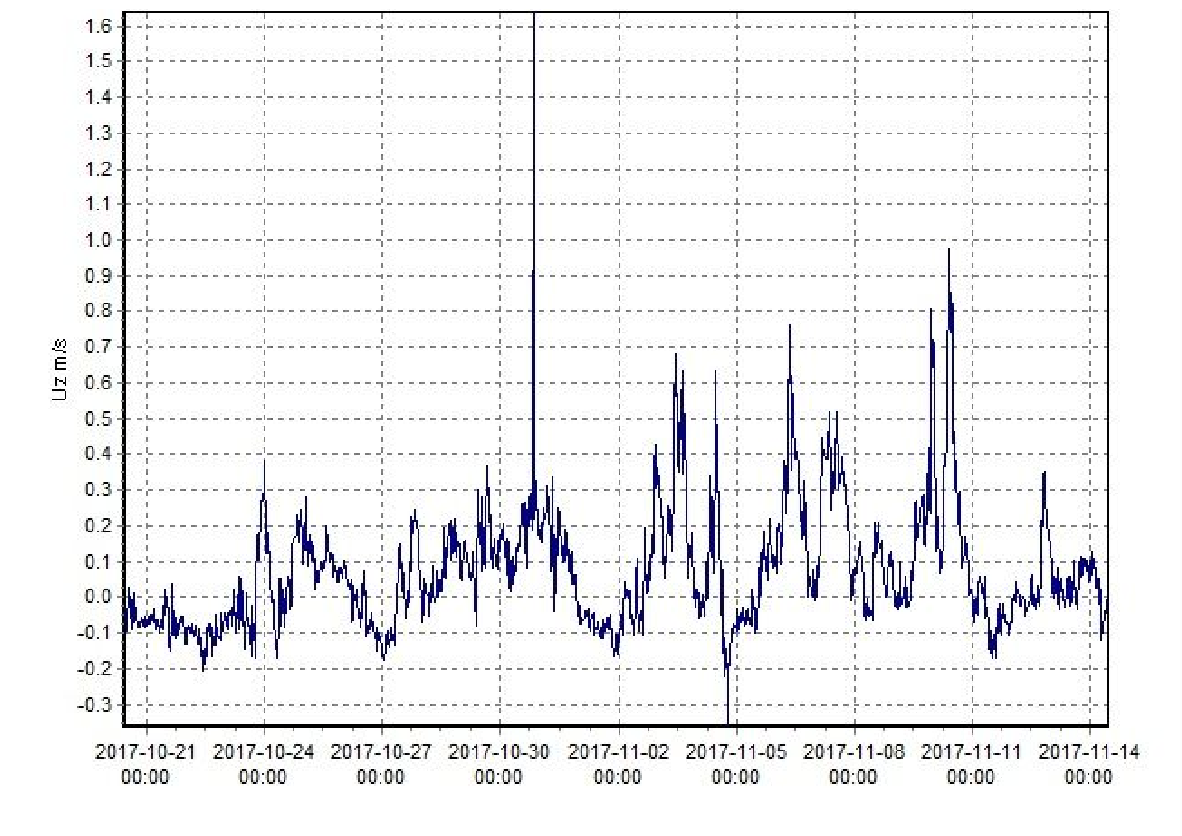


Figure 11. Measured 3D windspeed by CSAT3 for Uz over the study period.

However, when observing Uy, Uz and Ux together, this detail can be overlooked (Figure 12). When observing 3D wind speed, it is an important note to observe each component individually to see the finer scale and process the data further in EdiRe.

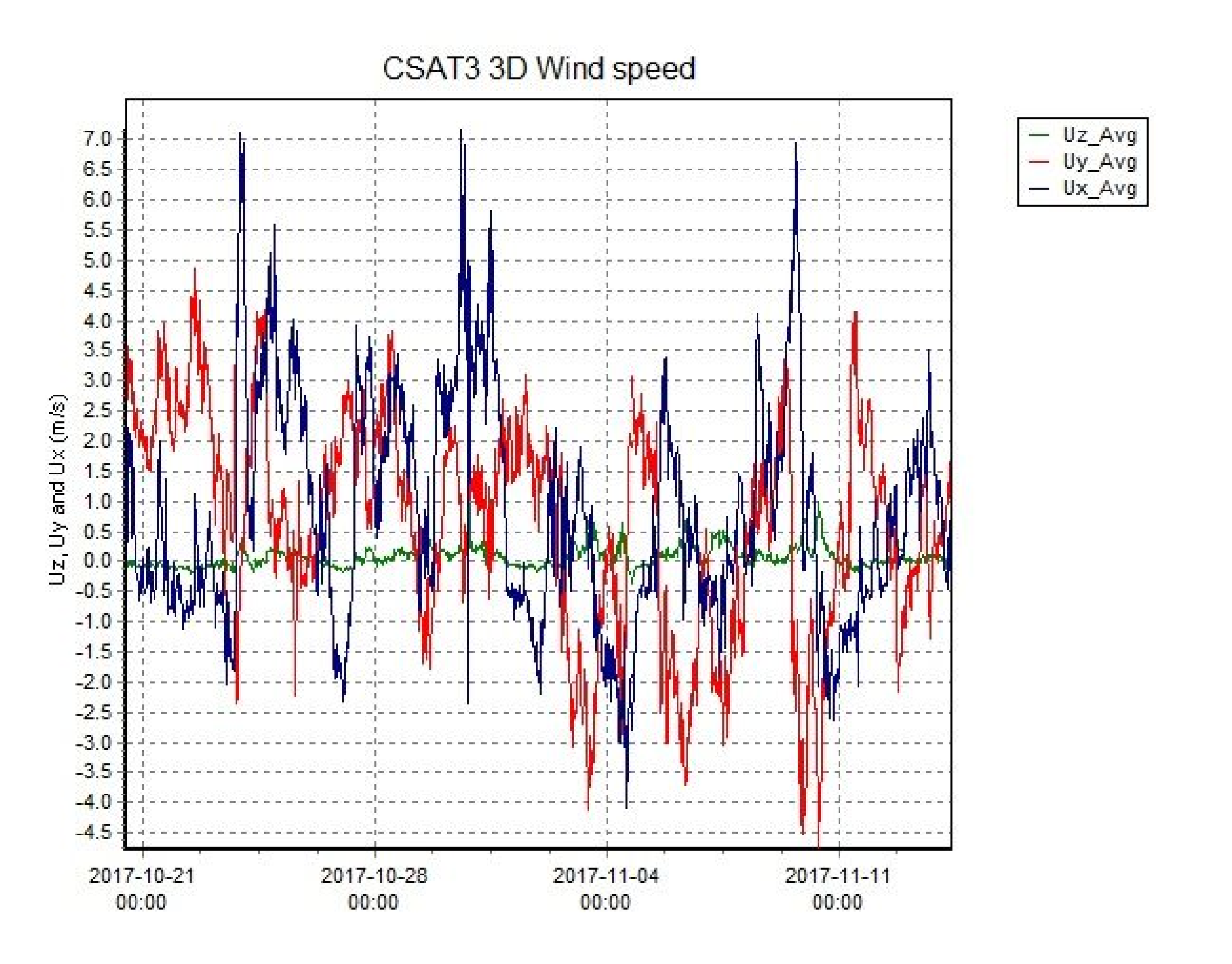


Figure 12. Measured 3D wind speed by CSAT3 for Uz, Uy, and Ux over the study period.

Comparison between Q.7 and CRN4

Radiation levels throughout the study period were collected from both Q.7 and the CRN4. Both instruments offer a total average, while the CRN4 can offer multiple levels of radiation height. A brief comparison in Figure 13 demonstrates how similar their measured results were.

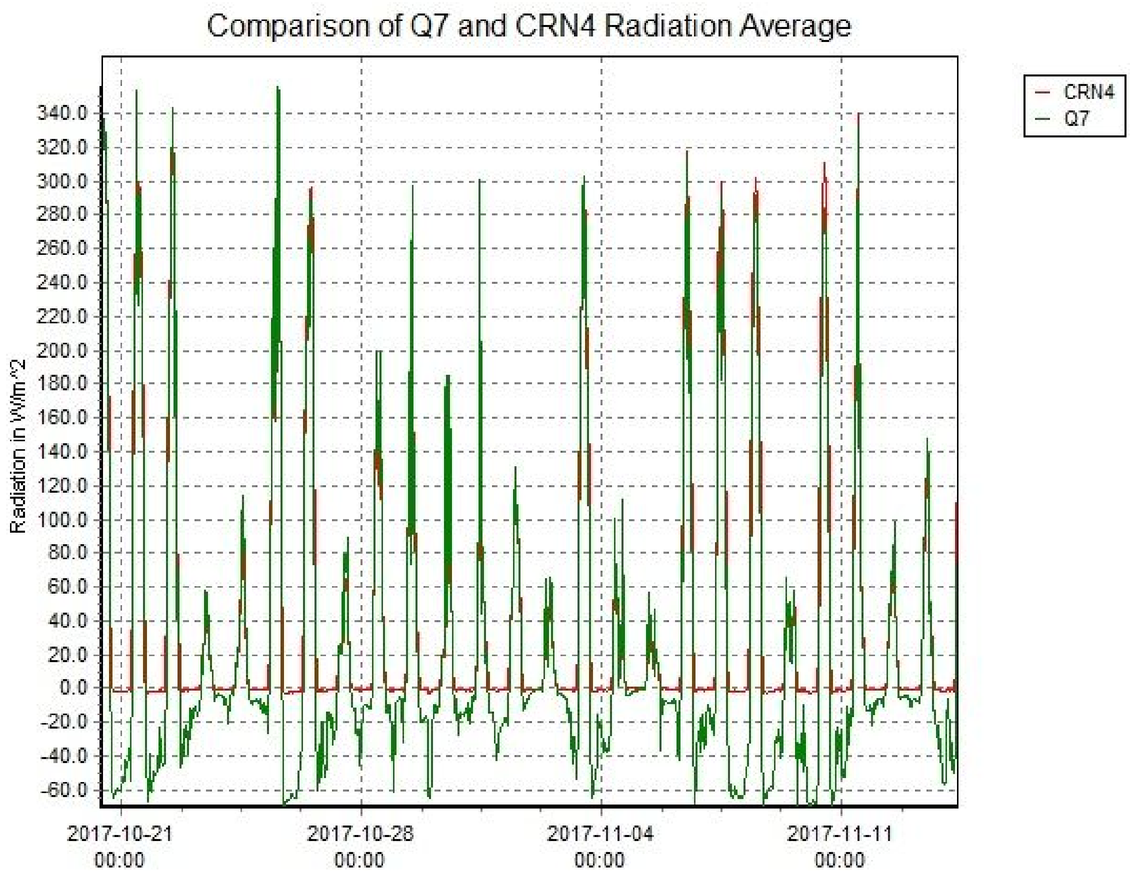


Figure 13. Measured comparisons of average radiation show little to no difference between the Q.7 and CRN4 at first glance.

CO^2 LI-7500

LI-7500 CO^2 results yielded fairly consistent data across our study time-frame (Fig 14). Average CO^2 was 785.1 mg/m³, with maximum ranges of 1061.1 mg/m³ and minimum of 623.3 mg/m³. Standard errors deviated at an average of 53.45 mg/m³. The current trend demonstrated a small but steady increase in CO^2 over the time period, which coincides well with the loss of vegetation during the fall season of October to November. This loss of vegetation would cause slightly more CO^2 to be present in the atmosphere from the decrease in photosynthesis. There were some spikes in the data which were observed on October 27th and October 30th. The LI-7500 may have been affected slightly from intensive precipitation which occurred on those dates and caused some error in data acquisition. These points most likely would have been flagged by EdiRe as anomalies and removed from the final data product. Overall, the LI-7500 is working effectively and the data acquired can be deemed as valid.

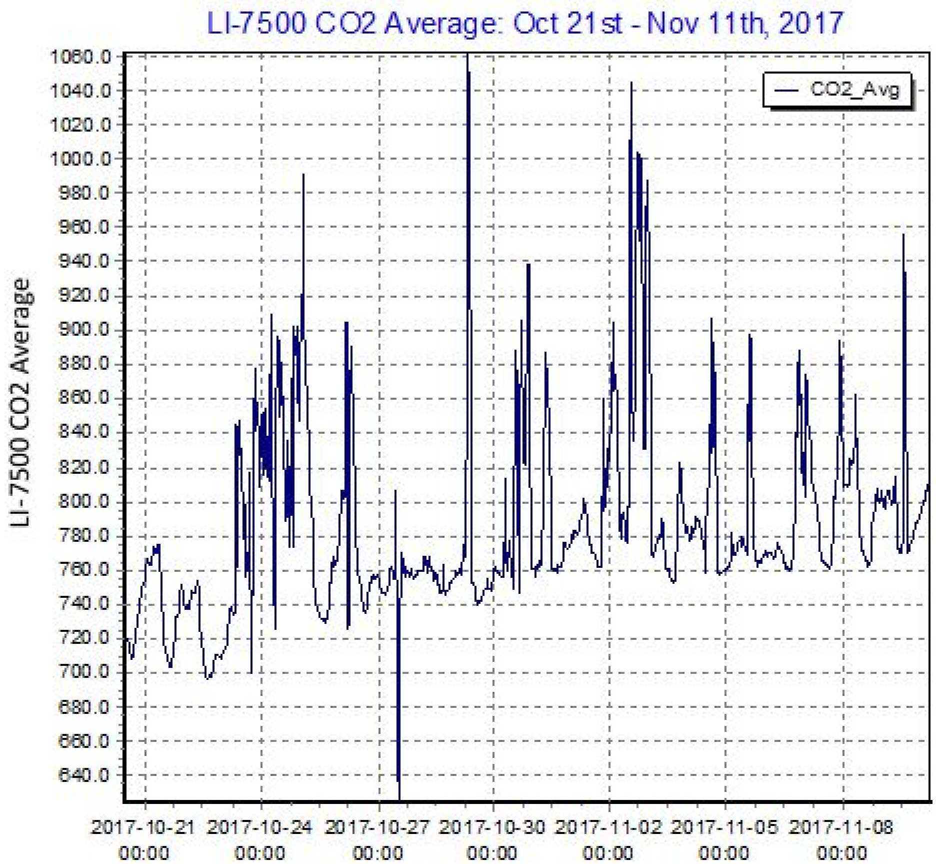


Figure 14. LI-7500 CO^2 averages for Baker Hall flux tower for one month. Results indicate a carbon dioxide average of 760 mg/m^3.

Acknowledgements

We are grateful for the guidance and support those outside of GEO 892. We thank Dr. David Reed for his workshop on programming and LoggerNet software. We also thank Dr. Zutao Ouyang for his technical support to connect internet access. Finally, we are grateful for those involved with Infrastructure Planning and Facilities and IT at Michigan State University for their counseling and troubleshooting, particularly Wayne Buck and his team and Tom Hakes.

Appendix A

Connect Loggernet to Baker Flux Tower remotely

To connect to the Baker flux tower, you must have previously installed Loggernet to view the data. Instructions vary by user location. If you are connecting from outside of Michigan State University's main campus, being at Section 1. If you are connecting on campus, you may skip to Section 2.

Section 1: Access Baker Tower outside of Michigan State University main campus

1. Open the following link and download Teamviewer
2. Open Teamviewer and enter the following into Partner ID: "50371242"
3. Enter "LEES" if prompted for a password
4. Open Loggernet
5. Select CR5000 from the Connect menu for Baker Tower

Section 2: Connect to Baker Tower using Loggernet

1. Open Loggernet
2. Click "Setup" on the main screen to open the Setup window
3. Click "Add" to open the EZSetup Wizard and follow the prompts using the following
    a. Introduction: Click "Next"
    b. Communication Setup:
        i. Click on the image of the CR5000 datalogger, enter a name, click "Next"
        ii. Select "IP Port", click "Next"
        iii. Enter Internet IP Address " 35.15.45.129.6785"
    c. Datalogger Settings: Default OK, click "Next"
    d. Setup Summary: Click "Next"
    e. Communication Test: Test Communication "Yes", click "Finish"
4. Return to the main menu and connect to the datalogger

Appendix B

Table of Variables and Units for Baker Hall Flux Tower

Name	Unit		Name	Unit
Fc_wpl	mg/(m^2s)		batt_volt_Avg	V
LE_wpl	W/m^2		std_wnd_dir	degrees
Hs	W/m^2		n_Tot	samples
tau	kg/(ms^2)		csat_warnings	samples
u_star	m/s		irga_warnings	samples
cov_Uz_Uz	(m/s)^2		del_T_f_Tot	samples
cov_Uz_Ux	(m/s)^2		track_f_Tot	samples
cov_Uz_Uy	(m/s)^2		amp_h_f_Tot	samples
cov_Uz_CO2	mg/(m^2s)		amp_l_f_Tot	samples
cov_Uz_H2O	g/(m^2s)		chopper_f_Tot	samples
cov_Uz_Ts	mC/s		detector_f_Tot	samples
cov_Ux_Ux	(m/s)^2		pll_f_Tot	samples
cov_Ux_Uy	(m/s)^2		sync_f_Tot	samples
cov_Ux_CO2	mg/(m^2s)		agc_Avg	unitless
cov_Ux_H2O	g/(m^2s)		Fc_irga	mg/(m^2s)
cov_Ux_Ts	mC/s		LE_irga	W/m^2
cov_Uy_Uy	(m/s)^2		CO2_wpl_LE	mg/(m^2s)
cov_Uy_CO2	mg/(m^2s)		CO2_wpl_H	mg/(m^2s)
cov_Uy_H2O	g/(m^2s)		H2O_wpl_LE	W/m^2

Name	Unit		Name	Unit
cov_Uy_Ts	mC/s		H2O_wpl_H	W/m^2
cov_CO2_CO2	(mg/m^3)^2		H2O_hmp_Avg	g/m^3
cov_H2O_H2O	(g/m^3)^2		t_hmp_Avg	C
cov_Ts_Ts	C^2		rh_hmp_Avg	fraction
Ux_Avg	m/s		Rn_Q7_1_Avg	W/m^2
Uy_Avg	m/s		Rad_short_Up_Avg	W/m^2
Uz_Avg	m/s		Rad_short_Dn_Avg	W/m^2
CO2_Avg	mg/m^3		Rad_long_Up_Avg	W/m^2
H2O_Avg	g/m^3		Rad_long_Dn_Avg	W/m^2
Ts_Avg	C		Rn_total_Avg	W/m^2
rho_a_Avg	kg/m^3		Temp_K_Avg	Temp_K
press_Avg	kPa		rain_fall_Tot	mm
panel_temp_Avg	C		par_totflx_Tot	mmol/m^2
wnd_dir_compass	degrees		par_flxdens_Avg	mmol/m^2
wnd_dir_csat3	degrees		e	kPa
wnd_spd	m/s		e_sat	kPa
rslt_wnd_spd	m/s		batt_volt	V

Appendix C

Wiring Diagram

Edited: Cheyenne Lei, October 24th, 2017
This program measures the following sensors:

CSAT3	Three dimensional Sonic Anemometer
LI-7500A	Open Path Infrared Gas Analyzer (CO2 and H2O)
CNR4	Net Radiometer
LI190SB-L	Quantum Sensor
HMP45C-L	Temp/RH Sensor
TE525-L	Tipping Bucket Rain Gauge
SI-111	Apogee Model SI-111 Infrared Radiometer
Q7.1	Net Radiometer

Wiring

Analog / Deferential Channel

1H	CNR4 Pyranometer Up Signal-short wave (red)
1L	CNR4 Pyranometer Up Reference-short wave (blue)
G	
2H	CNR4 Pyranometer Down Signal-short wave (white)
2L	CNR4 Pyranometer Down Reference-short wave (black)
G	
3H	CNR4 Pyrgeometer Up Signal-long wave (grey)
3L	CNR4 Pyrgeometer Up Reference-long wave (yellow)
G	
4H	CNR4 Pyrgeometer Signal-long wave (brown)
4L	CNR4 Pyrgeometer Down Reference-long wave (green)
G	Connect Jumper Wire to 4L; CNR4 shield (clear)
5H	
5L	CNR4 thermistor signal (white)
G	CNR4 thermistor signal reference (black); CNR4 thermistor shield (clear)
7H	LI190SB Signal+ (red) *green cable*
7L	LI190SB Signal- (black) *blue cable*
G	LI190SB Shield (clear); connect jumper to 7L

8H	Q7.1 signal (red)
8L	Q7.1 signal reference (black)
G	Q7.1 shield (clear)
9H	HMP45C #1 temperature signal (yellow)
9L	
G	HMP45C #1shield (clear), HMP45C #1 signal reference (purple)
10H	HMP45C #1relative humidity signal (blue)
10L	short jumper wire to 9L
G	
12H	apogee SI-111 (green)
12L	
G	apogee SI-111 (blue)
13H	apogee SI-111 (red)
13L	apogee SI-111 (black)
G	apogee SI-111 (clear)
Pulse Ports	
P1	TE525 (black)
G	TE525 (white), TE525 (clear)
Voltage excitation	
VX1	
VX2	apogee SI-111 (white)
VX3	CNR4 thermistor voltage excitation
G	
Power	
5V	HMP45C power control (orange)
12V	HMP45C power (red)
G	HMP45C power reference (black)
SW12	
G	
12V	CR5000 (red)
12V	Q.7. Fan (red)
G	CR5000 (black), Q.7. Fan (black & clear)
12V	CSAT3 power (red)
	LI-7500 power (red)
G	CSAT3 power reference (black)
G	CSAT3 power shield (clear)
G	LI-7500 power reference (black)

SDM Channels

SDM-C1	CSAT3 SDM Data (green)
	LI-7500 SDM Data (blue)
SDM-C2	CSAT3 SDM Clock (white)
	LI-7500 SDM Clock (white)
SDM-C3	CSAT3 SDM Enable (brown)
	LI-7500 SDM Enable (brown)
G	CSAT3 SDM reference (black)
	CSAT3 SDM shield (clear)
	LI-7500 SDM reference (black)
	LI-7500 SDM shield (clear)

Appendix D

Calibration of Sensors

Sensor	Constant Name	Calibration #	Variables	Units
TE 525-L	N/A	N/A	rainfall	mm
HMP 45C-L	N/A	N/A	e	kPa
			e_sat	kPa
			t_hmp	ºC
			rh_hmp	%
LI 190 SB-L	PAR	11.3	par_flxdens	umol/s/m^2
			par_totflx	mmol/m^2
SI-111	mc2	71530	SBTempC	ºC
	mc1	7483060	SBTempK	ºK
	mc0	1355440000	TargmV	mV
	bc2	18268	TargTempC	ºC
	bc1	309008	TargTempK	ºK
	bc0	4270710		
CSAT3	Angle	230	Ux	m/s
			Uy	m/s
			Uz	m/s
LI7500A	Internally Calibrated		CO^2	mg/m^3
			H^2O	g/m^3
			CO^2_u_mol	umol/mol
			H^2O_u_mol	umol/mol
Q.7.1	Positive	9.44	Rn_Q7_1	W/m^2
	Negative	11.76		
CNR4	Shortwave Upper	16.49	Rad_short_Up	W/m^2

	Shortwave Lower	19.67	Rad_short_Dn	W/m²
	Longwave Upper	7.94	Rad_long_Up	W/m²
	Longwave Lower	9.91	Rad_long_Dn	W/m²
			Rn_short	W/m²
			Rn_long	W/m²
			Rn_total	W/m²
Battery	Battery	N/A	batt_volt	V

Appendix E

Station design and part list

The following images were used in the final tower design stages and are near identical to the present construction. A table including the tower's frame and power supplies is provided below.

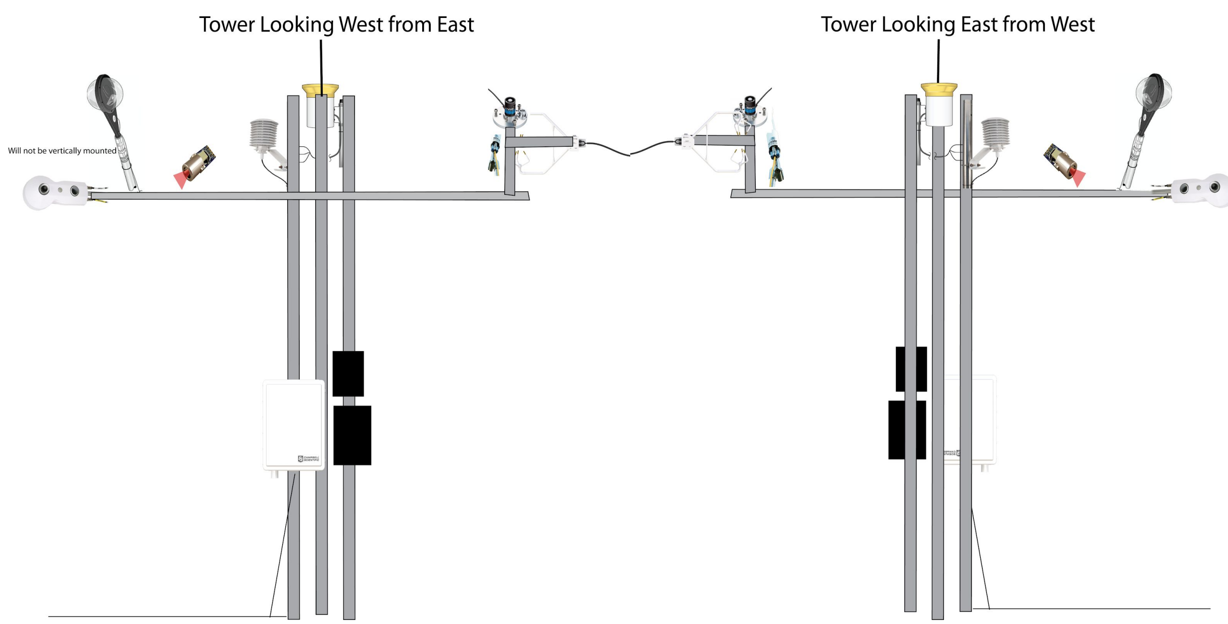


Figure 1. Design sketches depicting the tower orientation facing east (left) and west (right).

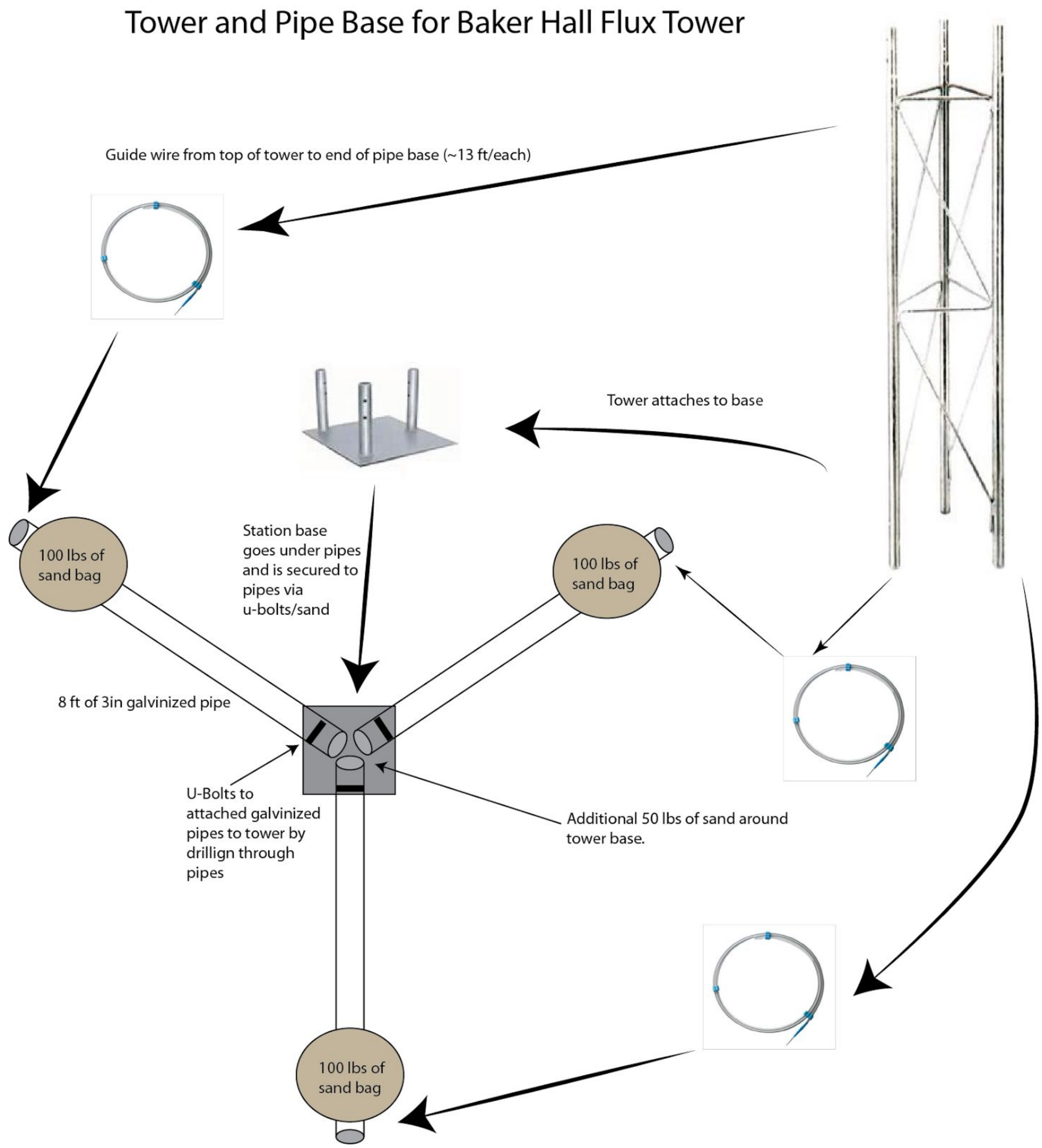


Figure 2. Framework of the tower base. An antenna tower is placed into a stabilizing base, which then rests under three galvanized steel pipes that intersect. Each pipe acts as a foot and is weighed down with 100lbs of sand. Guy wire is strung from each of the three outer ends of the pipe to the antenna tower's upper section.

Figure 3. Images used in tower designs and their labels.

Table 1. A complete list of all the infrastructure (i.e., no sensors) used to build the tower's frame and provide electricity.

Item	Measurements	Quantity	Purpose
Tower frame supplies			
Galvinized Pipe	2" x 8'	3	Tower footing
	1 3/4" x 6'	1	Tower arms
	1 3/4 x 4'	1	Tower arms
Geotextile	3 yrds	1	Sandbag cover
50 lb Bags of Sand/Gravel	50lbs	7	Stabilize footing
U-Bolts	3/4"	20	Attach sensors
	1/2"	10	Attach sensors
Aluminum plates	various	10	U-bolt brackets
Zip Ties	various	as needed	Fasten sensor wires
Guy wire clamp/thimble	1/4"	6 clamps; 3 thimbles	Fasten guy wire to feet
Guy wire	1/4" x 13'	3	Stabilize feet to tower
Lightning Rod	1/2" x 3'	1	Grounding
Grounding Cable	3/4" x 5'	1	Grounding
Power supplies			
12V Deep Cycle Marine Batteries	--	2	Backup power
Power cord, heavy duty	42'	1	Power from battery case to enclosure
Power Adapter/Charger	--	1	Convert power to from 110V to 12V
Battery cases (w/ belts)	--	3	Covers 12V batteries
Heavy Duty Storage bin	2.5x4x2'	1	Houses all 12V batteries
Copper wire, fibrous and covered	1/4" x 2'	1	Cut to connect batteries

Appendix F

CRBasics Code used on Baker Hall Flux Tower

```
'CR5000
'Copyright (c) 2003 Campbell Scientific, Inc. All rights reserved.
'3rd, October, 2017 for MSU Flux Tower
'Cheyenne Lei'
'This program measures the following sensors:
'*************************************************************************************
' CSAT3 Three dimensional Sonic Anemometer
' LI-7500A Open Path Infrared Gas Analyzer (CO2 and H2O)
' CNR4 Net Radiometer
' LI190SB-L Quantum Sensor
' HMP45C-L Temp/RH Sensors (three additional)
' TE525-L Tipping Bucket Rain Gauge
' SI-111 Apogee Model SI-111 Infrared Radiometers
' Q7.1 Net Radiometer
'*maybe* 31013 Vaisala Barometer*
'*************************************************************************************

'*************************************************************************************
' This CR5000 program measures turbulence sensors at 10 or 20 Hz. The time series can be
'saved to a PC Card. The CR5000 will also compute on-line turbulent fluxes from the
'measured data.
'The sign convention for the fluxes, except net radiation, is positive away from the
'surface and negative toward the surface.
'
' The CR5000 will introduce delays into the CSAT3 and CR5000 Panel Temperature
'data. These delays match the fixed 302.369 mSec delay that Campbell Scientific
'programs into the LI-7500 of the LI-7500 irga (see page 3-23 of the LI-7500 manual
'published by Li-Cor). The delays are a function of the Scan Interval and are computed
'automatically by the program.
'
'The site attendant must load in several constants and calibration values. Search
'for the text string "unique" to find the locations where unique constants and
'calibration values are entered.
'*************************************************************************************
```

```
'*** Unit Definitions *******************************************************************
'Units Units
'C Celsius
'degrees degrees (angle)
'g grams
'J Joules
'kg kilograms
'kPa kilopascals
'm meters
'mg milligrams
'mSeconds milliseconds
'mV millivolts
's seconds
'V volts
'W Watts
'******************************************************************************************

'*** Operation****************************************************************************
' Set the constant SCAN_INTERVAL to 100 mSec (10 Hz) or 50 mSec (20 Hz). The
'appropriate value of the constants CSAT_OPT, ANALOG_DELAY, and CSAT_DELAY will be
'computed by the CR5000. Connect all the sensors to the CR5000 as described in the
'Wiring section.
'
'Download the program to the CR5000. When servicing the station, set save_flx_flag_on
'low or set Custom Menu option "Save Fluxes?" to No, note that if save_flx_flag_on ("Save
'Fluxes?") is not set back to high (Yes), no flux data will be saved.
'*** User Control in "Custom" Menu of Program Control Flags ***
'The Custom Menu allows the station operator to conveniently change the status of the
'Program Control Flags using the CR5000 keyboard display.
'Save Fluxes? Change to "Yes" (default) to start processing flux data.
'save_flx_flag_on Change to "No" when cleaning webs from CSAT3 or performing other
' site maintenance. This "bad" data will not be included in the on-line fluxes.
'
'Save Time Series? Change to "Yes" (default) to begin collecting time series on PC Card.
'save_ts_flag_on Change to "No" to stop collecting time series on the PC Card.
'******************************************************************************************

'***Wiring********************************************************************************
'**********SDM INPUT CHANNEL**************
CSAT and LI-7500A
'SDM-C1 CSAT3 SDM Data (green)
' LI-7500 SDM Data (blue)
'SDM-C2 CSAT3 SDM Clock (white)
```

```
' LI-7500 SDM Clock (white)
'SDM-C3 CSAT3 SDM Enable (brown)
' LI-7500 SDM Enable (brown)
'G CSAT3 SDM reference (black)
' CSAT3 SDM shield (clear)
' LI-7500 SDM reference (black)
' LI-7500 SDM shield (clear)
'******************************************************

'*********ANALOG INPUT CHANNEL**************
CNR4 and LI190SB and Q7.1 and HMP45C***********
'1H CNR4 Pyranometer Up Signal-short wave (red)
'1L CNR4 Pyranometer Up Reference-short wave (blue)
'gnd Connect Jumper Wire to 1L
'
'2H CNR4 Pyranometer Down Signal-short wave (white)
'2L CNR4 Pyranometer Down Reference-short wave (black)
'gnd Connect Jumper Wire to 2L
'
'3H CNR4 Pyrgeometer Up Signal-long wave (grey)
'3L CNR4 Pyrgeometer Up Reference-long wave (yellow)
'gnd Connect Jumper Wire to 3L
'
'4H CNR4 Pyrgeometer Signal-long wave (brown)
'4L CNR4 Pyrgeometer Down Reference-long wave (green)
'gnd Connect Jumper Wire to 4L; CNR4 shield (clear)
'
'5H
'5L CNR4 thermistor signal (white)
'gnd CNR4 thermistor signal reference (black); CNR4 thermistor shield (clear)
'
'6H
'6L
'gnd
'
'7H LI190SB Signal+ (red) *green cable*
'7L LI190SB Signal- (black) *blue cable*
'gnd LI190SB Shield (clear); connect jumper to 7L
'
'8H Q7.1 signal (red)
'8L Q7.1 signal reference (black)
'gnd Q7.1 shield (clear)
'
```

```
'9H HMP45C #1 temperature signal (yellow)
'9L
'gnd HMP45C #1shield (clear) , HMP45C #1 signal reference (purple)
'
'10H HMP45C #1relative humidity signal (blue)
'10L short jumper wire to 9L
'gnd
'
'11H
'11L
'gnd
'
'12H (SE 23) apogee SI-111 (green)
'12L
'gnd apogee SI-111 (blue)
'
'13H apogee SI-111 (red)
'13L apogee SI-111 (black)
'gnd apogee SI-111 (clear)
'
'14H
'14L
'gnd
'
'15H
'15L
'gnd
'
'16H
'16L
'gnd
'
'17H
'17L
'gnd
'18H
'18L
'gnd
'
'19H
'19L
'gnd
'
```

```
'20H
'20L
'gnd
'*************************************************************************************
'***********PULSE PORT***************TE525 rain gauge***********************
'P1 TE525 (black)
'G TE525 (white)
'G TE525 (clear)
'P2
'G
'*************************************************************************************
'***********CURRENT EXCITATION*********************
‘CNR4(pt-100) and apogee SI-111**********
'IX1 Pt-100 Current Excitation (red)
'gnd Pt-100 Current Excitation Ground (blue)
'voltage excitation
'VX1
'VX2 apogee SI-111 (white)
'VX3 CNR4 thermistor voltage excitation
'G
'*************************************************************************************
'*************CONTROL PORT***********************************************************
'C1
'C2
'C3
'C4
'G
'
'*************************************************************************************
'***************POWER OUT*************************
‘HMP45C******************************
'5V HMP45C #1 power control (orange)
'12V HMP45C #1 power (red)
'
'
'G HMP45C #1 power reference (black)
'
'
'SW12
'
'
'
'G
```

```
'
'
'
'**************************************************************************************
'*********POWER IN*******************************************************************
'12V CR5000 (red)
'12V Q.7. Fan (red)
'G CR5000 (black)
'G Q.7. Fan (black & clear)
'**************************************************************************************
'*******EXTERNAL
POWERSUPPLY**********************************************************
'12V POS CSAT3 power (red)
' LI-7500 power (red with white) 'Power = red
' CR5000 (red)
'G NEG CSAT3 power reference (black)
'G CSAT3 power shield (clear)
' LI-7500 power reference (red with black)
' LI-7500 ground (green) 'G = black
' CR5000 (black)
'**************************************************************************************

'*** Constants************************************************************************
Const ANGLE_FROM_NORTH = 230 'Unique value changed from 320 to 210.
'Compass azimuth of the -x axis. For the figure
' below, ANGLE_FROM_NORTH = 90.
'The program computes the Compass wind direction, using the constant ANGLE_FROM_NORTH,
'and a CSAT3 wind direction. Good CSAT3 wind directions are between -90 to 0 and 0 to
'90 degrees (-pi/2 to 0 and 0 to pi/2 radians), i.e. the wind is blowing into the CSAT3
'sensor head.
'Measurement Rate '10 H,z 20 Hz
Const SCAN_INTERVAL = 100 '100 (mSec) 50 (mSec)

'Output period
Const OUTPUT_INTERVAL = 30 'On-line flux data output interval in minutes.
Const CSAT_OPT = INT (1/SCAN_INTERVAL*1000) '10 (Hz) 20 (Hz)
Const ANALOG_DELAY = INT (300/SCAN_INTERVAL+1) '4 (3 scan delay) 7 (6 scan delay)
Const CSAT_DELAY = INT (ANALOG_DELAY-2) '2 (1 scan delay) 5 (4 scan delay)
Const Q7_1_POS_CAL = 9.44 'Unique positive multiplier for Q7.1.
Const Q7_1_NEG_CAL = 11.76 'Unique negative multiplier for Q7.1.

'Unique CNR4 Sensitivity
Const CNR4_pyrano_up = 16.49 'Also called on sensor: Shortwave Upper'
```

```
Const CNR4_pyrano_dn = 19.67 'Also called on sensor: Shortwave Lower'
Const CNR4_pyrgeo_up = 7.94 'Also called on sensor: Longwave Upper'
Const CNR4_pyrgeo_dn = 9.91 'Also called on sensor: Longwave Lower'

'CNR4 multipliers
Const pyrano_up_mult = 1000/CNR4_pyrano_up
Const pyrano_dn_mult = 1000/CNR4_pyrano_dn
Const pyrgeo_up_mult= 1000/CNR4_pyrgeo_up
Const pyrgeo_dn_mult = 1000/CNR4_pyrgeo_dn
Const Excite_I=100 'Current Excitation at 100uA
Const par_cal = 11.30 'Unique calibration value for LI190SB in uA/mmol/s/m^2 'Changed
Const par_mult_flxdens = 1000/(par_cal*0.604) 'multiplier for Flux Density calculation
Const par_mult_totflx = (1/(par_cal*0.604))*(SCAN_INTERVAL/1000) 'multiplier for Total Flux c
Const CP = 1004.67 'Estimate of heat capacity of air [J/(kg K)].
Const LV = 2440 'Estimate of the latent heat of vaporization [J/g].
Const SDM_PER = 30 'Default SDM clock speed, 30 uSec bit period.
Const RD = 0.28704 'Gas constant for dry air [J/(g K)].
Const RV = 0.0004615 'Gas constant for water vapor [J/(mg K)].
Const R = 8.3143e-3 'Universal gas constant [ (kP m^3) / (K mol) ].
Const A_0 = 6.107800 'Coefficients for the sixth order approximating
Const A_1 = 4.436519e-1 ' saturation vapor pressure polynomial (Lowe,
Const A_2 = 1.428946e-2 ' Paul R., 1976.: An approximating polynomial for
Const A_3 = 2.650648e-4 ' computation of saturation vapor pressure, J. Appl.
Const A_4 = 3.031240e-6 ' Meteor., 16, 100-103).
Const A_5 = 2.034081e-8
Const A_6 = 6.136821e-11

'Declare constants for Apogee Model SI-111 Infrared Radiometers
'mC2 = polynomial coefficient (C2) used to calculate slope (m)
'mC1 = polynomial coefficient (C1) used to calculate slope (m)
'mC0 = polynomial coefficient (C0) used to calculate slope (m)
'bC2 = polynomial coefficient (C2) used to calculate intercept (b)
'bC1 = polynomial coefficient (C1) used to calculate intercept (b)
'bC0 = polynomial coefficient (C0) used to calculate intercept (b)
'Note that all calibration coefficients are sensor-specific
Const mC2 = 71530
Const mC1 = 7483060
Const mC0 = 1355440000
Const bC2 = 18268
Const bC1 = 309008
Const bC0 = 4270710
Const YES = -1 'Yes is defined as True or -1.
Const NO = 0 'No is defined as False or 0.
```

```
'*************************************************************************************

'*** Variables **************************************************************************
'CSAT3 variables with additional one or four scan delay, from the Data Table scan_1_4.
Public wind(5) 'Wind, temperature, and diagnostic data from CSAT3.
Alias wind(1) = Ux
Alias wind(2) = Uy
Alias wind(3) = Uz
Alias wind(4) = Ts
Alias wind(5) = diag_csat
Units wind = m/s
Units Ts = C
Units diag_csat = unitless

'LI-7500 has a fixed delay of 302.369 mSec (three scans at 10 Hz or six scans at 20 Hz).
Public irga(6) 'CO2, H2O, pressure, and diagnostic from the LI-7500.
Alias irga(1) = CO2
Alias irga(2) = H2O
Alias irga(3) = press
Alias irga(4) = diag_irga
Alias irga(5) = CO2_u_mol
Alias irga(6) = H2O_m_mol
Units CO2 = mg/m^3
Units H2O = g/m^3
Units press = kPa
Units diag_irga = unitless
Units CO2_u_mol = umol/mol
Units H2O_m_mol = mmol/mol
Public diag_bits(9) 'Warning flags.
Alias diag_bits(1) = del_T_f 'Delta temperature warning flag.
Alias diag_bits(2) = track_f 'Tracking (signal lock) warning flag.
Alias diag_bits(3) = amp_h_f 'Amplitude high warning flag.
Alias diag_bits(4) = amp_l_f 'Amplitude low warning flag.
Alias diag_bits(5) = chopper_f 'Chopper warning flag.
Alias diag_bits(6) = detector_f 'Detector warning flag.
Alias diag_bits(7) = pll_f 'PLL warning flag.
Alias diag_bits(8) = sync_f 'Synchronization warning flag.
Alias diag_bits(9) = agc 'Automatic gain control.
Units diag_bits = samples
Units agc = unitless

'Analog variables with three or six scan delay.
Public panel_temp 'CR5000 panel temperature.
```

```
Units panel_temp = C

'No delay on meteorological sensors.
Public rho_a 'Density of moist air.
Public rho_d 'Density of dry air.
Public hmp(3) 'HMP45C vapor density, temperature and relative humidity.
Public e 'HMP45C vapor pressure.
Public e_sat 'Saturation vapor pressure.
Public batt_volt
Alias hmp(1) = H2O_hmp
Alias hmp(2) = t_hmp
Alias hmp(3) = rh_hmp
Units rho_a = kg/m^3
Units rho_d = kg/m^3
Units H2O_hmp = g/m^3
Units t_hmp = C
Units rh_hmp = fraction
Units e = kPa
Units e_sat = kPa
Units batt_volt = V

'No delay energy balance sensor.
'declare public variables
Public Rn_Q7_1
Public Rn(9)
Public Temp(2)

'Public solar_rad
Public rain_fall
Public par(3)

'public variables for apogee model SI-111 Infrared Radiometers
'SBTempC = sensor body temperature in degrees Celsius
'SBTempK = sensor body temperature in Kelvin
'TargmV = mV output of thermopile infrared detector (dependent on temperature difference betwee
'body)
'm = slope of equation relating target and sensor body temperatures to mV output of thermopile
'b = intercept of the equation relating target and sensor body temperatures to mV output of the
'TargTempK = target temperature in Kelvin
'TargTempC = target temperature in degrees Celsius
Public SBTempC, SBTempK, TargmV, m, b, TargTempK, TargTempC
Dim Rs
Dim Vs_Vx
```

```
Dim i
Public hor_wind

Alias Rn(1) = Rad_short_Up
Alias Rn(2) = Rad_short_Dn
Alias Rn(3) = Rad_long_Up
Alias Rn(4) = Rad_long_Dn
Alias Rn(5) = Rad_long_Up_TCor
Alias Rn(6) = Rad_long_Dn_TCor
Alias Rn(7) = Rn_short
Alias Rn(8) = Rn_long
Alias Rn(9) = Rn_total
Alias Temp(1) = Temp_C
Alias Temp(2) = Temp_K
Alias par(1) = par_mV
Alias par(2) = par_flxdens
Alias par(3) = par_totflx

'******Units defintion*********************
Units Rn_Q7_1 = W/m^2
Units Rn = W/m^2
Units Temp(1) = Temp_C
Units Temp(2) = Temp_K
Units rain_fall = mm
Units par_mV = mV
Units par_flxdens = umol/s/m^2
Units par_totflx = mmol/m^2

'Flux variables.
Dim Fc_wpl 'CO2 flux, with Webb et al. term.
Dim LE_wpl 'Latent heat flux, with Webb et al. term.
Dim Hs 'Sensible heat flux using sonic temperature.
Dim tau 'Momentum flux.
Dim u_star 'Friction velocity.
Dim Fc_irga 'CO2 flux, raw.
Dim LE_irga 'Latent heat flux, raw.
Dim CO2_wpl_LE 'CO2 flux, Webb et al. term do to latent heat flux.
Dim CO2_wpl_H 'CO2 flux, Webb et al. term do to sensible heat flux.
Dim H2O_wpl_LE 'Latent heat flux, Webb et al. term do to latent heat flux.
Dim H2O_wpl_H 'Latent heat flux, Webb et al. term do to sensible heat flux.
Dim cov_out(51) 'Covariances of wind and scalars, wind vector, mean rho_a, rho_d, t_hmp, CO
Units Fc_wpl = mg/(m^2 s)
Units LE_wpl = W/m^2
```

```
Units Hs = W/m^2
Units tau = kg/(m s^2)
Units u_star = m/s
Units Fc_irga = mg/(m^2 s)
Units LE_irga = W/m^2
Units CO2_wpl_LE = mg/(m^2 s)
Units CO2_wpl_H = mg/(m^2 s)
Units H2O_wpl_LE = W/m^2
Units H2O_wpl_H = W/m^2

'Aliases for covariances.
Alias cov_out(1) = cov_Uz_Uz
Alias cov_out(2) = cov_Uz_Ux
Alias cov_out(3) = cov_Uz_Uy
Alias cov_out(4) = cov_Uz_CO2
Alias cov_out(5) = cov_Uz_H2O
Alias cov_out(7) = cov_Uz_Ts
Alias cov_out(9) = cov_Ux_Ux
Alias cov_out(10) = cov_Ux_Uy
Alias cov_out(11) = cov_Ux_CO2
Alias cov_out(12) = cov_Ux_H2O
Alias cov_out(14) = cov_Ux_Ts
Alias cov_out(16) = cov_Uy_Uy
Alias cov_out(17) = cov_Uy_CO2
Alias cov_out(18) = cov_Uy_H2O
Alias cov_out(20) = cov_Uy_Ts
Alias cov_out(22) = cov_CO2_CO2
Alias cov_out(27) = cov_H2O_H2O
Alias cov_out(34) = cov_Ts_Ts
Units cov_Uz_Uz = (m/s)^2
Units cov_Uz_Ux = (m/s)^2
Units cov_Uz_Uy = (m/s)^2
Units cov_Uz_CO2 = mg/(m^2 s)
Units cov_Uz_H2O = g/(m^2 s)
Units cov_Uz_Ts = m C/s
Units cov_Ux_Ux = (m/s)^2
Units cov_Ux_Uy = (m/s)^2
Units cov_Ux_CO2 = mg/(m^2 s)
Units cov_Ux_H2O = g/(m^2 s)
Units cov_Ux_Ts = m C/s
Units cov_Uy_Uy = (m/s)^2
Units cov_Uy_CO2 = mg/(m^2 s)
Units cov_Uy_H2O = g/(m^2 s)
```

```
Units cov_Uy_Ts = m C/s
Units cov_CO2_CO2 = (mg/m^3)^2
Units cov_H2O_H2O = (g/m^3)^2
Units cov_Ts_Ts = C^2

'Wind directions and speed.
Alias cov_out(39) = wnd_dir_compass
Alias cov_out(41) = wnd_spd
Alias cov_out(42) = rslt_wnd_spd
Alias cov_out(43) = wnd_dir_csat3
Alias cov_out(44) = std_wnd_dir
Alias cov_out(45) = rho_a_mean
Alias cov_out(46) = rho_d_mean
Alias cov_out(47) = t_hmp_mean
Alias cov_out(48) = CO2_mean
Alias cov_out(49) = H2O_mean
Units wnd_dir_compass = degrees
Units wnd_spd = m/s
Units rslt_wnd_spd = m/s
Units wnd_dir_csat3 = degrees
Units std_wnd_dir = degrees

'Diagnostic variables.
Dim disable_flag_on(4) 'Intermediate processing disable.
'disable_flag_on(1) 'Set high when CSAT3 diagnostic warning flags are on or CSAT3 has no da
'disable_flag_on(2) 'Set high when LI-7500 diagnostic warning flags are on or LI-7500 faile
'disable_flag_on(3) 'Set high when CSAT3 diagnostic warning flags are on.
' Used to filter the sum of CSAT3 diagnostic warning flags.
'disable_flag_on(4) 'Set high when LI-7500 diagnostic warning flags are on.
' Used to filter the sum of LI-500 diagnostic warning flags.
Dim n 'Number of samples in the on-line covariances.
Units n = samples

'Program Control flags.
Public save_flx_flag_on
Public save_ts_flag_on

'Measurement variables without delays.
Dim wind_in(5) 'CSAT3 data, before adding delay.
Dim panel_temp_in 'Panel temperature, before adding delay.

'Arrays to store delayed data.
Dim analog_data(3) 'Three or six scan old data from the Data Table scan_3_6.
```

```
Dim csat_data(5) 'One or four scan old data from the Data Table scan_1_4.

'Working variables.
Dim cov_in(8) 'Array used in the covariance instruction.
Dim CO2_mm_m3 'Carbon dioxide concentration [mmol/m^3], used to compute umol/mol concentr
Dim H2O_mm_m3 'Water vapor concentration [mmol/m^3], used to compute vapor pressure and m
Dim t_hmp_K 'HMP45C temperature in Kelvin.
Dim sigma_wpl 'Webb et al. sigma = density of water / density of dry air.
Dim j 'Counter variable.
Dim scan_count 'Counts the number scans that have been executed.
Dim wind_east 'Uy wind in compass coordinate system.
Dim wind_north 'Ux wind in compass coordinate system.
Dim even_min_flag_on 'Used to synchronize the time series output to the even minute.
'************************************************************************************

'*** Final Output Data Tables ***********************************************************
'This table will hold fourteen days of flux data on the CPU or twenty-eight days on the
'PC Card of flux data. This data is output every OUTPUT_INTERVAL minutes.
DataTable (flux,TRUE,2016)
DataInterval (0,OUTPUT_INTERVAL,Min,10)
CardOut (0,-1)
Sample (1,Fc_wpl,IEEE4)
Sample (1,LE_wpl, IEEE4)
Sample (1,Hs,IEEE4)
Sample (1,tau,IEEE4)
Sample (1,u_star,IEEE4)
Sample (5,cov_out(1),IEEE4)
Sample (1,cov_out(7),IEEE4)
Sample (4,cov_out(9),IEEE4)
Sample (1,cov_out(14),IEEE4)
Sample (3,cov_out(16),IEEE4)
Sample (1,cov_out(20),IEEE4)
Sample (1,cov_out(22),IEEE4)
Sample (1,cov_out(27),IEEE4)
Sample (1,cov_out(34),IEEE4)

Average (3,Ux,IEEE4,(disable_flag_on(1) OR NOT (save_flx_flag_on))
Average (2,CO2,IEEE4,(disable_flag_on(2) OR NOT (save_flx_flag_on))
Average (1,Ts,IEEE4,(disable_flag_on(1) OR NOT (save_flx_flag_on))
Average (1,rho_a,IEEE4,disable_flag_on(2) OR NOT (save_flx_flag_on))
Average (1,press,IEEE4,disable_flag_on(2) OR NOT (save_flx_flag_on))
Average (1,panel_temp,IEEE4,FALSE)
```

```
Sample (1,wnd_dir_compass,IEEE4)
Sample (1,wnd_dir_csat3,IEEE4)
Sample (1,wnd_spd,IEEE4)
Sample (1,rslt_wnd_spd,IEEE4)
Average (1,batt_volt,IEEE4,FALSE)
Sample (1,std_wnd_dir,IEEE4)
Totalize (1,n,IEEE4,disable_flag_on(1) OR disable_flag_on(2) OR NOT (save_flx_flag_on))
Totalize (1,n,IEEE4,NOT (disable_flag_on(1) XOR disable_flag_on(3)) OR NOT (save_flx_flag_on)

FieldNames ("csat_warnings")
Totalize (1,n,IEEE4,NOT (disable_flag_on(2) XOR disable_flag_on(4)) OR NOT (save_flx_flag_on)
FieldNames ("irga_warnings")
Totalize (4,del_T_f,IEEE4,disable_flag_on(3) OR NOT (save_flx_flag_on))
Totalize (4,chopper_f,IEEE4,disable_flag_on(4) OR NOT (save_flx_flag_on))
Average (1,agc,IEEE4,disable_flag_on(2) OR NOT (save_flx_flag_on))

Sample (1,Fc_irga,IEEE4)
Sample (1,LE_irga,IEEE4)
Sample (1,CO2_wpl_LE,IEEE4)
Sample (1,CO2_wpl_H,IEEE4)
Sample (1,H2O_wpl_LE,IEEE4)
Sample (1,H2O_wpl_H,IEEE4)
Average (3,H2O_hmp,IEEE4,FALSE) ' modified to add output of rh
Average (1,Rn_Q7_1,IEEE4,FALSE)
Average (9,Rn,IEEE4,False)
Average (1,Temp(2),IEEE4,False)
Totalize (1,rain_fall,FP2,False)

'*********apogee model SI-111 output********
Average (1,SBTempC,IEEE4,False)
Average (1,TargTempC,IEEE4,False)
'===(deleted)===Average (6,hmp(4),IEEE4,False)
Totalize (1,par_totflx,IEEE4,False)
Average (1,par_flxdens,IEEE4,False)
EndTable

'Set save_ts_flag_on high or set "Save Time Series?" custom menu option to Yes
'to save time series data.
DataTable (ts_data,even_min_flag_on,-1)
DataInterval (0,SCAN_INTERVAL,mSec,100)
CardOut (0,-1)
Sample (3,Ux,IEEE4)
Sample (2,CO2,IEEE4)
```

```
Sample (1,Ts,IEEE4)
Sample (1,press,IEEE4)
Sample (1,diag_csat,IEEE4)
EndTable

'*** Working Data Tables ***
'Delay the analog measurements by three or six scans.
DataTable (scan_3_6,TRUE,ANALOG_DELAY)
Sample (1,panel_temp_in,IEEE4)
EndTable

'Delay the CSAT3 measurements by one or four scans.
DataTable (scan_1_4,TRUE,CSAT_DELAY)
Sample (5,wind_in(1),IEEE4)
EndTable

'Compute the covariances of vertical wind, CO2, H2O, and sonic temperature, as well as
'the other cross products, required to rotate the data into natural wind coordinates.
'This data is output every OUTPUT_INTERVAL minutes.
DataTable (comp_cov,TRUE,1)
DataInterval (0,OUTPUT_INTERVAL,min,1)
Covariance (8,cov_in(1),IEEE4,(disable_flag_on(1) OR disable_flag_on(2) OR NOT (save_flx_flag
WindVector (1,wind_east,wind_north,IEEE4,(disable_flag_on(1) OR NOT (save_flx_flag_on)),0,1,2
WindVector (1,Uy,Ux,IEEE4,(disable_flag_on(1) OR NOT (save_flx_flag_on)),0,1,2)
Average (1,rho_a,IEEE4,disable_flag_on(2))
Average (1,rho_d,IEEE4,disable_flag_on(2))
Average (1,t_hmp,IEEE4,FALSE)
Average (1,CO2,IEEE4,disable_flag_on(2))
Average (1,H2O_hmp,IEEE4,FALSE)
EndTable

'*** Subroutines ***
'*** Define Custom Menus ***
DisplayMenu ("Program Control",TRUE)
MenuItem ("Save Fluxes?",save_flx_flag_on)
MenuPick (YES,NO)
MenuItem ("Save Time Series?",save_ts_flag_on)
MenuPick (YES,NO)
EndMenu
************************************************************************

'*** Program ***************************
BeginProg
```

```
n = 1
'Save fluxes on compile.
save_flx_flag_on = TRUE
'Save the time series on compile.
save_ts_flag_on = TRUE

'Set all CSAT3 variables to NaN.
Move (wind_in(1),5,NaN,1)

'Set all LI-7500 variables to NaN.
Move (irga(1),4,NaN,1)

'Set the SDM clock speed.
SDMSpeed (SDM_PER)
Scan (SCAN_INTERVAL,mSec,10,0)

'CR5000 panel temperature.
PanelTemp (panel_temp_in,250)

'Get CSAT3 wind and sonic temperature data.
CSAT3 (wind_in(1),1,3,91,CSAT_OPT)

'Get LI-7500 data.
CS7500 (irga(1),1,7,6)

'Save the molar density to compute molar concentration.
CO2_mm_m3 = CO2
H2O_mm_m3 = H2O

'Convert LI-7500 data from molar density [mmol/m^3] to mass density.
' 44 [g/mol] - molecular weight of carbon dioxide
' 0.018 [g/mmol] - molecular weight of water vapor
If ( NOT (CO2 = -99999) ) Then ( CO2 = CO2*44 )
H2O = H2O*0.018

'Measure the HMP45AC temperature and fraction humidity.
VoltDiff(t_hmp,1,mV1000,9,TRUE,200,250,0.1,0)
VoltDiff(rh_hmp,1,mV1000,10,TRUE,200,250,0.1,0)

'Find the engineering units for the HMP45C temperature and humidity.
t_hmp = t_hmp - 40
rh_hmp = rh_hmp*0.01
```

```
'Find the HMP45C vapor pressure, in kPa, using a sixth order polynomial (Lowe, 1976).
e_sat = 0.1*(A_0+t_hmp*(A_1+t_hmp*(A_2+t_hmp*(A_3+t_hmp*(A_4+t_hmp*(A_5+t_hmp*A_6))))))
e = e_sat*rh_hmp

'Compute the HMP45C vapor density.
H2O_hmp = e/((t_hmp+273.15)*RV)

'Compute dry and moist air density.
rho_d = (press-e)/((t_hmp+273.15)*RD)
rho_a = rho_d+(H2O_hmp/1000)

'Measure battery voltage.
Battery (batt_volt)

'Measure the Q7.1 net radiometer.
VoltDiff (Rn_Q7_1,1,mV200C,8,TRUE,200,250,1,0)

'Measure the CNR4 Net Radiometer
'Measure short wave up radiometer
VoltDiff (Rn(1),1,mV50,1,True,200,250,pyrano_up_mult,0)
'Measure short wave dn radiometer
VoltDiff (Rn(2),2,mV50,1,True,200,250,pyrano_dn_mult,0)
'Measure long wave up radiometer
VoltDiff (Rn(3),3,mV50,1,True,200,250,pyrgeo_up_mult,0)
'Measure long wave dn radiometer
VoltDiff (Rn(4),4,mV50,1,True,200,250,pyrgeo_dn_mult,0) 'change on 7/16 from Rn(3) to R

'Measure CNR4 thermistor
BrHalf (Vs_Vx,1,mV5000,10,Vx3,1,2500,True,0,250,1.0,0)
Rs = 1000*(Vs_Vx/(1-Vs_Vx))
Temp_C = 1/(1.0295*(10^(-3))+2.391*(10^(-4))*LN(Rs)+1.568*(10^(-7))*LN(Rs))-273.15

'Convert Temperature to Kelvin
Temp_K = Temp_C+273.15

'Apply Temperature Corrections to CNR4 Long Wave Radiation Outputs
Rn(5) = Rn(3)+5.67*10^(-8)*Temp_K^4
Rn(6) = Rn(4)+5.67*10^(-8)*Temp_K^4

'Caclulate Short Wave Net Radiation
Rn_short = Rad_short_Up - Rad_short_Dn

'Calculate Long Wave Net Radiation
```

```
Rn_long = Rad_long_Up - Rad_long_Dn

'Calculate Total Net Radiation
Rn_total = Rn_short + Rn_long
'****** end of the CNR4 measurement***********

'Measure LI190SB Quantum sensor
VoltDiff (par_mV,1,mV20,7,True,200,250,1,0)

'Calculate Flux Density & Total Flux
par_flxdens = par_mV*par_mult_flxdens
par_totflx = par_mV*par_mult_totflx

'Measure TE525 Tipping Bucket Rain gage and report the rain fall in mm
PulseCount (rain_fall,1,1,2,0,0.254,0)

'measure apogeen model SI-111
'Therm109 added by Jason Ritter, CSI, 10/17/17 so that BrHalf & SBTempC instructions could be c
Therm109 (SBTempC,1,23,Vx2,0,250,1.0,0)

''Commented out for testing 10/17/17 by Jason Ritter, CSI
'measure sensor body temperature (green wire to SE23, white wire to EX1, blue wire to ground)
''BrHalf(SBTempC,1,mV5000,23,VX1,1,2500,TRUE,200,_60Hz,1.0,0)
'BrHalf measures the ratio of the measured voltage divided by the excitation voltage, use the f
'first convert to resistance
''SBTempC=24900*(1/SBTempC-1)
'second the Steinhart and Hart equation is used to calcualte temperature from resistance
''SBTempC=1.129241*(10^(-3))+2.341077*(10^(-4))*LN(SBTempC)+8.775468*(10^(-8))*((LN(SBTempC))
^3
''SBTempC=1/SBTempC-273.15

'measure mV output of thermopile detector (red wire to 13H, black wire to 13L, clear wire to gr
VoltDiff (TargmV,1,mV20,13,True,200,250,1.0,0)

'Calculation of m (slope) and b (intercept) coefficients for target temperature calculation
m = mC2 * SBTempC^2 + mC1 * SBTempC + mC0
b = bC2 * SBTempC^2 + bC1 * SBTempC + bC0

'Calculation of target temperature
SBTempK = SBTempC + 273.15
TargTempK = ((SBTempK^4) + m * TargmV + b)^0.25
TargTempC = TargTempK - 273.15
```

```
'Delay the analog measurements by three or six scans.
CallTable scan_3_6

'Delay the CSAT3 measurements by one or four scans.
CallTable scan_1_4
If ( scan_count >= ANALOG_DELAY ) Then

'Load in analog measurements that have been delayed by three or six scans.
GetRecord (analog_data(1),scan_3_6,ANALOG_DELAY)
panel_temp = analog_data(1)

'Load in CSAT3 measurements that have been delayed by one or four scans.
GetRecord (Ux,scan_1_4,CSAT_DELAY)
wind_east = -1*Uy
wind_north = Ux

'Define 61502 in the diagnostic word as NaN.
If ( diag_csat = NaN ) Then ( diag_csat = 61502 )

'Break up the four CSAT3 warning flags into four separate bits.
del_T_f = (diag_csat AND &h8000)/&h8000
track_f = (diag_csat AND &h4000)/&h4000
amp_h_f = (diag_csat AND &h2000)/&h2000
amp_l_f = (diag_csat AND &h1000)/&h1000
'Turn on the intermediate processing disable flag when any CSAT3 warning flag is high, inc
'special cases NaN (61502), a Lost Trigger (61440), No Data (61503), an SDM error (61441),
'embedded code (61442).
If ( diag_csat AND &hf000 ) Then
disable_flag_on(1) = TRUE
Else
disable_flag_on(1) = FALSE
EndIf

'Totalize the CSAT3 diagnostic warning flags only. Turn on the intermediate processing di
'the CSAT3 special cases NaN (61502), a Lost Trigger (61440), No Data (61503), an SDM erro
'or wrong CSAT3 embedded code (61442).
If ( (diag_csat AND &hf000) = &hf000 ) Then
disable_flag_on(3) = TRUE
Else
disable_flag_on(3) = FALSE
EndIf
'Save only the four most significant bits of the CSAT3 diagnostics, except for the special
'NaN (61502), a Lost Trigger (61440), No Data (61503), an SDM error (61441), or wrong CSAT
```

```
'embedded code (61442).
If ( diag_csat < &hf000 ) Then ( diag_csat = INT ((diag_csat AND &hf000)/&h1000) )

'Swap the LI-7500 diagnostic bit state.
diag_irga = diag_irga XOR &h00f0

'Turn on the intermediate processing disable flag when the LI-7500 has failed to send data
'CR5000 via SDM.
If ( (CO2 < -99990) OR (CO2 = NaN) ) Then (diag_irga = &h00ff) 'Set all flags high and

'Compute the AGC.
agc = INT ((diag_irga AND &h000f) * 6.25 + 0.5)
'Break up the four LI-7500 warning flags into four separate bits and swap bit state.
chopper_f = (diag_irga AND &h0080)/&h0080
detector_f = (diag_irga AND &h0040)/&h0040
pll_f = (diag_irga AND &h0020)/&h0020
sync_f = (diag_irga AND &h0010)/&h0010

'Turn on the intermediate processing disable flag when any LI-7500 warning flag is high, i
'special cases NaN or an SDM error.
If ( diag_irga AND &h00f0 ) Then
disable_flag_on(2) = TRUE
Else
disable_flag_on(2) = FALSE
EndIf

'Totalize the LI-7500 diagnostic warning flags only. Turn on the intermediate processing
' the LI-7500 special cases NaN (255) or SDM error (255).
If ( (diag_irga AND &h00ff) = &h00ff ) Then
disable_flag_on(4) = TRUE
Else
disable_flag_on(4) = FALSE
EndIf

'Save only the four most significant bits of the LI-7500 diagnostic word.
diag_irga = INT ((diag_irga AND &h00f0)/&h0010)

'Apply calibration and wind correction to net radiometer measurement.
If ( NOT (disable_flag_on(1)) ) Then
hor_wind = SQR (Ux*Ux+Uy*Uy)
If (Rn_Q7_1 > 0) Then
Rn_Q7_1 = Rn_Q7_1*Q7_1_POS_CAL*(1+(0.0132*hor_wind)/(0.066+(0.2*hor_wind)))
Else
```

```
Rn_Q7_1 = Rn_Q7_1*Q7_1_NEG_CAL*((0.00174*hor_wind)+0.99755)
EndIf
Else
If (Rn_Q7_1 > 0) Then
Rn_Q7_1 = Rn_Q7_1*Q7_1_POS_CAL*1.045
Else
Rn_Q7_1 = Rn_Q7_1*Q7_1_NEG_CAL
EndIf
EndIf

'Compute the molar concentration of CO2 and H2O.
t_hmp_K = t_hmp+273.15
CO2_u_mol = CO2_mm_m3*R*t_hmp_K/press*1000
H2O_m_mol = H2O_mm_m3*R*t_hmp_K/press

'Write a file mark to the time series table every twenty-four hours with a five hour offse
'The filemark is written only to the PC Card if time series data are being stored.
If ( even_min_flag_on AND IfTime (300,1440,Min) ) Then ( FileMark (ts_data) )

'Start saving the time series data on a even minute boundary.
If ( save_ts_flag_on ) Then
If ( IfTime (0,1,Min) ) Then ( even_min_flag_on = TRUE )
Else
even_min_flag_on = FALSE
EndIf
CallTable ts_data

'Load cov_in() array for the covariance computation.
cov_in(1) = Uz
cov_in(2) = Ux
cov_in(3) = Uy
cov_in(4) = CO2
cov_in(5) = H2O
cov_in(7) = Ts
CallTable comp_cov
If ( comp_cov.Output(1,1) ) Then
GetRecord (cov_out(1),comp_cov,1)

If ( comp_cov.Output(1,1) ) Then
GetRecord (cov_out(1),comp_cov,1)
'The compass wind direction will be between 0 and 360 degrees.
wnd_dir_compass = (wnd_dir_compass+ANGLE_FROM_NORTH)
If ( wnd_dir_compass ) < 0 Then ( wnd_dir_compass = wnd_dir_compass+360 )
```

```
wnd_dir_compass = wnd_dir_compass MOD 360
'The CSAT3 wind direction will be between 0 to 180 degrees and 0 to -180 degrees.
If ( wnd_dir_csat3 ) > 180 Then ( wnd_dir_csat3 = wnd_dir_csat3-360 )

'Compute on-line fluxes.
Fc_irga = cov_Uz_CO2
LE_irga = LV*cov_Uz_H2O
Hs = rho_a_mean*CP*cov_Uz_Ts
tau = SQR ((cov_Uz_Ux*cov_Uz_Ux)+(cov_Uz_Uy*cov_Uz_Uy))
u_star = SQR (tau)
tau = rho_a_mean*tau

'Convert dry air density to g/m^3.
rho_d_mean = 1e3*rho_d_mean
sigma_wpl = H2O_mean/rho_d_mean
'Webb et al. term for water vapor Eq. (25).
H2O_wpl_LE = 1.61*sigma_wpl*LE_irga
H2O_wpl_H = (1+(1.61*sigma_wpl))*H2O_mean/(t_hmp_mean+273.15)*LV*cov_Uz_Ts
'Webb et al. term for carbon dioxide Eq (24).
CO2_wpl_LE = 1.61*CO2_mean/rho_d_mean*cov_Uz_H2O
CO2_wpl_H = (1+(1.61*sigma_wpl))*CO2_mean/(t_hmp_mean+273.15)*cov_Uz_Ts
LE_wpl = LE_irga+H2O_wpl_LE+H2O_wpl_H
Fc_wpl = Fc_irga+CO2_wpl_LE+CO2_wpl_H
EndIf
CallTable flux
Else
scan_count = scan_count+1
EndIf
NextScan
EndProg
```