12-2021

Greenhouse Tomatoes: Process Simulation

Juan Gabriel Marin Jr.
University of Arkansas, Fayetteville

Citation

Marin, J. G. (2021). Greenhouse Tomatoes: Process Simulation. *Graduate Theses and Dissertations* Retrieved from https://scholarworks.uark.edu/etd/4256

Greenhouse Tomatoes: Process Simulation

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Chemical Engineering

by

Juan Gabriel Marin Jr
University of Arkansas
Bachelor of Science in Chemical Engineering, 2019

December 2021
University of Arkansas

This thesis is approved for the recommendation to the Graduate Council.

____
Greg Thoma, PhD
Thesis Director

____
Lauren Greenlee, PhD
Committee Member

____
Marty Matlock, PhD
Committee Member

ABSTRACT

Growing population demand and challenges brought on by climate change have spurred the need for more resilient fruit and vegetable supply chains. One agricultural technology of significant interest is the use of greenhouses for food production. Greenhouses create a stable and adaptable environment for crops such as tomatoes to grow year-round. Fresh tomatoes are the second most consumed vegetable per capita in U.S. diets, currently averaging 20.7 pounds. The growing consumption of fresh tomatoes has been the result of increasing cultural diversity in the United States.

To meet the growing demand, Venlo-type greenhouses have been frequently used by growers. It provides an economical solution to produce multiple crops in various climate environments while withstanding severe weather conditions. While there have been many studies and advancements in using greenhouse technology to grow tomatoes in Europe, production has yet to be analyzed in the U.S. This study seeks to fill the gap of greenhouse tomato production by simulating growing scenarios using the openly accessible Modelica Greenhouse Library in 10 select locations across the mainland USA.

Two growing scenarios were explored, a base case without CO_2 enrichment and a CO_2+ case with a continuous flow of CO_2 being externally supplied to the system. All simulations had yields above 3-8 $kg/m^2/yr$, which is expected of field-grown tomatoes. However, a few locations were below the expected range of 50-80 $kg/m^2/yr$ yield for greenhouse-grown tomatoes. CO_2 enrichment in most cases resulted in increased fresh weight yield, reduced the use of resources which improved Product Water Use (PWU), Electrical and Thermal Energy Efficiency. Factors influencing the model such as Temperature, Supplemental Lighting, and CO_2 enrichment were discussed.

ACKNOWLEDGMENTS

I am sincerely grateful to Dr. Lauren Greenlee for her council and guidance in pursuing graduate school. She has constantly motivated me to do my best and explore the world of possibilities engineering offers throughout my academic career. Thank you for being on my committee and helping me become a better professional.

I want to thank my advisor, Dr. Greg Thoma, for giving me the opportunity to grow as a young engineer and researcher. He took a risk on me with minimal research experience to be a part of his research group, and allowed me to pursue my leadership and management goals. His patience and assistance have helped me acquired numerous skills necessary to thrive in this ever-changing world.

I want to thank Dr. Marty Matlock for your wise words at the start of my research project and for being one of my committee members. He had many thoughtful suggestions for class electives that would benefit me and my career.

I want to thank Dr. Ranjan Parajuli for being a great postdoc and guiding me throughout this project. His feedback helped me to navigate this complex journey easier and see where the path was leading.

I want to thank Dr. Vincent Lemont and Queralt Altés Buch for your kind assistance in helping me understand the model’s components in the early stages and other questions I had.

I want to express my deepest gratitude to Dr. Eric Specking for being a phenomenal mentor and friend throughout this venture. He has been instrumental in me pursuing my most ambitious and audacious goals as well as a voice of reason throughout graduate school and life. I hope that every person would have a mentor with boundless wisdom on their journey.

I want to thank Dr. Tammy Lutz-Rechtin for your continuous support and wisdom. Those hallway conversations and check-ins were crucial to me becoming a better student and person. I am grateful for you helping me appreciate the role of safety in the workplace and routine tasks.

I want to thank Dr. Carl Laird for helping me better appreciate the world and chemical engineers' role in it. Our numerous conversations during the first semester were vital for me to become the engineer and leader I wanted to be.

I want to thank Karen Beitle for helping me throughout the first year and baking those wonderful cookies, especially the caramel one during finals week.

I want to thank Ben Putman for helping me understand and appreciate research in Sustainability and Life Cycle Analysis.

I want to thank my colleagues in the graduate program in the department; I am grateful for all the laughs, joy, and knowledge you all have shared with me. Each of your conversations helped shape my research experience and my professional and personal development. Rogelio Garcia, your guidance in research techniques and assistance with MATLAB has been crucial in teaching and developing this thesis. Jessica Bain, I can't thank you enough for being my friend and mentor throughout the experience. Those many days in your office working, laughing, and singing were worth it. Martin Christie, your assistance and key insights in my first year were vital for the graduate program and research group. Zakary Ford, you helped me find my footing during those first days of the first year. Shadrach Ibinola, you provided helpful feedback and kind words on numerous occasions. Joshua Corbitt, you were one of my first graduate student friends during my senior year of undergrad, and continued to be so into grad school. You helped me understand the adventure I was embarking on and inspired me to continue to do my best.

Karla Morrissey, you helped me feel like I belonged in the research group when I was lost and an all-around great friend during grad school.

I want to thank the University Staff such as Laura Moix, Wendy Echeverria, and Kelsey Lavigne, and many others who helped and talked to me over the years. I valued all those moments and joy we shared. You all have made the experience worth living.

I want to thank my teachers who have taught me over the years, as your advice and wisdom have helped me in many moments on this journey.

I want to thank my family for their love and support all these years. Especially my grandparents, who have always pushed for me to do my best and not give up when the days get hard. Your calls always brought me joy and hope when I most needed them. I can’t thank my parents enough for their continued support and patience while studying abroad. I could not be the person I am without your guidance. Whether mentioned here explicitly or not, everyone I have met has helped me grow and become a better me, so I must say thank you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

T_out- Temperature of the outside air

RH_out- Relative Humidity of the outside air

P_out- Pressure of the outside air

I_glob- Global irradiation

u_wind- Wind speed of outside air

T_sky- Sky Temperature

T_air_sp- temperature setpoint of the greenhouse inside air

CO2_air_sp—Carbon dioxide set-point

ilu_sp- sunlight illumination set-point

CO2_sp- Carbon dioxide set-point

T_sp- temperature set-point

SC_usable- time when the thermal screen is deployed

SP- set-point

th-thermal

el-electrical

ΔT- temperature difference

CHP- Combined Heat and Power unit

H.P.- Heat Pump

CRD- Crop Reporting District

CO_2- Carbon Dioxide

Tamb- ambient temperature

Tam- atmospheric temperature

PWU- Product Water Use

CO$_2$+- CO$_2$ enrichment

HPS- High Pressure Sodium lamps

LED- Light Emitting Diode

PAR- Photosynthetically active radiation

Chapter 1: Introduction

The demand for easily accessible fresh fruits and vegetables has risen in recent years, potentially hindered by climate change.[1] Current fruit and vegetable production methods such as open-field tomato farms are impacted by various natural and human factors such as changing weather patterns, increasing nutritional demands, and growing crop requirements.[1] Tomatoes are the second most consumed fresh market vegetable per capita.[2] Fresh market tomatoes per capita availability has been steadily increasing since the 1980s and has averaged around 20.7 pounds in 2010-2017.[2,3]The rise in consumption is due to increase cultural diversity in the United States, with the Hispanic population consuming more than any other ethnic group.[2, 4] Consumers have been shifting their diet towards a more balanced and nutritious lifestyle.[2,5,6,7] The prime advantage of greenhouse technology is that more food can be produced in the same space as an open field farm. Crops would also be more resilient to seasonal changes in the external environment, potentially reducing crop losses. This study seeks to analyze the potential of tomato greenhouse production by modeling crop growth in 10 select locations across the U.S. Thus, informing a variety of stakeholders involved in fruit and vegetable supply chains to make critical decisions about the current and future food supply outlook.

Tomato production in the 2010s averaged about 3.3 billion pounds of field-grown tomatoes per year in 2010-2016.[2] Imported tomatoes currently augment U.S. production and provide consumers with year-round access to supplies of diverse varieties of fresh tomatoes.[2] Tomato production is primarily concentrated in California and Florida, accounting for about 80 percent of U.S. field-grown fresh tomatoes.[2] U.S. production peaks around late May and declines in mid-to-late summer before reaching seasonal lows from July to October.[2] Another spike occurs in the late fall when Florida production reaches the market.[2]

Greenhouse technology can supplement the supply of field-grown tomatoes and provide a strategy for climate adaptation and mitigation in the fruit and vegetable supply chain. Greenhouse production and other protected-culture technologies help extend the growing season and make production feasible for a wider variety of geographic locations.[2,8] In 2000, USDA Agricultural Marketing Service (AMS) reported that no shipments of greenhouse-grown tomatoes.[2] From 2005 to 2012, AMS reports that U.S. greenhouse-grown tomatoes shipments grew steadily to 475 million pounds annually. By 2017, greenhouse-grown tomatoes constituted more than 5 percent of U.S. shipments.[2] However, that share is likely understated because of the withdrawal of a major US shipper from the AMS voluntary reporting process after 2013, where shipments fell to around 200 million pounds and have remained so in recent years.[2] Some greenhouse production is clustered in traditional field-grown-tomato-producing states like California, which produces over 10 million lbs of greenhouse tomatoes, according to the USDA, National Agricultural Statistics Service (NASS) *2014 Census of Horticulture Specialties.*[2] Nontraditional market leaders such as Nebraska, Minnesota, New York also produce over 10 million lbs of greenhouse tomatoes each.[2] Greenhouse tomatoes have greater market access both in the off-season and in northern retail produce markets, better product consistency, and improved yields. Greenhouse-grown tomatoes primarily consist of cherry and grape varieties.[2]

Mexico's share of the greenhouse tomato market has grown steadily, averaging 35 percent annual growth, outpacing shipments of field-grown fresh tomatoes from Mexico to the United States.[2] In 2017, imports from Mexico accounted for almost 84 percent (1.8 billion pounds) of the greenhouse volume coming into the U.S. market.[2] Imports of Canadian greenhouse-grown tomatoes, approximated by fresh market tomato shipments to the United

States, have remained at about 300 million pounds.[2,9,10] Fresh tomatoes from Mexico (field and greenhouse) accounted for over 90 percent of total fresh tomato imports in 2016.[2,3]

A greenhouse climate model describes the indoor climate resulting from the greenhouse design, the outdoor climate, and specific control. The indoor climate is characterized by temperature, the vapor pressure of water, and the CO_2 concentration of the indoor air. Together with the temperature of the heating pipes, the indoor climate constitutes the climate controller feedback variables. Other variables with an indirect effect on the climate need to be modeled, such as those relative to the canopy and the envelope (i.e., the cover, the floor, and the thermal screen). The canopy temperature impacts the rate of photosynthesis and transpiration, decreasing the CO_2 concentration and increasing the air's vapor content, respectively. Condensation on surfaces may occur depending on the water vapor pressure difference. The temperature of the envelope influences the vapor pressure of water in the air, which is decreased by condensation at the cover and the thermal screen. The thermal screen is a membrane used to reduce the energy required to heat the greenhouse. Air and moisture are exchanged through the fabric due to its porous nature. Air exchange with the outside environment decreases the vapor pressure of water and the CO_2 concentration in the air, which can be increased by externally supplied CO_2.[11]

Many greenhouse climate models have been proposed in literature; however, most are designed for specific locations and greenhouse structures.[12-14]A generic model has been developed, which has combined previous works and validated for a range of climates and designs.[12, 14-16] The model uses a Venlo design greenhouse used by professional growers, which provides an economical solution to produce multiple crops in various climate environments while withstanding severe weather conditions.[17] The simulation model was implemented in the Modelica Language and featured a wide range of tools in the Greenhouse Library.[16, 18]

Two growing scenarios were explored in each of the 10 locations in this study, a base case without CO_2 enrichment and a CO_2+ case with a continuous flow of CO_2 being externally supplied to the system. The simulated results are expected to exceed the yield of field-grown tomatoes, which ranges from 3-8 kg/m^2/yr. The estimated yield for greenhouse-grown tomatoes should range from 50-80 kg/m2/yr yield.[19] CO_2 enrichment supplemented to the system should increase the yield of tomatoes by increasing crop growth during the simulated period.[19] The study looked at tomato yield performance while assessing factors that impact the feasibility of large-scale greenhouse production, such as electrical and thermal efficiency, Product Water Usage (PWU), and Carbon footprint. Two example models are used for production analysis based on typical values for Venlo-type greenhouse to achieve this result. Greenhouse_1 assumes that electrical and heating energy is added from an external source while Global System_2 adds a CHP unit and a Heat pump (H.P.) to supplement the needs for operation.[18] Both models helped simulate results for the various Crop Reporting Districts (CRDs) of interest. Emergent factors influencing the simulation were analyzed and explored for the impact on tomato production.

Chapter 2: Methods

2.1 Greenhouse Model

Tomatoes cultivated utilizing greenhouse technology were simulated using sample models provided by the open-source Greenhouses Library developed for the Modelica Language.[18] The library has numerous components that can simulate a variety of greenhouse structures and climates. The two example models used in this study were Greenhouse_1 and GlobalSystem_2 .[11, 14, 20] Figure 1 shows the overview of the state variables of the canopy, air zones, and the envelope, which is comprised of the cover, floor, and thermal screen in the greenhouse climate model.[11, 14] Figure 2-3 shows an expanded view of components and the graphical interface in the greenhouse model of the two models used. All the sub-models such as the Solar Model and Canopy (highlighted) in Figure 2 are shown to perform the energy and mass balances to calculate the temperature, vapor pressure of water, and CO_2 concentration of the state variables housed in Greenhouse_1.[14] The assumptions for the structure are illustrated in Table 1 which is of Venlo-type greenhouse construction design dedicated to tomato crop cultivation. Greenhouse_1 consists of two levels of heating circuits, roof windows, natural ventilation, and a movable thermal screen. When the thermal screen is drawn, the air is divided into two zones, which are assumed to be homogenous. GlobalSystem_2 expands on Greenhouse_1 by connecting a CHP unit, thermal energy storage tank (TES) and a heat pump shown in Figure 3. The thermal energy storage tank is sized to provide a storage period of 8 hrs. and given a nominal temperature difference (ΔT) between the inlet and outlet. The heat pump is sized to equal the excess of electricity produced by the CHP in nominal conditions. The heating demand is produced by both the heat pump and the CHP in nominal conditions. A heat-driven controller decides when to run the CHP and the heat pump.[14] Table 2 lists the parameters for the greenhouse heating system used in the

GlobalSystem_2, which describes the capacity of the thermal storage tank, the thermal power of the CHP unit, and the heat pump. The source code for Greenhouse_1, and Global System_2 are located in **Appendix B**.

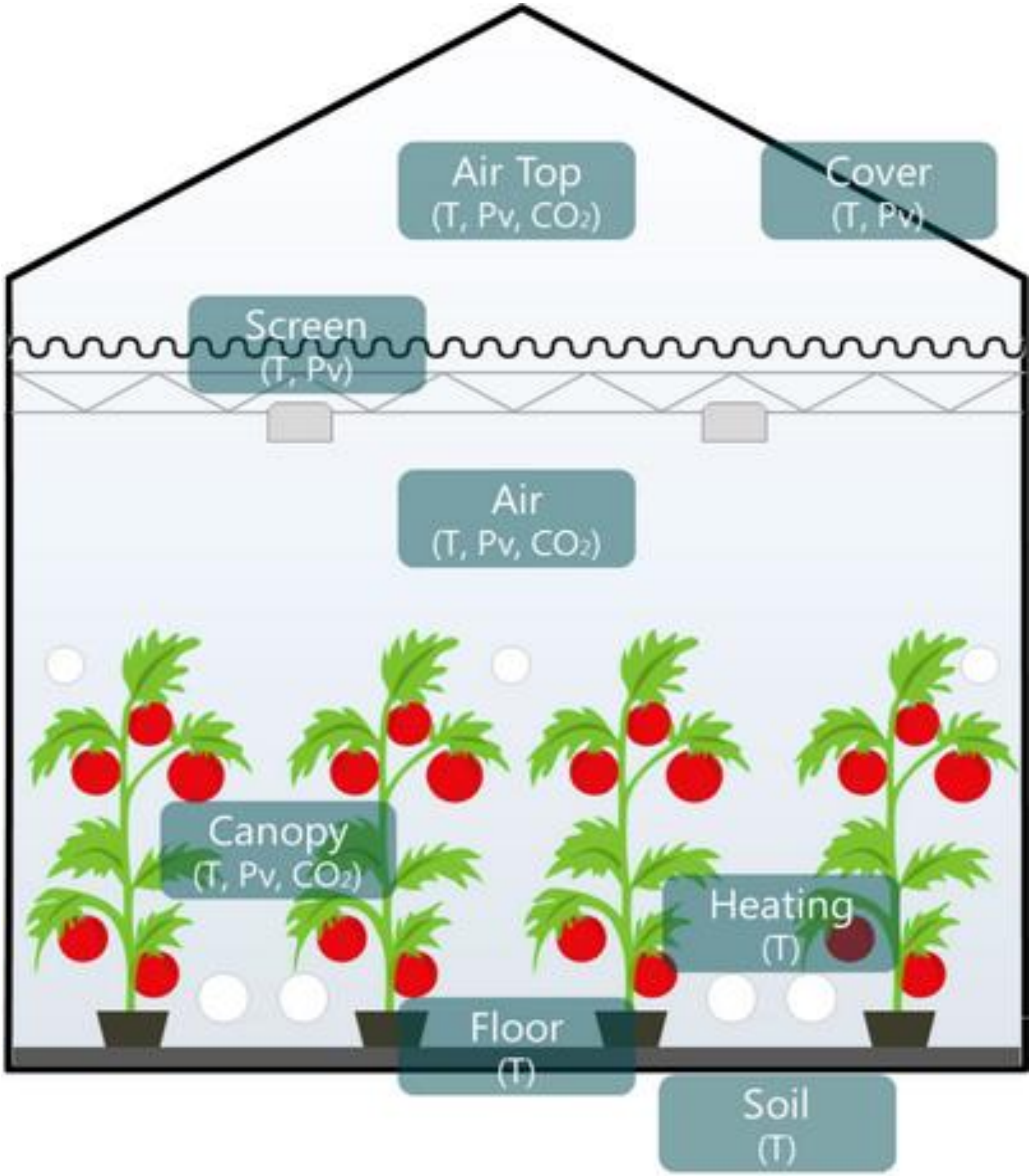


Figure 1: Graphical representation of the state variables (T: temperature, Pv: vapor pressure of water, CO2: CO2 concentration) of the greenhouse climate model.[11, 14]

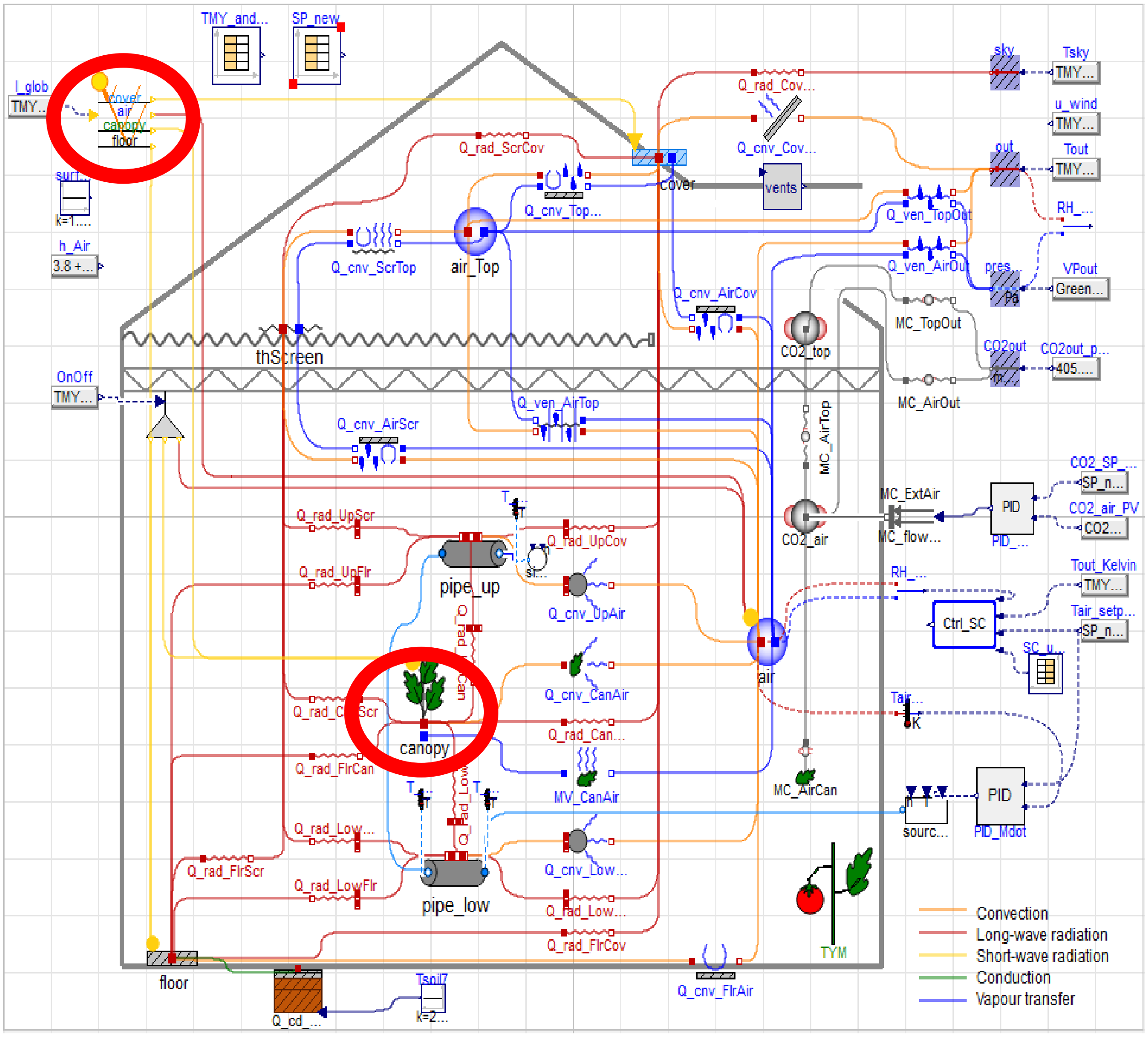


Figure 2:Graphical interface of the greenhouse model (Greenhouse1 in the Examples package)[14, 18]

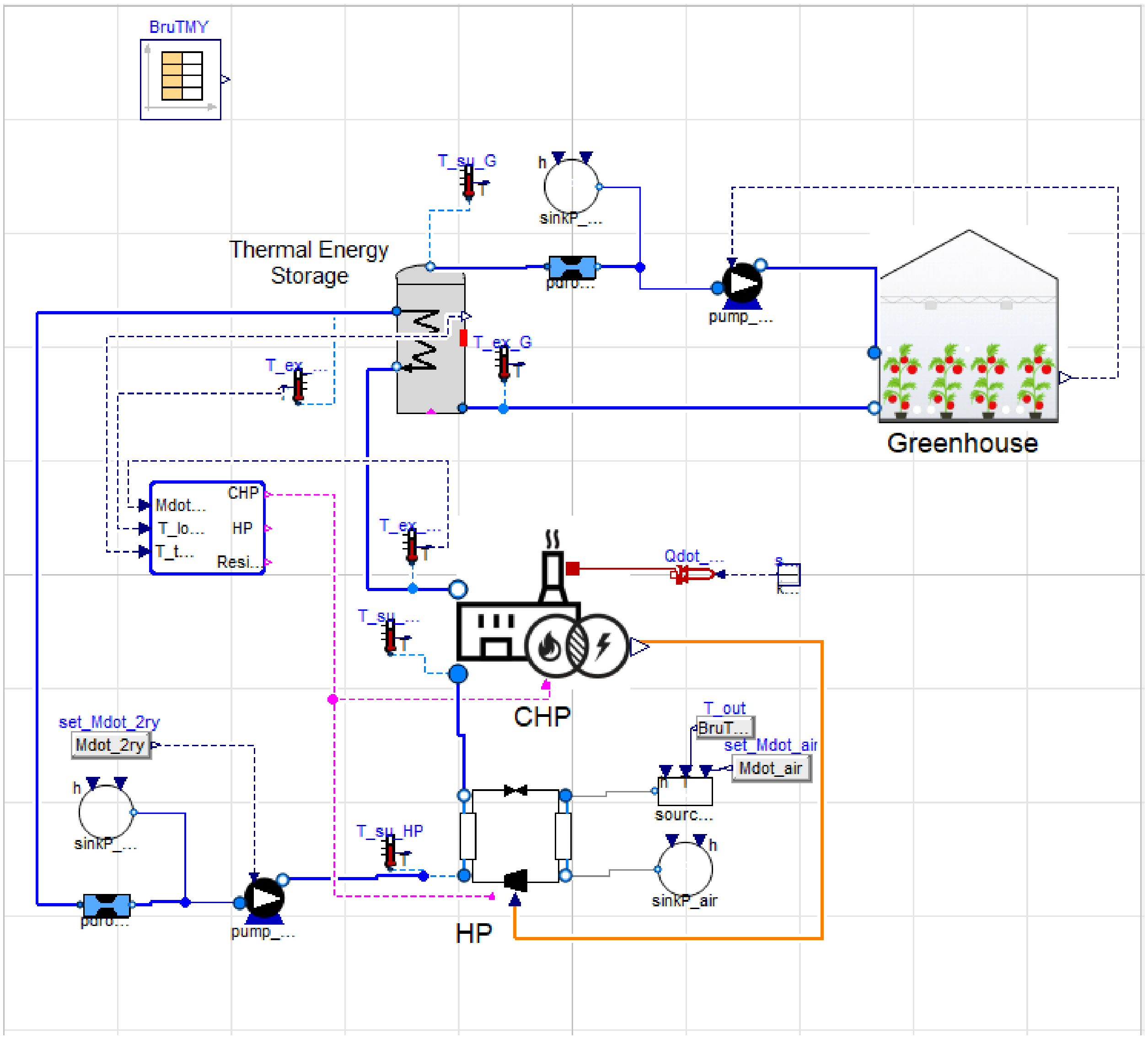


Figure 3:Graphical interface of the HVAC system combined with the greenhouse model (GlobalSystem_2 in Examples package)[14, 20]

Table 1:Greenhouse design parameters for the study[11]

Parameters	Value	Units
Mean greenhouse cover slope	25	°
Surface of the cover including sidewalls	18000	m^2
Surface of the greenhouse floor	14000	m^2
Height at which the screen is installed	3.8	m
Mean height of the greenhouse	4.2	m
Cover		
Density	$2.6 \cdot 10^3$	kg m^{-3}
Thickness	$4 \cdot 10^{-3}$	m
Thermal Screen		
Density	$0.2 \cdot 10^3$	kg m^{-3}
Thickness	$0.35 \cdot 10^{-3}$	m
Floor		
Density	2300	kg m^{-3}
Thickness	0.02	m
Ventilation properties		
Vertical dimension of a single vent opening	0.68	m
CO_2 enrichment		
Capacity of the external CO_2 source (Base Case)	0	g m^{-2} hr^{-1}
Capacity of the external CO_2 source (CO2+ Case)	270	g m^{-2} hr^{-1}
Supplementary lighting (HPS)		
Capacity of the lamps	100	W m^{-2}

Table 2: Parameters for heating system.[14]

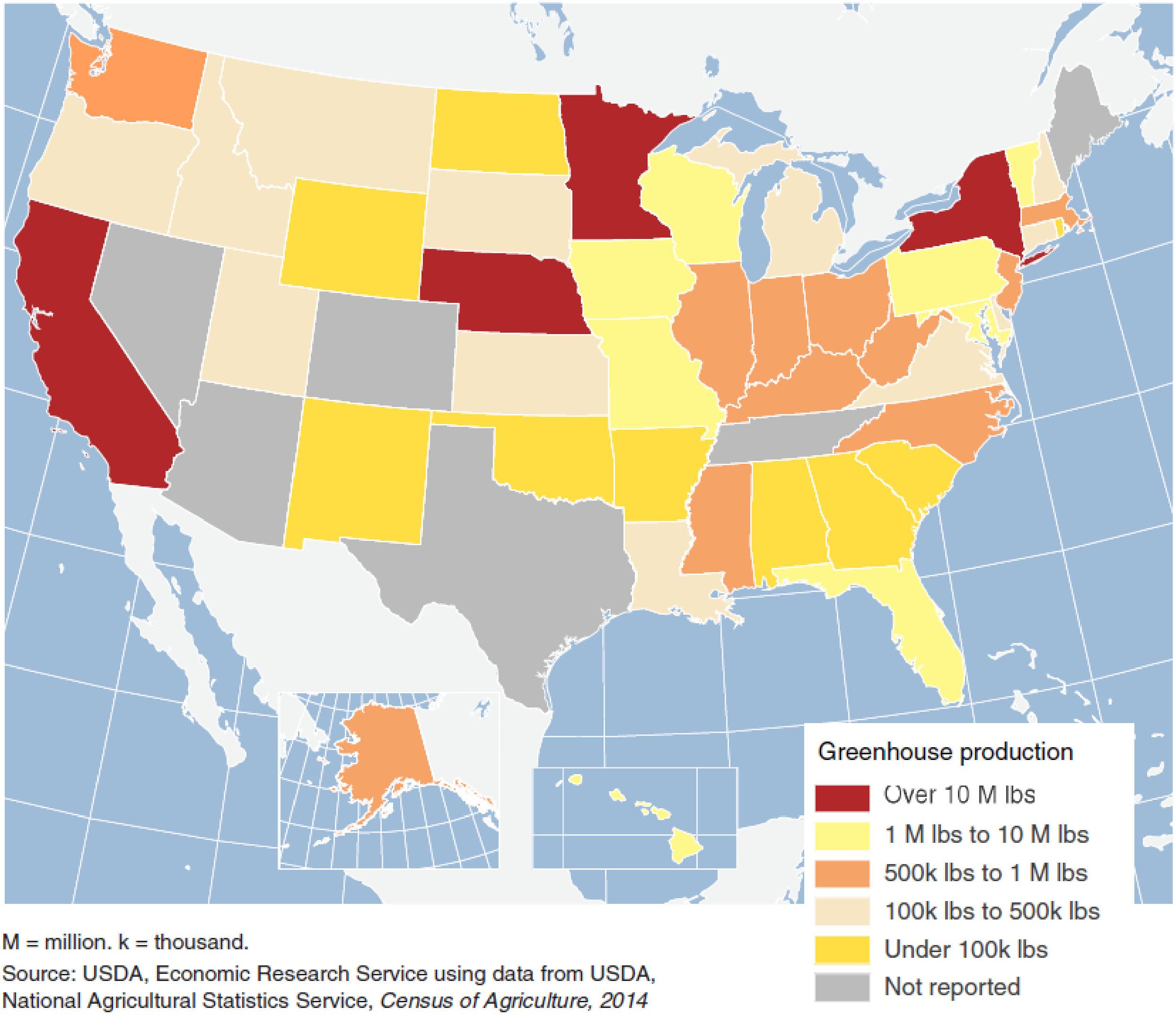
Parameters	Value	Units
Thermal Energy Storage	313	m^3
CHP capacity	884	kW_{th}
HP capacity	490	kW_{th}

2.2 Selection of greenhouse production locations

Ten locations were investigated for this study which was selected for the counties with the highest sales in 2017.[21] Figure 4 shows the areas in the country with the highest concentrations of greenhouses.[2] Table 3 lists the location and the models used to simulate greenhouse production.

Table 3:Crop Reporting Districts of focus and simulation used.

State	County	CRD	Time Zone	Latitude	Longitude	Model
Texas	Parker	TX30	-6	32.77	-97.82	Greenhouse 1
Pennsylvania	Lancaster	PEN42	-5	40.01	-76.26	Global System 2
Ohio	Highland	OH80	-5	39.21	-83.62	Global System 2
New York	Washington	NY60	-5	43.25	-73.46	Greenhouse 1
Missouri	Morgan	MIS50	-6	38.37	-92.9	Global System 2
Minnesota	Dakota	MN90	-6	44.69	-93.14	Greenhouse 1
Illinois	Moultrie	IL70	-6	39.61	-88.62	Global System 2
Colorado	Adams	CO68	-7	39.89	-104.18	Global System 2
California	Fresno	CA51	-8	36.77	-119.7	Greenhouse 1
Arizona	Maricopa	AZ80	-7	33.49	-112.46	Greenhouse 1



M = million. k = thousand.

Source: USDA, Economic Research Service using data from USDA, National Agricultural Statistics Service, *Census of Agriculture, 2014 Census of Horticultural Specialties.*

Figure 4: U.S. greenhouse tomato production, 2014[21]

2.3 Model and Input Data Management

The greenhouse model requires three input files to simulate crop growth: climate data, controller, and thermal screen setpoints. These files can be easily linked to the greenhouse model by clicking the CombiTimeTable icons TMY_and_control, SP_new, and SC_usable, respectively. Other variables can be altered in the graphical interface, such as the flow rate of CO_2 supplied from an external source, crop growth parameters, and ventilation system.

2.3.1 Production Location Climate Data

Climate data such as Temperature of the outside air (T_out), Relative Humidity of the outside air (RH_out), Pressure of the outside air (P_out), Global irradiation (I_glob), and Wind speed of outside air (u_wind) was retrieved from the National Solar Radiation Database (NSRDB) which compiles meteorological data and measurements for solar radiation.[22] The weather input file for each CRD contained the following variables: Time, T_out, RH_out, P_out, I_glob, u_wind, Sky Temperature (T_sky), Temperature setpoint of the greenhouse inside air (T_air_sp), Carbon dioxide setpoint (CO2_air_sp), ilu_sp. These files were compiled for 2017 to 2019 for each county in text (tab-delimited) files. T_sky was calculated by the equation[23]:

$$Tsky = 0.037536\mathrm{Tamb}^{1.5} + 0.32\mathrm{Tam}$$

CO2_air_sp is set to atmospheric conditions for the Base Case presented in Table 12 in **Appendix A** and increased for the CO_2 enrichment case. In the second scenario, the CO_2 enrichment scheme was kept from the sample greenhouse location[11, 18] and applied to the CRDs of interest, shown in Table 13 in **Appendix A**.

In the graphical interface, the parameter CO2out_ppm_to_mgm3 was modified for the specific year, and the average CO_2 concentration for the year was inputted from Table 12 in **Appendix A**. MC_ExtAir parameter was also modified using 0 $\frac{g}{m^2 hr}$ and 270 $\frac{g}{m^2 hr}$ for the base case and CO_2 enrichment, respectively.

Natural sunlight illumination setpoint (ilu_sp) is calculated from the “dusk_dawn.m” file from **Appendix C**.

2.3.2 Sunrise and Sunset Time

Each CRD is customized with location-specific sunset and sunrise time, allowing for optimal real-life simulation results. The sunrise sunset model is computed from a MATLAB resource, illustrating the ability to estimate sunrise and sunset time for a given longitude and latitude.[24] The sunrise and sunset time are used to calibrate the setpoints for natural illumination (ilu_sp) and the deployment of the aluminized thermal screen (SC_usable), which is done by "dusk_dawn.m" and "thermalscrn.m" in **Appendix C**.

2.3.3 Controller Setpoint

The setpoint files are adjusted for each year with the optimal temperature setpoint (T_sp), and the CO_2 supplied (CO2_sp) for growth enrichment. The temperature is set at a constant 20°C which is an optimal temperature for tomato plant growth[25], and the CO_2 scheme follows the externally supplied profile from Table 12 from **Appendix A**.[26] This elevated CO_2 allows plants to increase yield by enhancing photosynthesis during the day.[14]

2.4 Output Data Collection

The critical data points for electrical and thermal energy, natural gas used, water, and carbon footprint with respect to the floor area (m^2) were collected and are listed in Table 4, which specifies the variable name and where it can be found in the results after the simulated period.

Table 4:Collected variables from Greenhouse Model.

Variable	**Model Equivalent**	**Units**
Total Electrical	E_el_tot_kWhm2	kWhr/m^2
Buy Electrical	E_el_buy_kWhm2	kWhr/m^2
Sell Electrical	E_el_sell_kWhm2	kWhr/m^2
Electrical generated from CHP unit	E_el_CHP_kWhm2	kWhr/m^2
Electrical used by HP	E_el_HP_kWhm2	kWhr/m^2
Total Thermal	E_th_tot_kWhm2	kWhr/m^2

Table 4 (Cont.): Collected variables from Greenhouse Model.

Variable	**Model Equivalent**	**Units**
Thermal generated from CHP unit	E_th_CHP_kWhm2	kWhr/m^2
Thermal sustained by HP	E_th_HP_kWhm2	kWhr/m^2
Combined heat	E_th_total_kWhm2	kWhr/m^2
Natural Gas used	E_gas_CHP_kWhm2	kWhr/m^2
Transpiration	MV_CanAir.E_kgsm2	L/m^2
Dry Matter Harvested	DM_Har	kg/m^2
CO_2 externally supplied	MC_ExtAir.MCflow	kg/m^2

The greenhouse library does not have a component that explicitly states water consumption, so an alternative is analyzing the transpiration of the tomato plants. To calculate the water footprint of the greenhouse, transpiration data was extracted calculated by *"Waterconverter.m"* in **Appendix C** using the numerical analytical technique of the Trapezoid Rule. Similarly, CO_2 exhausted from the system did not have a component identifying cumulative emission. Analysis of the externally supplied CO_2 was utilized for the carbon footprint of the location and was calculated by *"Co2_enrichment.m"* in **Appendix C** using the Trapezoid Rule as well. Other values of interest were calculated and are listed in Table 5.

Table 5:Calculated Variables.

Variable	**Unit**
Net Electrical Energy	kWhr/m^2
PWU	L/kg
Fresh Weight	kg/m^2

Net Purchased Electricity (Net Electrical Energy) was calculated by:

$$Net\ Electrical\ Energy = Buy\ Electrical\ Energy - Sell\ Electrical\ Energy$$

Product Water Use (PWU) was calculated by[19]:

$$PWU = \frac{Water\ Used}{Fresh\ Weight}$$

Fresh Weight was calculated using the *"DM_harvested.m"* script in **Appendix C**. Reports were created using Power B.I. for Electrical and Thermal Energy, Tomato Yield, and CHP and H.P. performance. Figs. 8-13,15-26 illustrate the for both cases from 2017-2019. The data can be reviewed in Tables 14-16 of **Appendix E**.

Chapter 3: Results

In this study, each of the 10 CRDs was simulated for both the base case and CO2+ scenario for 2017-2019. There were three simulations each for the base case and CO_2 enrichment, respectively, for a location. The solving time for a 1-year simulation using the Greenhouse_1 model took an average of 55 mins, and the Global Systems_2 model took 139 mins using a 3.3 GHz i5 processor. The output data from the model were accessed using tabular or graphical representation where appropriate in Dymola for further processing. The simulated results generated from Global Systems 2 builds upon the data calculated from Greenhouse 1, so all data points of interest were accessible in each case. Table 6 shows a sample data set in tabular form for the total electrical energy used, dry matter harvested, and the thermal energy needed. The table also lists where the values were extracted and their location in the model. The generated results are consistent at 1800s (30 min) intervals with the frequency of data points in the location climate data file. Figs. 5-7 illustrate sample output data generated from the example models of the greenhouse library. The simulated dry matter harvested from the tomato plants during the 1-year growth period (Figure 5) was used to calculate the fresh weight tomato yield. Canopy transpiration and externally supplied CO_2 for CO_2 enrichment during the simulated period (Figures 6 and 7) were used to estimate the water and carbon footprint, respectively, for each location.

.

Table 6: Sample tabular results of dry matter harvested, electrical and thermal energy.

	0	1800	3600	5400	7200	9000	10800	12600	14400	16200	18000	19800	21600	23400
E_el_tot [kW·h]	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E_el_tot_kWhm2 [kW·h/m^2]	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DM_Har [mg/m^2]	0	-0	-0	-0	0	-0	-0	-0	-0	-0	-0	-0	-0	-0
E_th_tot [kW·h]	0	1325.4563	2003.2695	2428.7349	2828.0425	3249.0525	3812.4368	4592.3657	5178.811	5580.294	6344.574	6796.063	7434.871	8054.7627
E_th_tot_kWhm2 [kW·h/m^2]	0	0.09467545	0.14309068	0.17348105	0.20200303	0.23207517	0.27231693	0.32802615	0.36991507	0.39859244	0.45318386	0.48543307	0.53106225	0.5753402

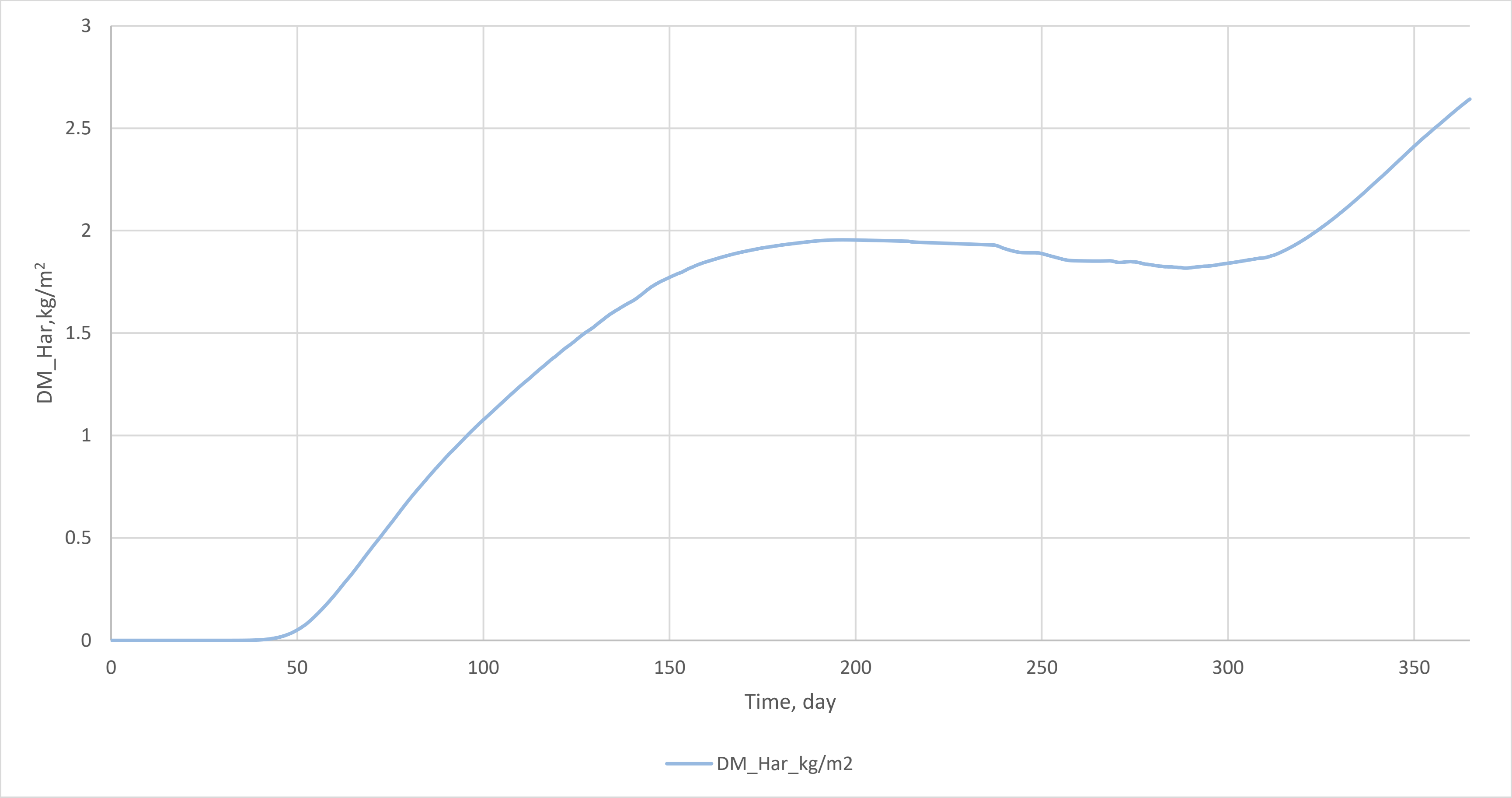


Figure 5:Sample plot of Harvested Dry Matter.

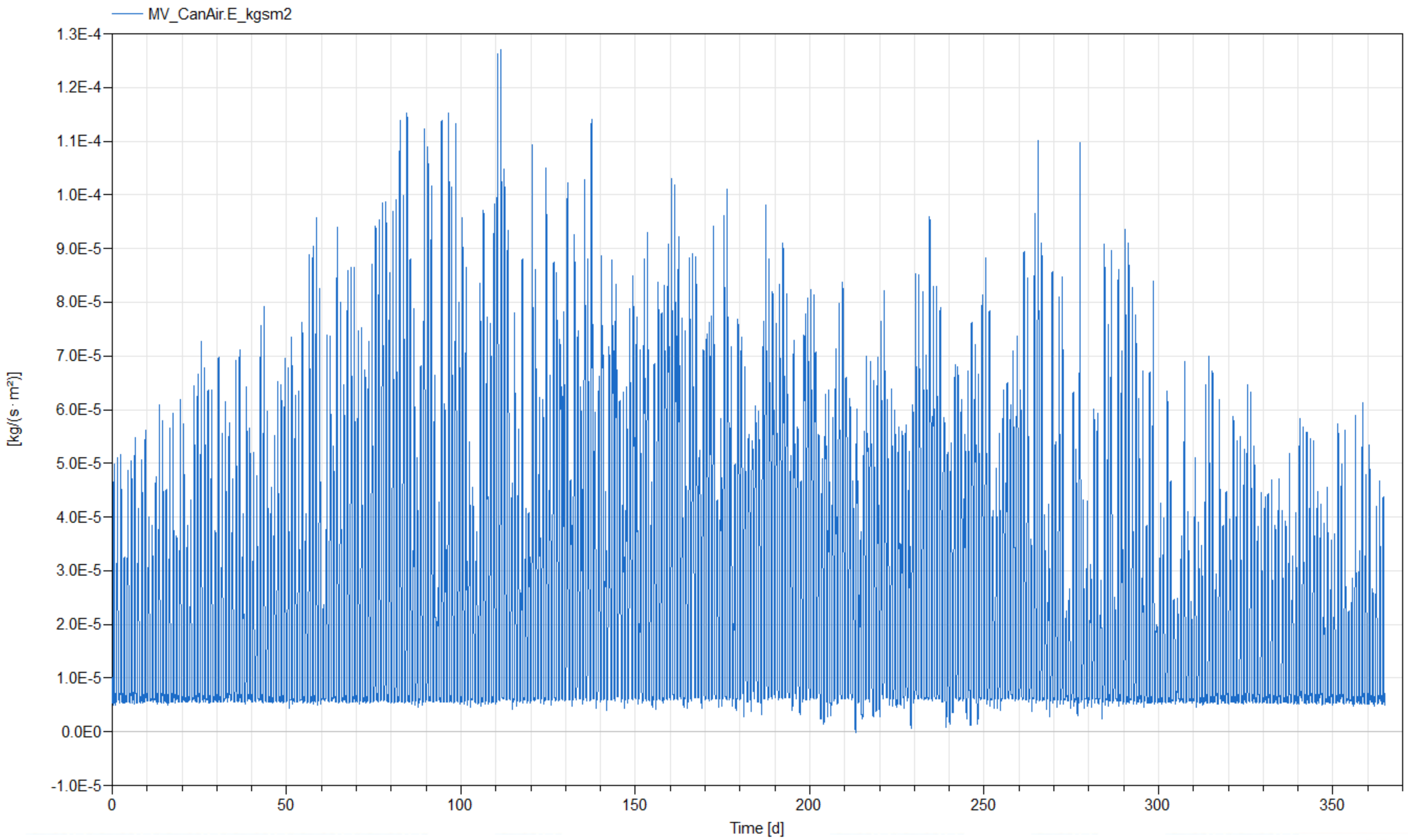


Figure 6: Sample plot of Canopy transpiration.

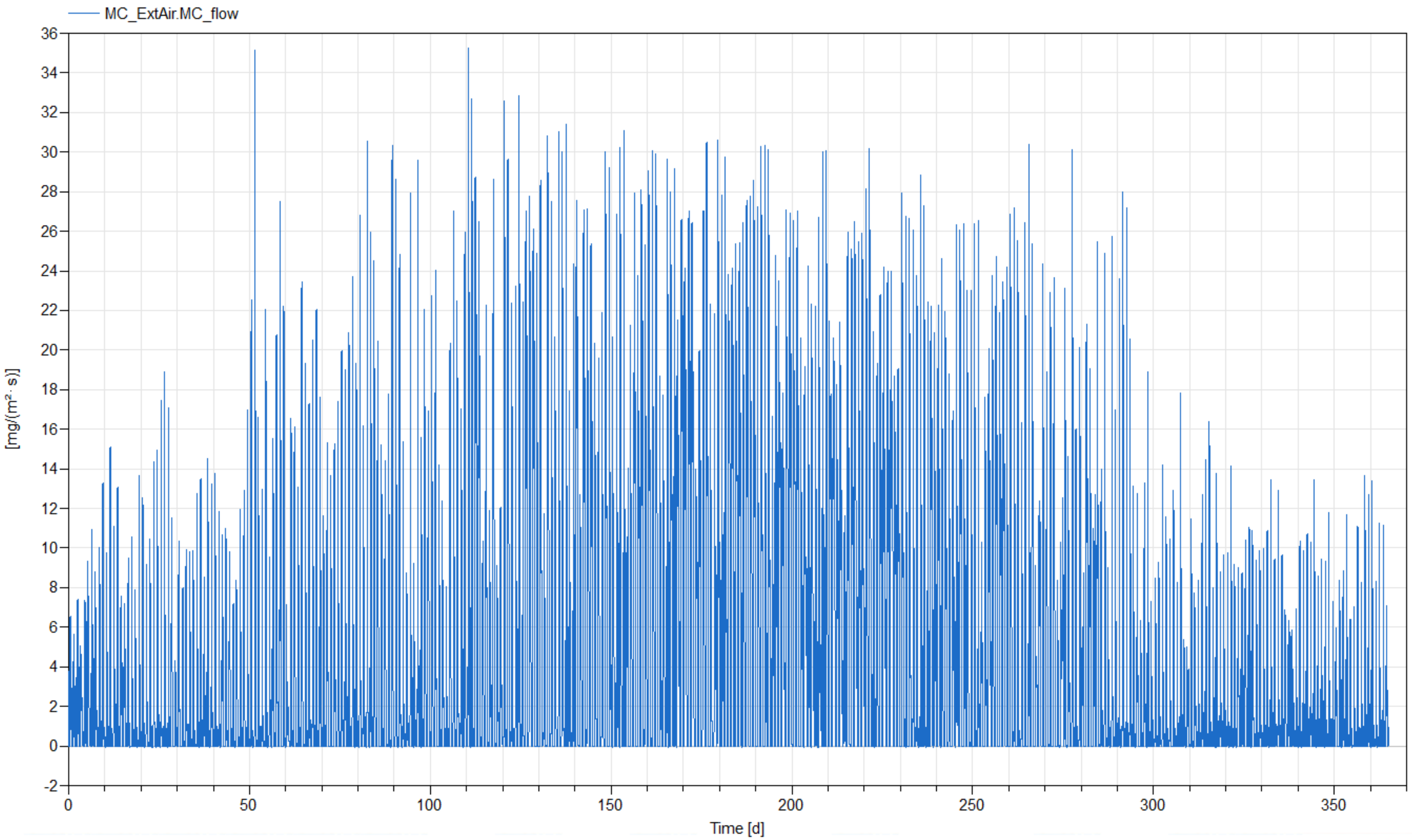
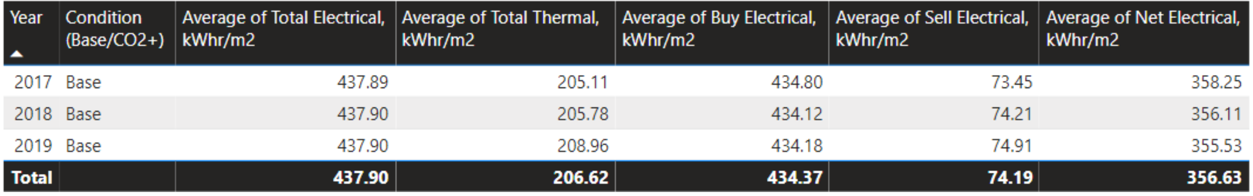


Figure 7: Sample plot of CO_2 enrichment from an external source.

3.1 Electrical and Thermal Energy

3.1.1 Base Case scenario

Table 7:Average Electrical and Thermal Energy for Base Case scenario.

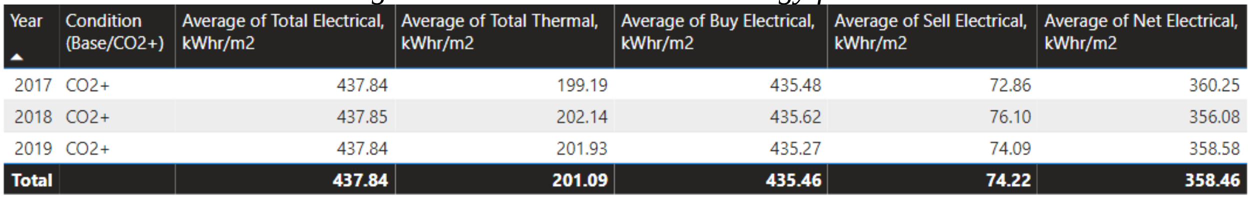


Year ▲	Condition (Base/CO2+)	Average of Total Electrical, kWhr/m2	Average of Total Thermal, kWhr/m2	Average of Buy Electrical, kWhr/m2	Average of Sell Electrical, kWhr/m2	Average of Net Electrical, kWhr/m2
2017	Base	437.89	205.11	434.80	73.45	358.25
2018	Base	437.90	205.78	434.12	74.21	356.11
2019	Base	437.90	208.96	434.18	74.91	355.53
Total		**437.90**	**206.62**	**434.37**	**74.19**	**356.63**

The simulated data for all 10 CRDs were in line with the averages listed in Table 7 for electricity used (Electrical Energy). The Net Electrical Energy, and Sell Electrical Energy were limited to 5 CRDS as the information necessary for those calculations can only be obtained from the Global Systems _2 model. NY60 and MN90 were switched from Global Systems_2 to Greenhouse_1 for simulation because the model failed shortly after initiation. The cause of the failure was that the average temperature during winter ranged from -10°C to -3°C, which registers as extreme weather for the model and triggered errors in solving the start values for the calculations. This error was also noted for MIS50 in 2018, as the average winter temperature was 1°C. Hence the data for 2018 lacks the necessary variables for Net Electrical Energy. Buy Electrical Energy is reduced as part of the electricity needed was produced by the CHP unit for the 5 CRDs using Global Systems_2. Most of the generated electricity is sold back to the grid. Thermal energy was consistent among the 10 CRDs with the averages listed in Table 8. MN90 had the highest thermal requirement with 305.58 kWhr/m^2, and AZ80 had the lowest at 160.1m1 kWhr/m^2.

3.1.2 CO_2+ scenario

Table 8: Average Electrical and Thermal Energy for CO2+ scenario.



Year	Condition (Base/CO2+)	Average of Total Electrical, kWhr/m2	Average of Total Thermal, kWhr/m2	Average of Buy Electrical, kWhr/m2	Average of Sell Electrical, kWhr/m2	Average of Net Electrical, kWhr/m2
2017	CO2+	437.84	199.19	435.48	72.86	360.25
2018	CO2+	437.85	202.14	435.62	76.10	356.08
2019	CO2+	437.84	201.93	435.27	74.09	358.58
Total		**437.84**	**201.09**	**435.46**	**74.22**	**358.46**

The data for all 10 CRDs were in line with the averages listed in Table 7 for Electrical Energy used. As previously stated, Net Electrical Energy, Buy Electrical Energy, Sell Electrical Energy, NY60, MN90, and MIS50 still apply for this scenario as the only change to the simulation was the conditions for CO_2 enrichment. Thermal energy usage is consistent among the 10 CRDs with the averages listed in Table 8. MN90 had the highest thermal requirement with 295.24 kWhr/m^2, and AZ80 had the lowest at 155.19 kWhr/m^2.

3.1.3 Electrical and Thermal Energy Reports

To better illustrate the observed behavior of the CRDs, reports were generated using Power B.I. for Electrical and Thermal Energy for each CRD. The average Buy and Total Electrical Energy and the Total Thermal Energy for the specific year are listed on the upper left. Charts for Buy Electrical Energy, Total Electrical Energy, Net Electrical Energy, and Total Thermal Energy were created to see the performance across each location. The lower right corner has customizable filters to narrow the report by year, state, and scenario condition. Figs. 8,10,12 illustrate the results for the Base Case scenario and Figs. 9,11,13 illustrate the results for CO_2+ scenarios. Tables 14-15 in **Appendix E** list all the data used for calculations in both scenarios.

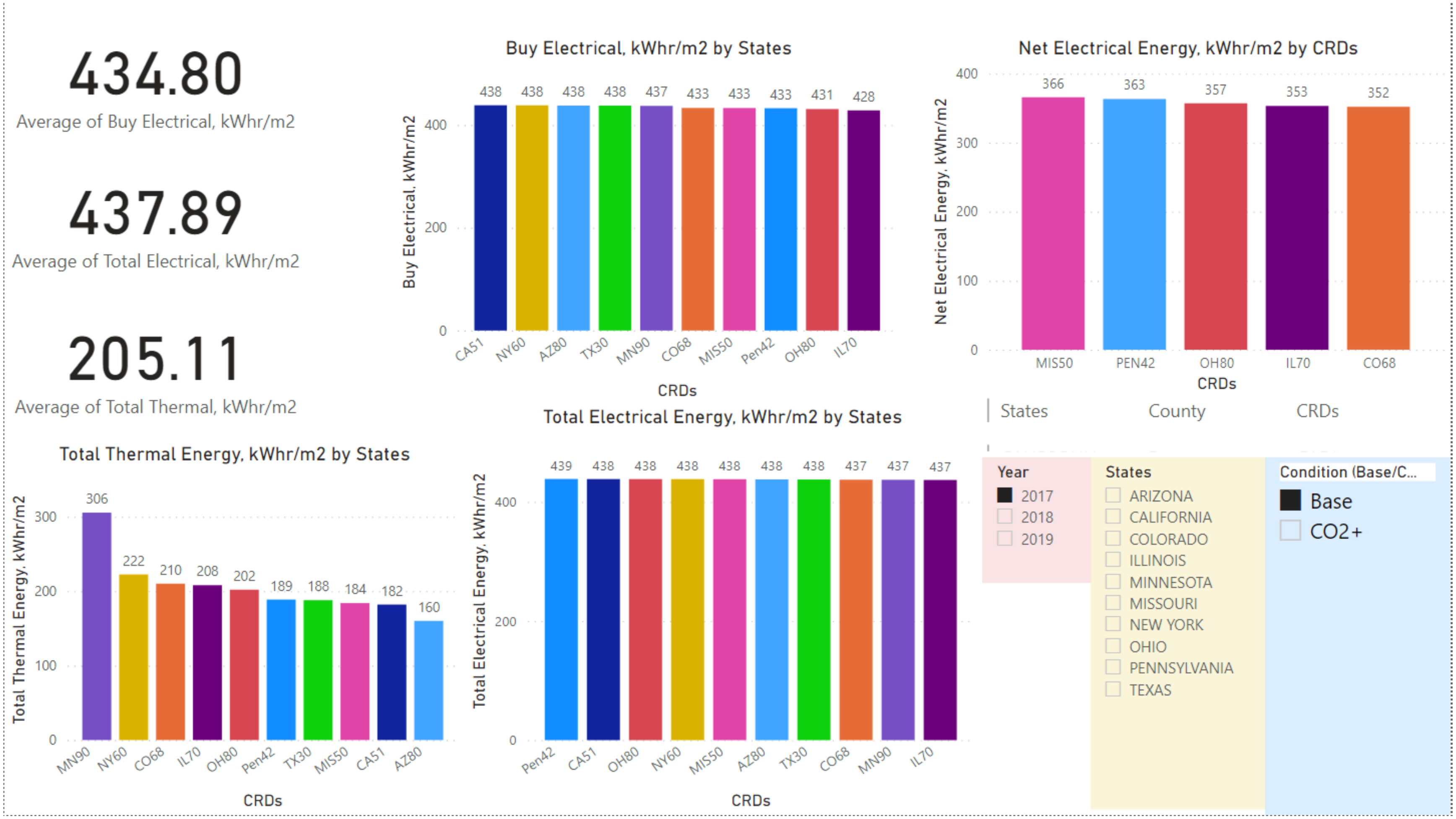
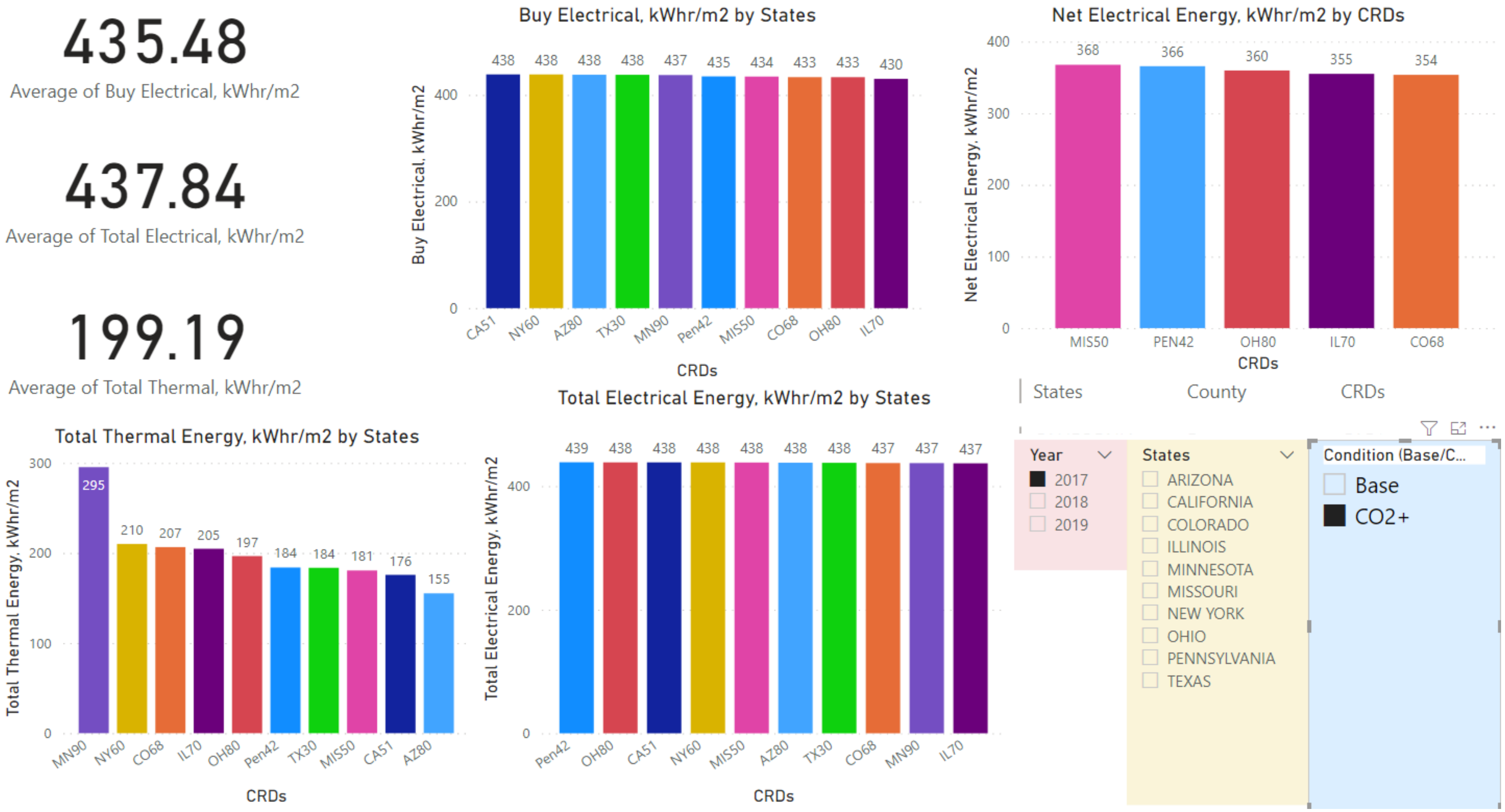


Figure 8:Report of Electrical and Thermal Energy for 2017 for Base Case.



435.48
Average of Buy Electrical, kWhr/m2

437.84
Average of Total Electrical, kWhr/m2

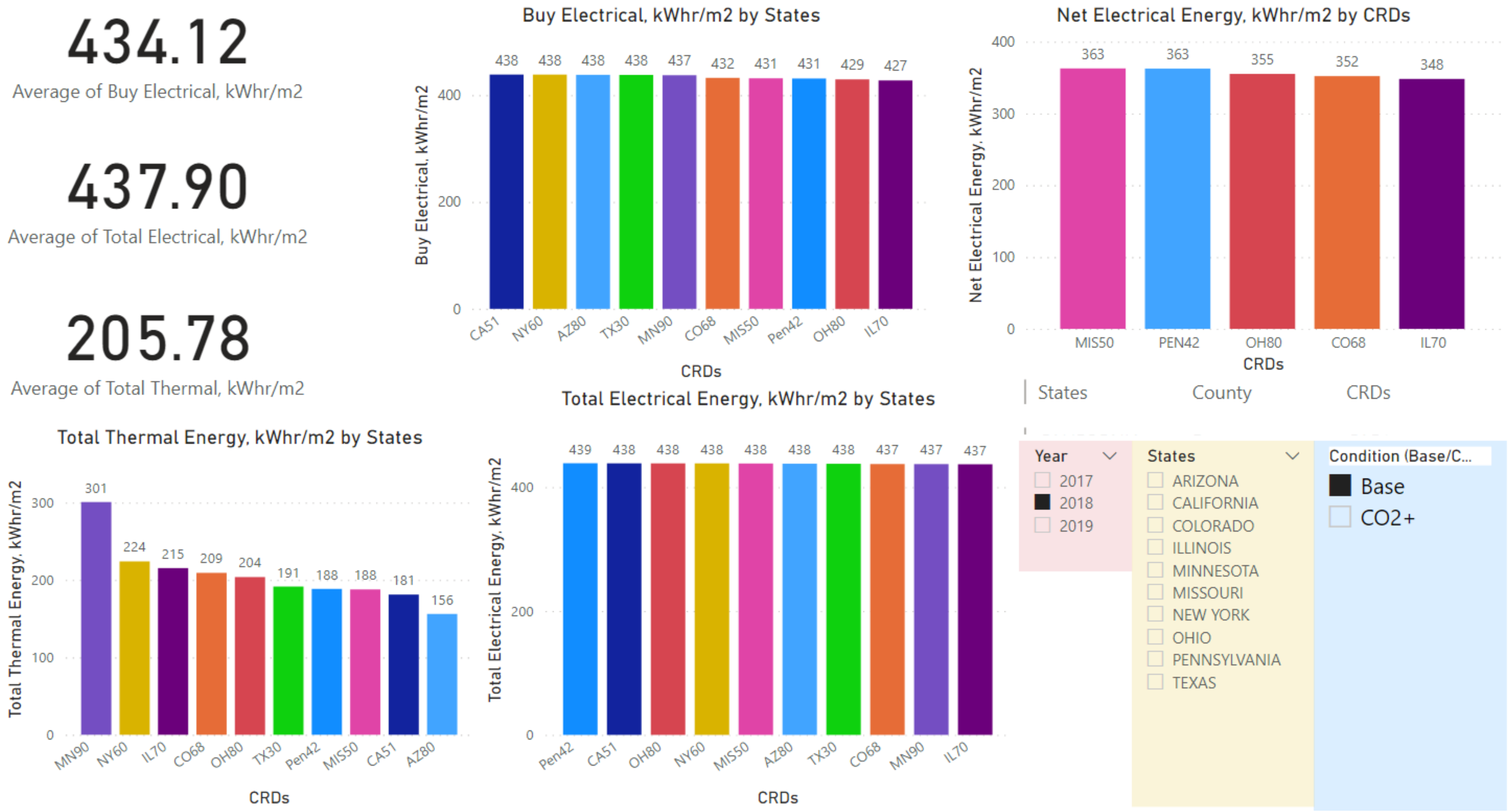
199.19
Average of Total Thermal, kWhr/m2

Year
☒ 2017
☐ 2018
☐ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
☐ Base
☒ CO2+

Figure 9: Report of Electrical and Thermal Energy for 2017 with CO_2 Enrichment.



434.12
Average of Buy Electrical, kWhr/m2

437.90
Average of Total Electrical, kWhr/m2

205.78
Average of Total Thermal, kWhr/m2

Year
☐ 2017
☒ 2018
☐ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
☒ Base
☐ CO2+

Figure 10:Report of Electrical and Thermal Energy for 2018 for Base Case.

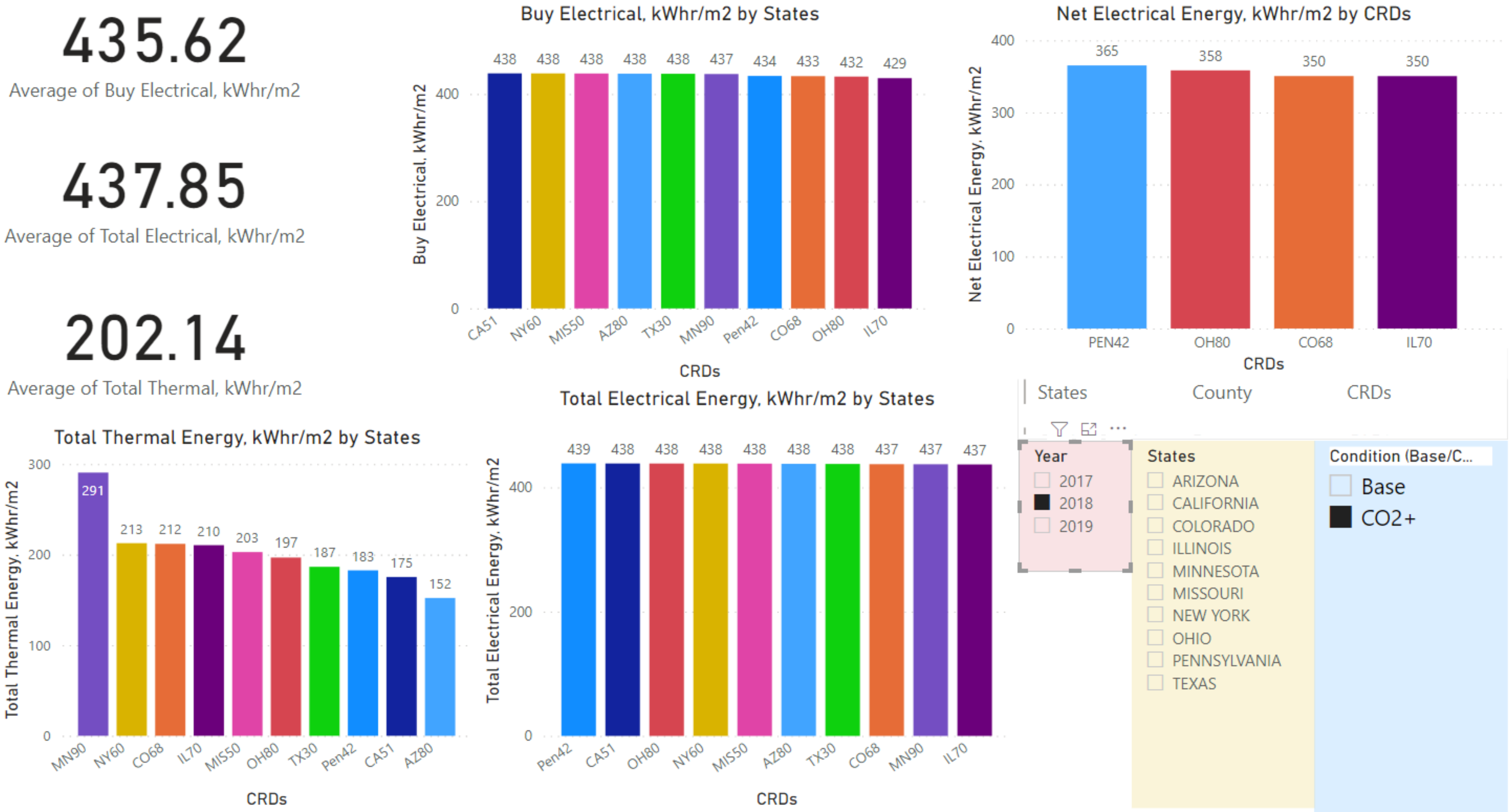
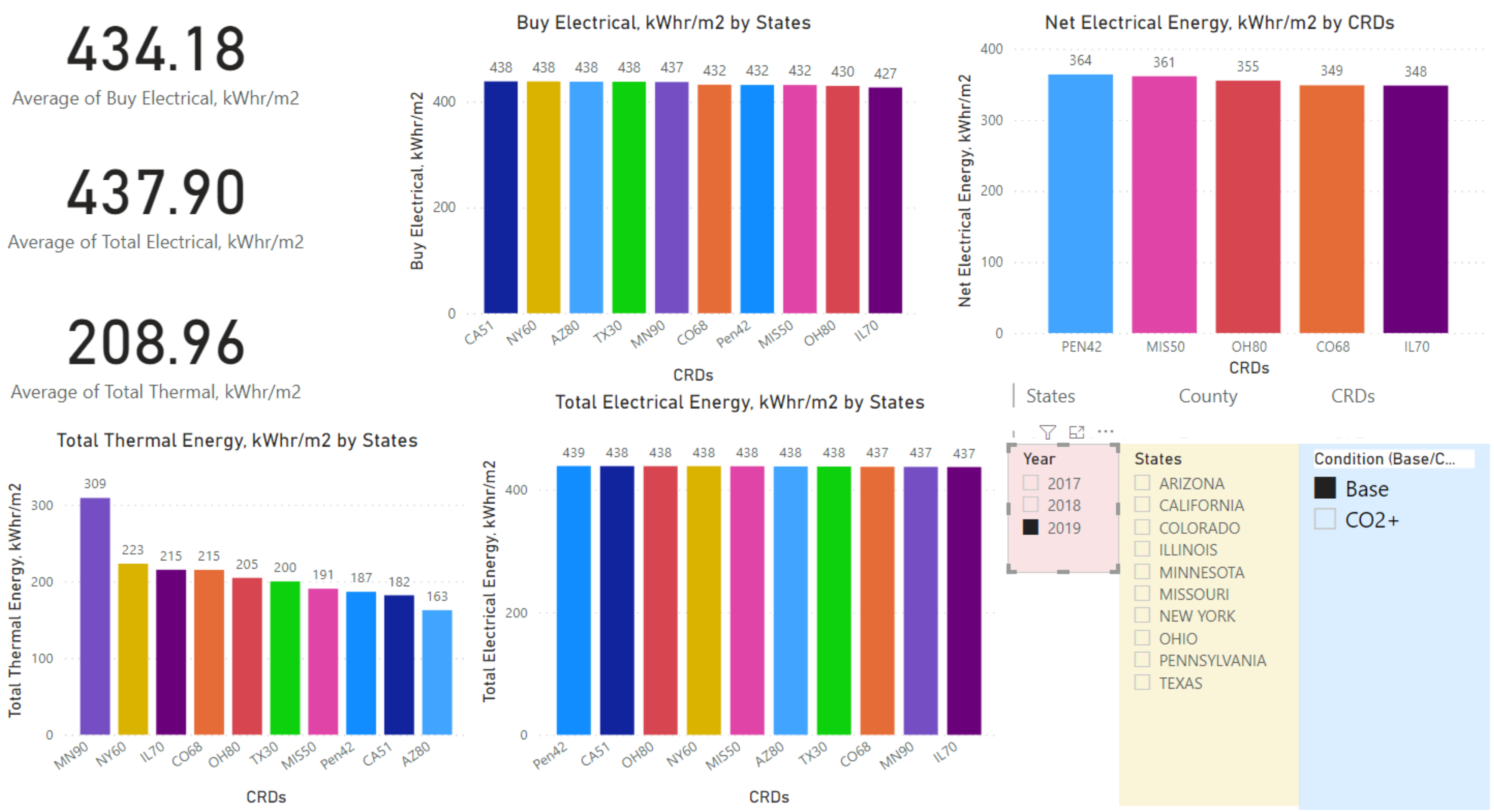


Figure 11: Report of Electrical and Thermal Energy for 2018 with CO_2 Enrichment.



434.18
Average of Buy Electrical, kWhr/m2

437.90
Average of Total Electrical, kWhr/m2

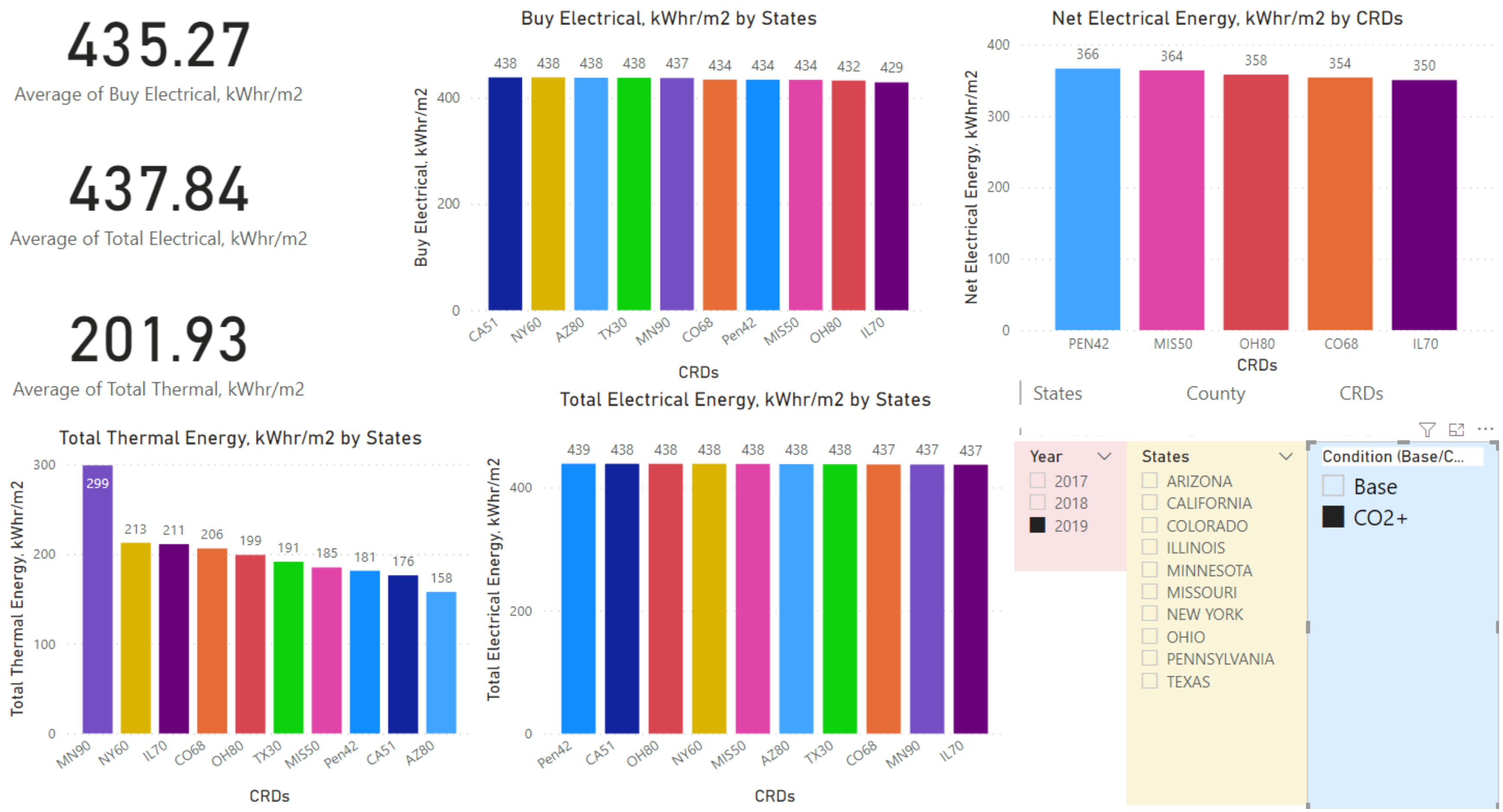
208.96
Average of Total Thermal, kWhr/m2

Year
☐ 2017
☐ 2018
■ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
■ Base
☐ CO2+

Figure 12:Report of Electrical and Thermal Energy for 2019 for Base Case.



435.27

Average of Buy Electrical, kWhr/m2

437.84

Average of Total Electrical, kWhr/m2

201.93

Average of Total Thermal, kWhr/m2

Year

☐ 2017
☐ 2018
■ 2019

States

☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...

☐ Base
■ CO2+

Figure 13:Report of Electrical and Thermal Energy for 2019 with CO₂ enrichment.

3.2 Tomato Production

Tomato production from greenhouses was expected to generate 50-80 kg/m2/yr yield.[19] The observed PWU for a semi-closed greenhouse can range as low as 4 L/kg, but usually varies from 12.5-20 L/kg. Greenhouse operations in locations in hot arid regions cooled with pad-and-fan systems, and fogging can range from over 60 to 90 L/kg.[19]

3.2.1 Koppen Climate Characterization

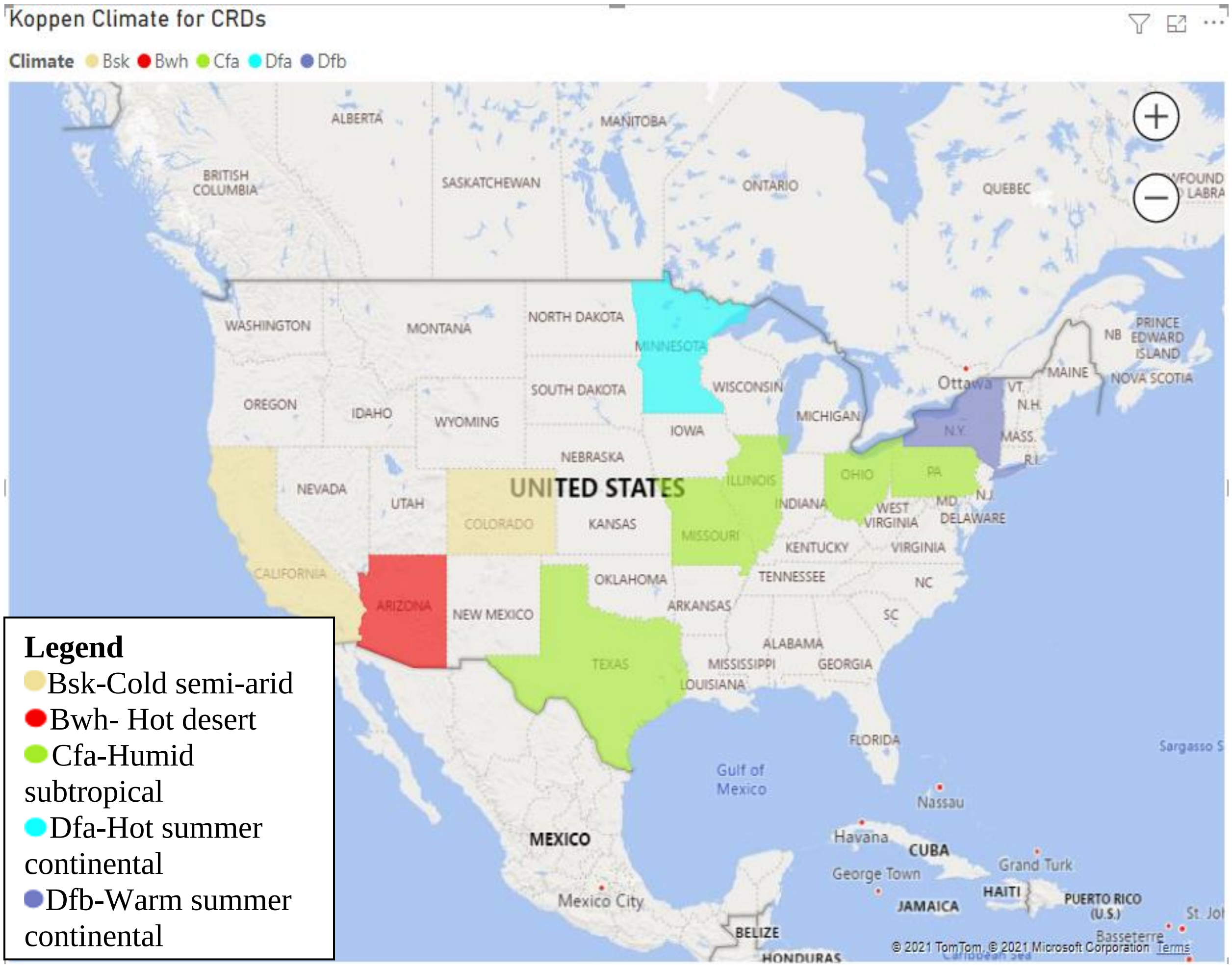


Figure 14: Map showing Koppen Climate for CRDs.[27]

Koppen Climate Characterization was used to connect CRDs within similar climates to see their performance as a group, and the potential impact the external environment may have on greenhouses. Observations made from the results may inform various stakeholders on the need for additional crop controls such as pad-and-fan cooling systems to reduce high internal temperatures in the greenhouse.

Table 9:Koppen Climate for each CRD.[27]

States	CRDs	County	Koppen Climate	Model
CALIFORNIA	CA51	Fresno	Cold semi-arid, Bsk	Greenhouse 1
TEXAS	TX30	Parker	Humid subtropical, Cfa	Greenhouse 1
OHIO	OH80	Highland	Humid subtropical, Cfa	Global system 2
NEW YORK	NY60	Washington	Warm summer continental, Dfb	Greenhouse 1
MINNESOTA	MN90	Dakota	Hot summer continental, Dfa	Greenhouse 1
ARIZONA	AZ80	Maricopa	Hot desert, Bwh	Greenhouse 1
ILLINOIS	IL70	Moultrie	Humid subtropical, Cfa	Global system 2
MISSOURI	MIS50	Morgan	Humid subtropical, Cfa	Global system 2
COLORADO	CO68	Adams	Cold semi-arid, Bsk	Global system 2
PENNSYLVANIA	PEN42	Lancaster	Humid subtropical, Cfa	Global system 2

3.2.1.1 Humid subtropical, Cfa

Most of the CRDs in the Humid subtropical (Cfa) climate performed within the expected range for tomatoes in Greenhouses. OH80, IL70, and PEN 42 were the best performing, and MIS50 and TX30 were average to slightly below the expected range for tomatoes grown in greenhouses. The fresh weight of tomatoes grown ranged from 72 kg/m^2 to 42 kg/m^2 for the base case and 79 kg/m^2 to 45 kg/m^2 with CO_2 enrichment. The tomato plants' total water consumed varied from 1227.19 L/m^2 to 889.63 L/m^2 for the base case, and 1075.21 L/m^2 to 807.48 L/m^2 with CO_2+.

The PWU ranged from 13.99 L/kg to 26.86 L/kg for the base case, and 11.36 L/kg to 22.58 L/kg with CO_2+. The best performing locations had the highest yield and lowest water footprint than other CRDs in this climate zone. CO_2 enrichment positively increased tomato product yield and reduced the water footprint of greenhouse production.

3.2.1.2 Cold semi-arid, BSk

CO68 had better performance than CA51 concerning tomato yield, and water footprint. CO68 had excellent production performance compared with yields reaching up to 85 kg/m^2 and lows at 74 kg/m^{2}CA51 had average greenhouse tomato performance reaching as high as 54.16 kg/m^2 and low as 46.82 kg/m^2. In 2017 and 2018, the fresh weight yield for CA51 with CO_2 enrichment decreased by 2.65 kg/m^2 and 2.61 kg/m^2, respectively. In 2017 for CO68 with CO_2 enrichment, there was also decreased yield by 2.41 kg/m^2. Further investigation outside the scope of this study is needed to understand this behavior. In all three years, the water footprint decreased as a result of CO_2 enrichment.

3.2.1.3 Hot desert, BWh

AZ80 had the lowest performance compared to the other CRDs. The Base Case had a higher yield with values in the mid-30s than with CO_2 enrichment, which was in the upper 20s. This observation requires further study to see what triggered this response in the model. Water usage decreased due to CO2 enrichment in all three years, but was the highest of all locations. As a result of low yield, and high water usage, the PWU increased.

3.2.1.4 Hot summer continental, Dfa

MN90 had the best performance of all the CRDS, with yield reaching as high as 96.05 kg/m^2 with CO_2 enrichment and low as 71.98 kg/m^2 for the base case. In all three years, water footprint decreased as a result of CO_2 enrichment.

3.2.1.5 Warm summer continental, Dfb

NY60 had high yields reaching as high as 71.02 kg/m^2 with CO_2 enrichment and low as 57.31 kg/m^2 for the base case. In all three years, water footprint decreased as a result of CO_2 enrichment.

3.2.2 Tomato Production Report

To better illustrate tomato production across each location, reports were generated using Power B.I. for each CRD. A smaller version of Figure 14 is attached on the upper left corner, identifying the Koppen Climate Zone. In the bottom center, a treemap chart helps show the size of the fresh weight yield with respect to each CRD. Charts for Fresh Weight Yield, Water footprint, and PWU are listed at the top. A chart for the Carbon footprint is located on the bottom left, but is only visible in the CO_2+ scenario. The lower right corner has customizable filters to narrow the report by year, state, and scenario condition. Figs. 15-20 illustrate the observed behavior, and the data collected from simulations can be reviewed in Table 16 in **Appendix E**.

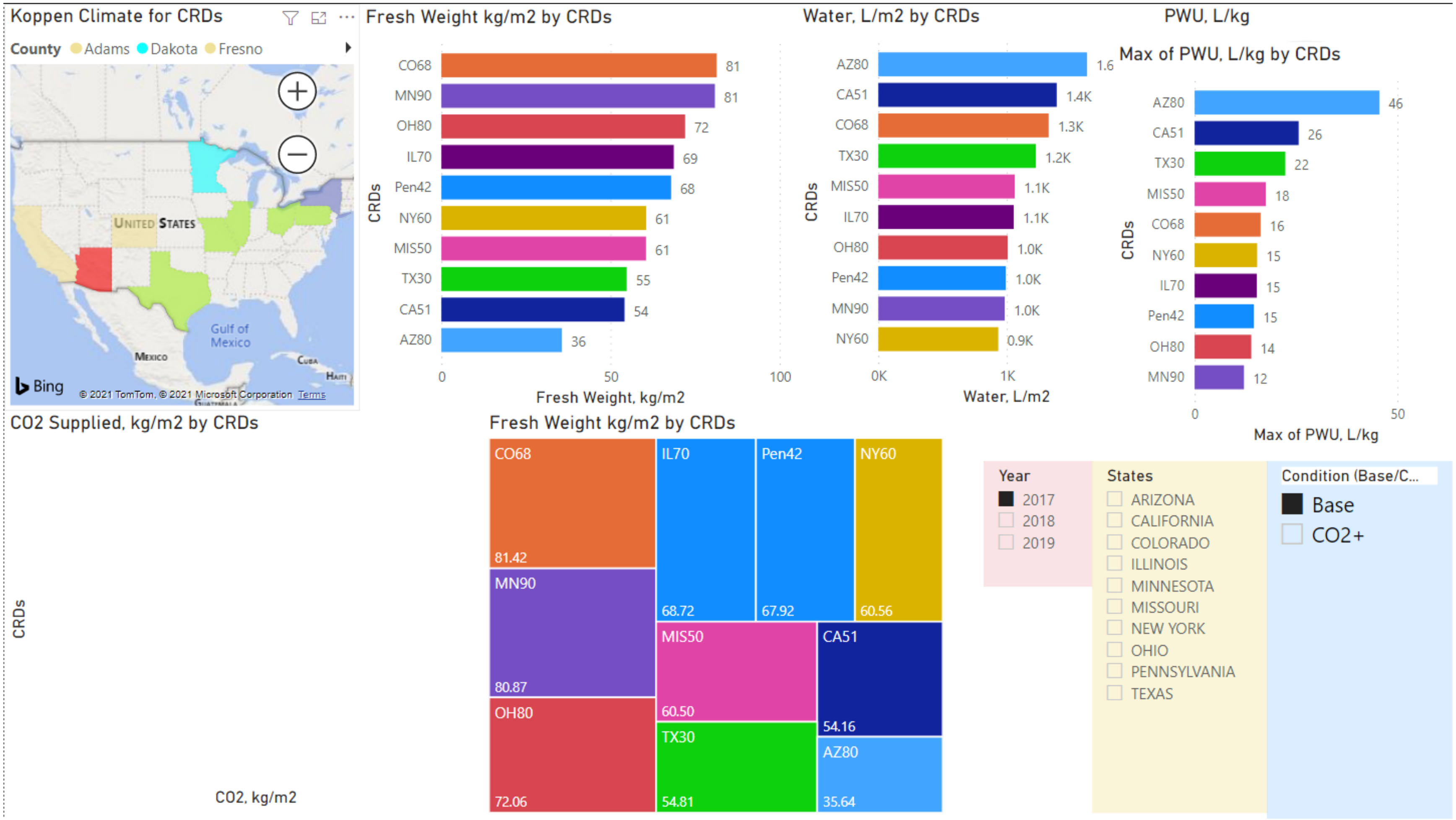
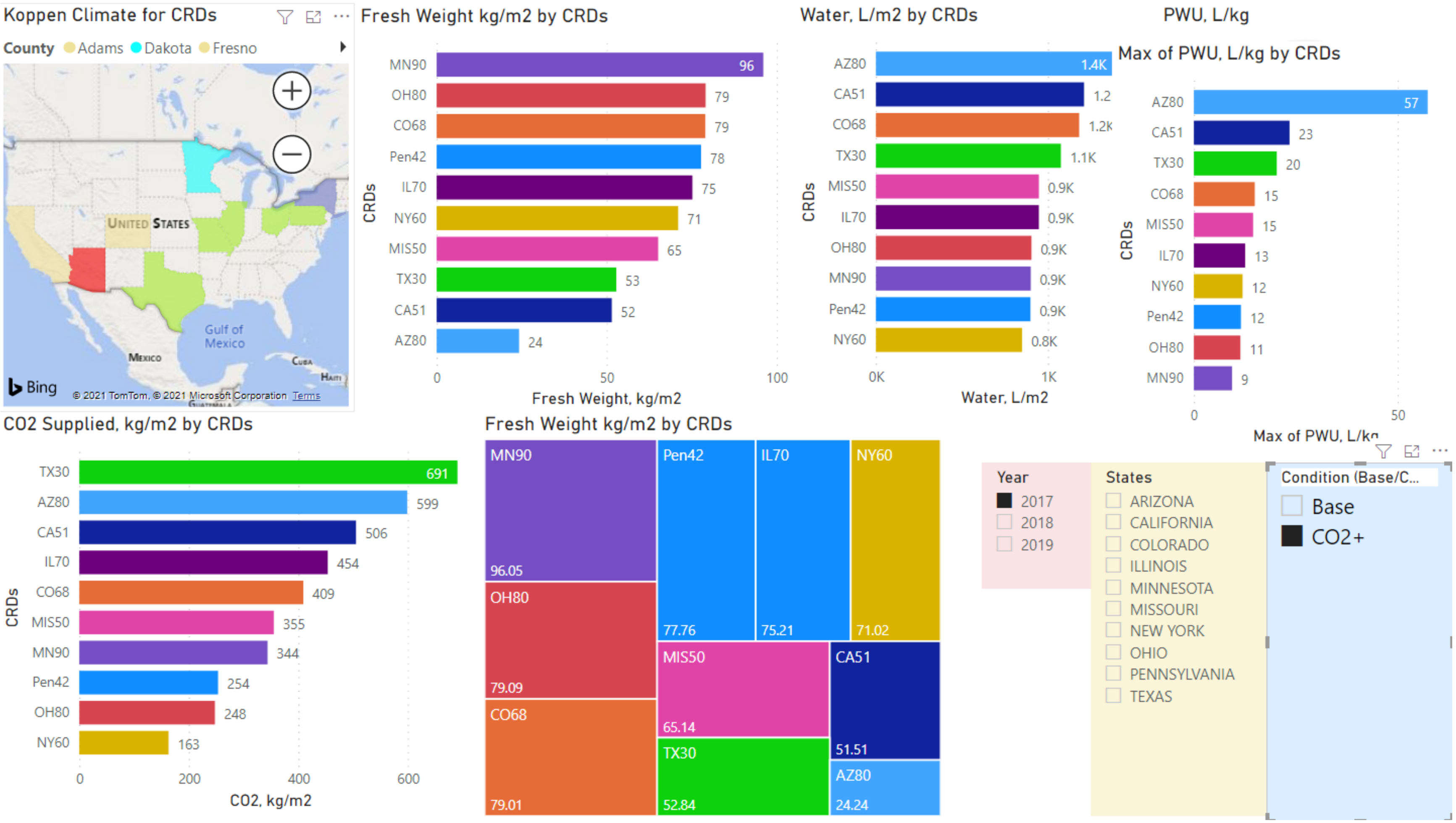


Figure 15:Report of Fresh Weight Tomato Yield for 2017 for Base Case.

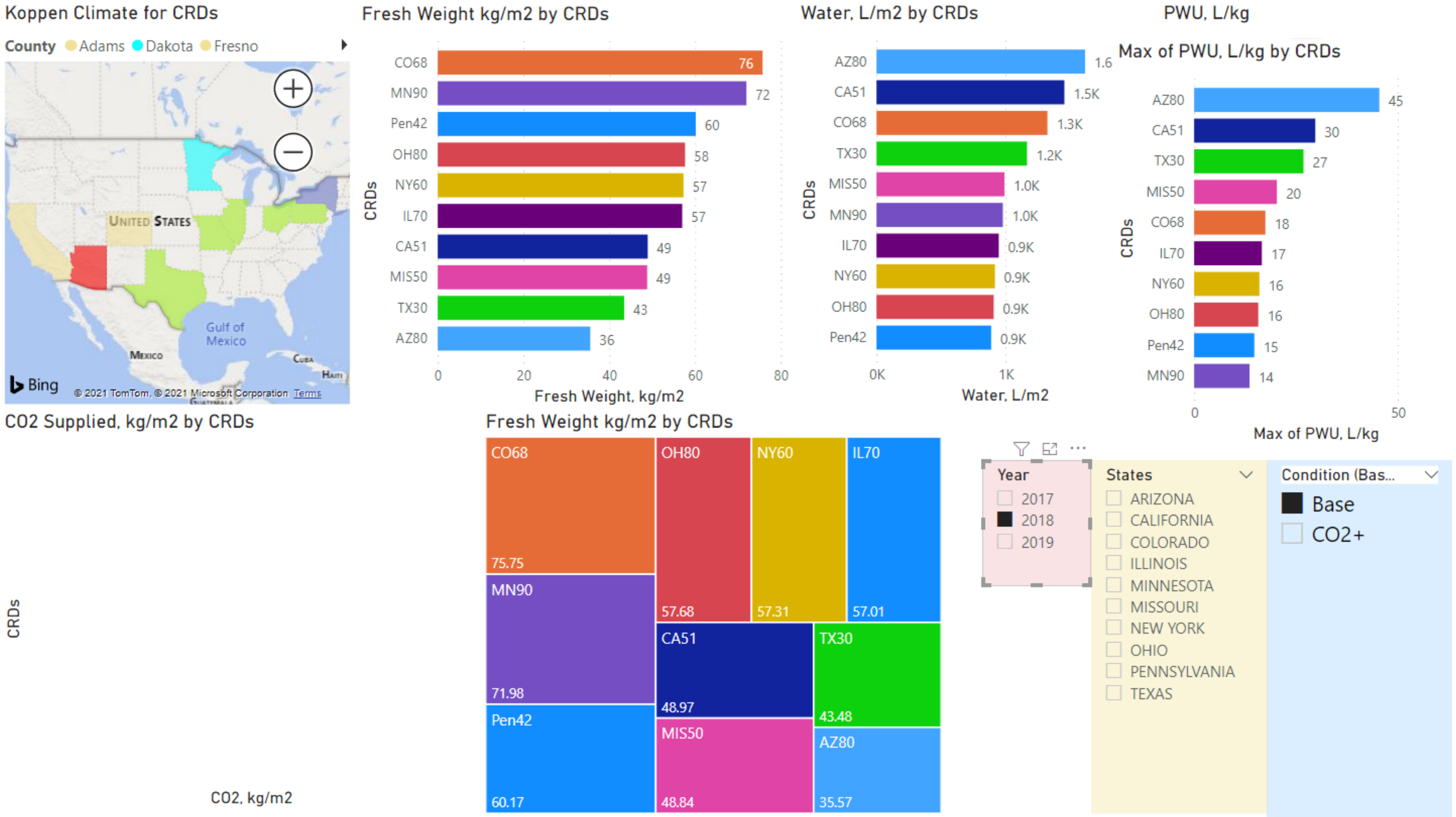


Year
■ 2017
☐ 2018
☐ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
☐ Base
■ CO2+

Figure 16: : Report on Fresh Weight Tomato Yield for 2017 with CO_2 Enrichment.

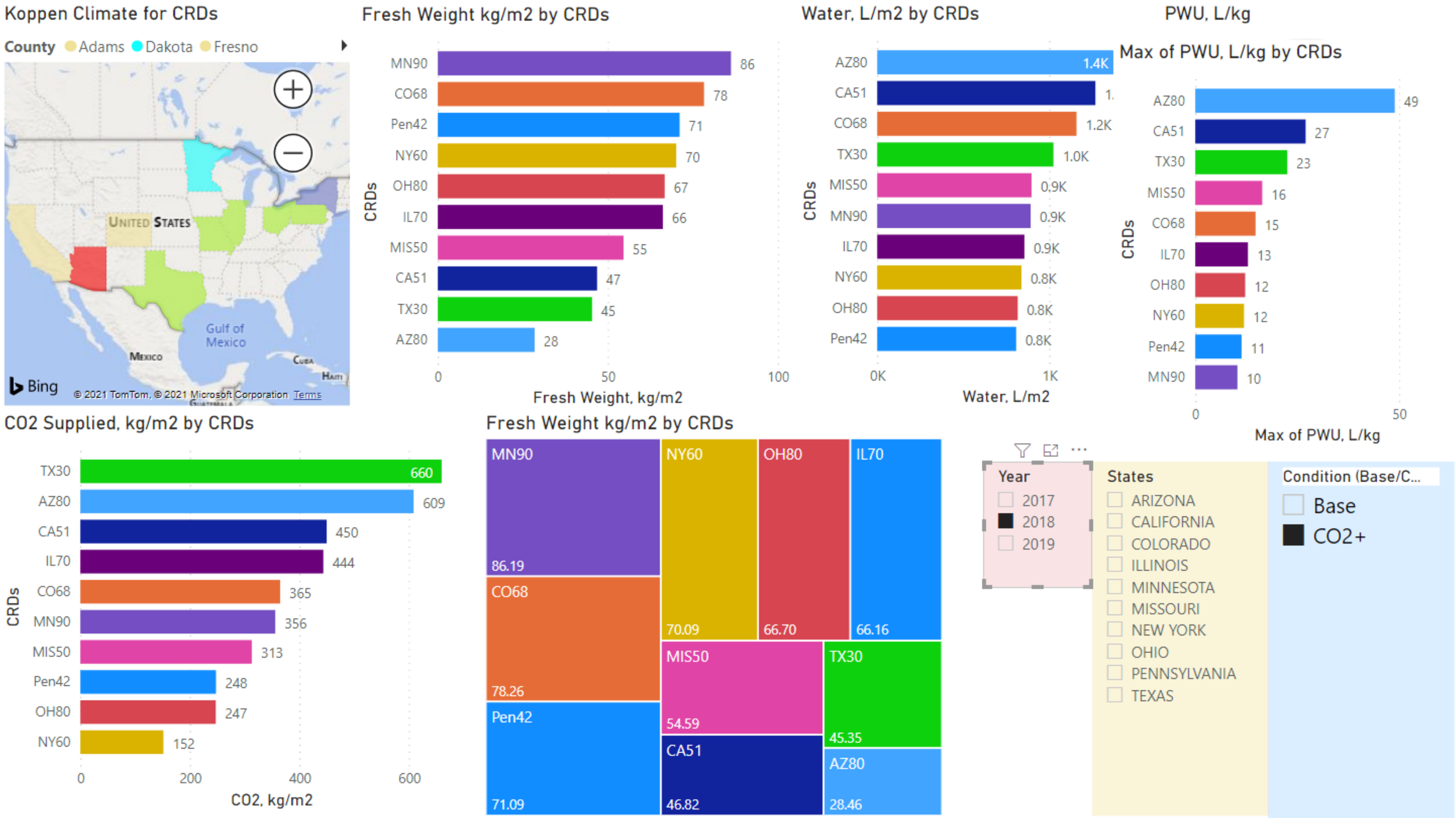


Year
☐ 2017
☒ 2018
☐ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Bas...
☒ Base
☐ CO2+

Figure 17:Report on Fresh Weight Tomato Yield for 2018 for Base Case.

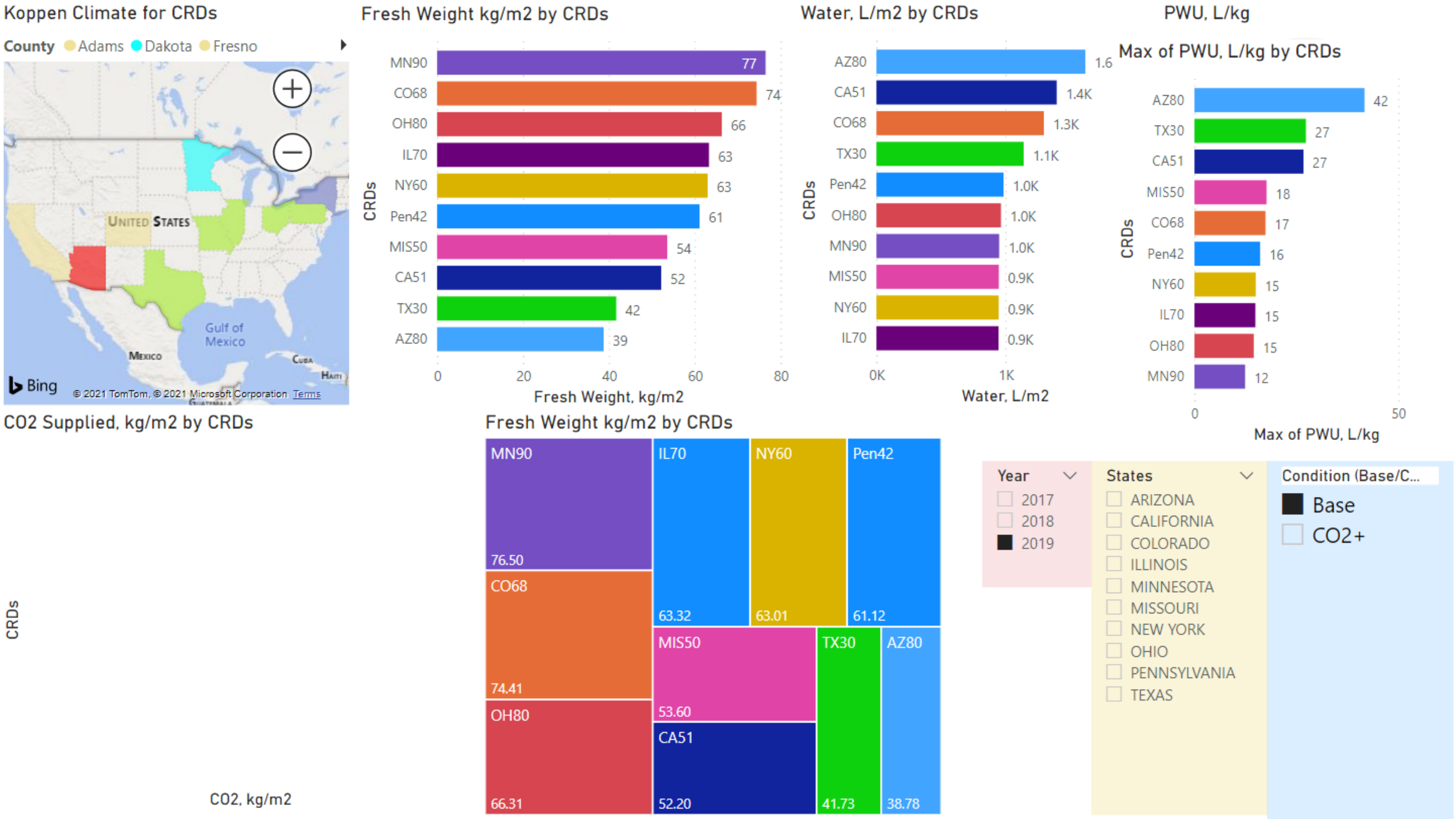


Year
☐ 2017
☒ 2018
☐ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
☐ Base
☒ CO2+

Figure 18: Report on Fresh Weight Tomato Yield for 2018 with CO_2 Enrichment.



Year
☐ 2017
☐ 2018
☒ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
☒ Base
☐ CO2+

Figure 19:Report on Fresh Weight Tomato Yield for 2019 for Base Case.

Year
☐ 2017
☐ 2018
■ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

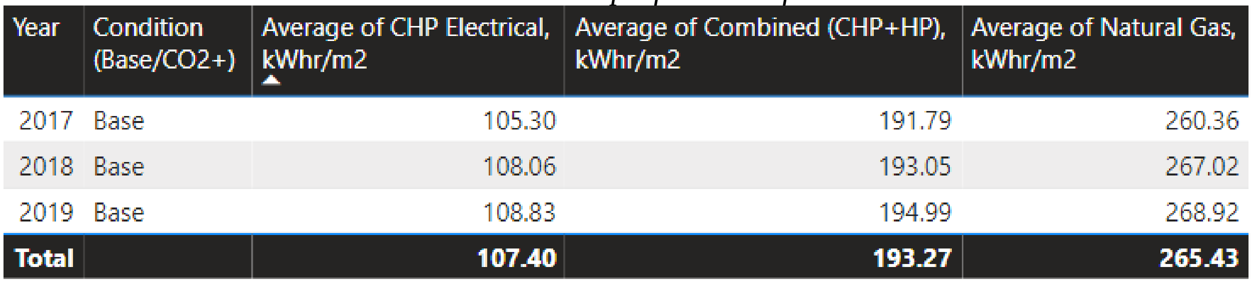
Condition (Base/C...
☐ Base
■ CO2+

Figure 20:Report on Fresh Weight Tomato Yield for 2019 with CO$_2$ Enrichment.

3.3 CHP and H.P. performance

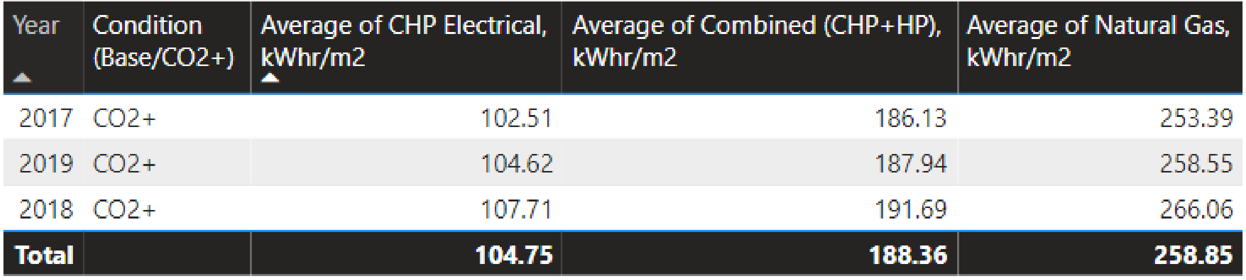
Global System_2 model uses the CHP and H.P. to provide a portion of the electricity and most of the heat needed for the greenhouse. Overall, CO68, IL70, and OH80 had a higher electrical and heating demand than PEN42 and MIS50. This resulted in more natural gas being supplied to the CHP. CO_2 enrichment increased tomato yield performance which reduced the electrical and heating demand of the greenhouses, thus reducing the amount of natural gas consumed. This is reflected in Tables 10-11, which show the average electrical energy produced (CHP Electrical), thermal energy generated (Combined CHP+HP), and natural gas burned for the respective years and scenarios.

Table 10: CHP and H.P. performance for Base Case.



Year	Condition (Base/CO2+)	Average of CHP Electrical, kWhr/m2	Average of Combined (CHP+HP), kWhr/m2	Average of Natural Gas, kWhr/m2
2017	Base	105.30	191.79	260.36
2018	Base	108.06	193.05	267.02
2019	Base	108.83	194.99	268.92
Total		**107.40**	**193.27**	**265.43**

Table 11:CHP and H.P. performance for CO_2+.

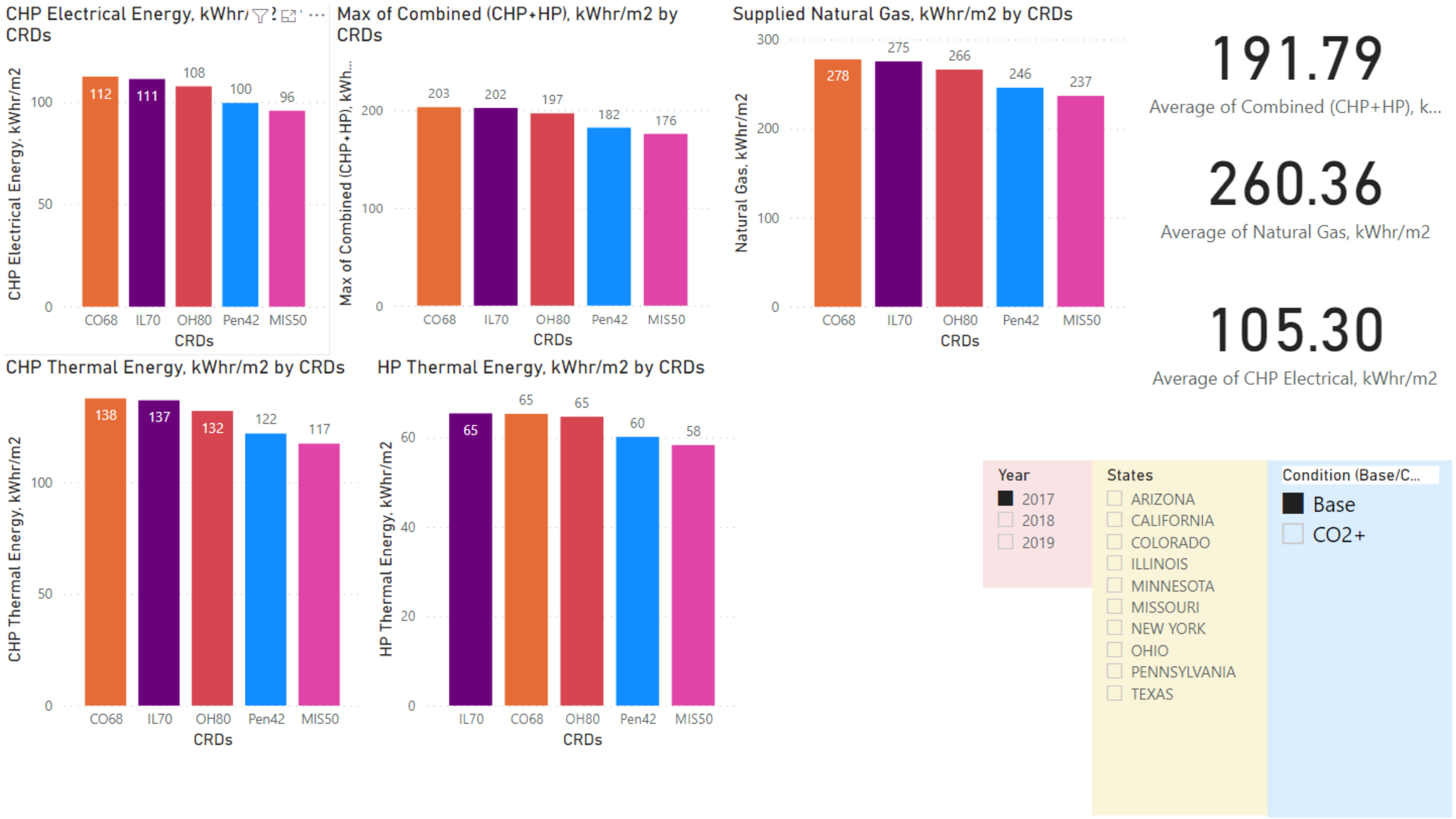


Year	Condition (Base/CO2+)	Average of CHP Electrical, kWhr/m2	Average of Combined (CHP+HP), kWhr/m2	Average of Natural Gas, kWhr/m2
2017	CO2+	102.51	186.13	253.39
2019	CO2+	104.62	187.94	258.55
2018	CO2+	107.71	191.69	266.06
Total		**104.75**	**188.36**	**258.85**

3.3.1 CHP and H.P. Report

To better illustrate the performance of the CHP and the H.P., reports were generated using Power B.I. for each CRD. The average Combined Thermal Energy generated, Natural Gas burned, and CHP Electrical Energy produced for the specific year is listed in the upper right corner. Charts for CHP Electrical Energy, Natural Gas, Combined Thermal Energy (Combined CHP+HP), CHP

Thermal Energy, and H.P. Thermal Energy were created to see the performance across each location. The lower right corner has customizable filters to narrow the report by year, state, and scenario condition. Figs. 21-26 and Tables 14-15 in **Appendix E** show the data in graphical and tabular form. It should be noted that the data for MIS50 in 2018 is missing from the data set due to extreme weather conditions that led to model failure.



191.79
Average of Combined (CHP+HP), k...

260.36
Average of Natural Gas, kWhr/m2

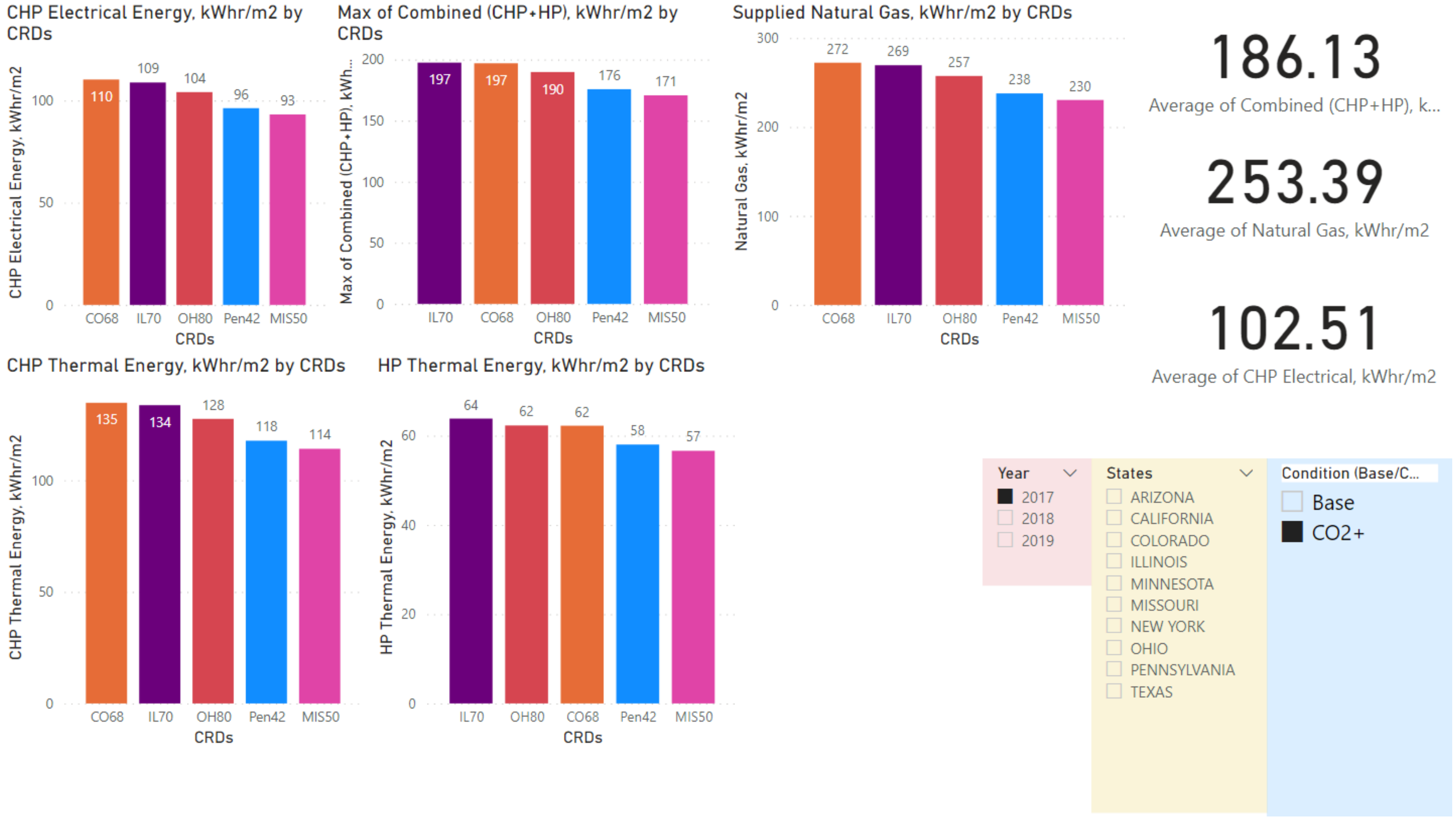
105.30
Average of CHP Electrical, kWhr/m2

Year
■ 2017
☐ 2018
☐ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
■ Base
☐ CO2+

Figure 21:Report of CHP and H.P. performance for 2017 for Base Case.



186.13
Average of Combined (CHP+HP), k...

253.39
Average of Natural Gas, kWhr/m2

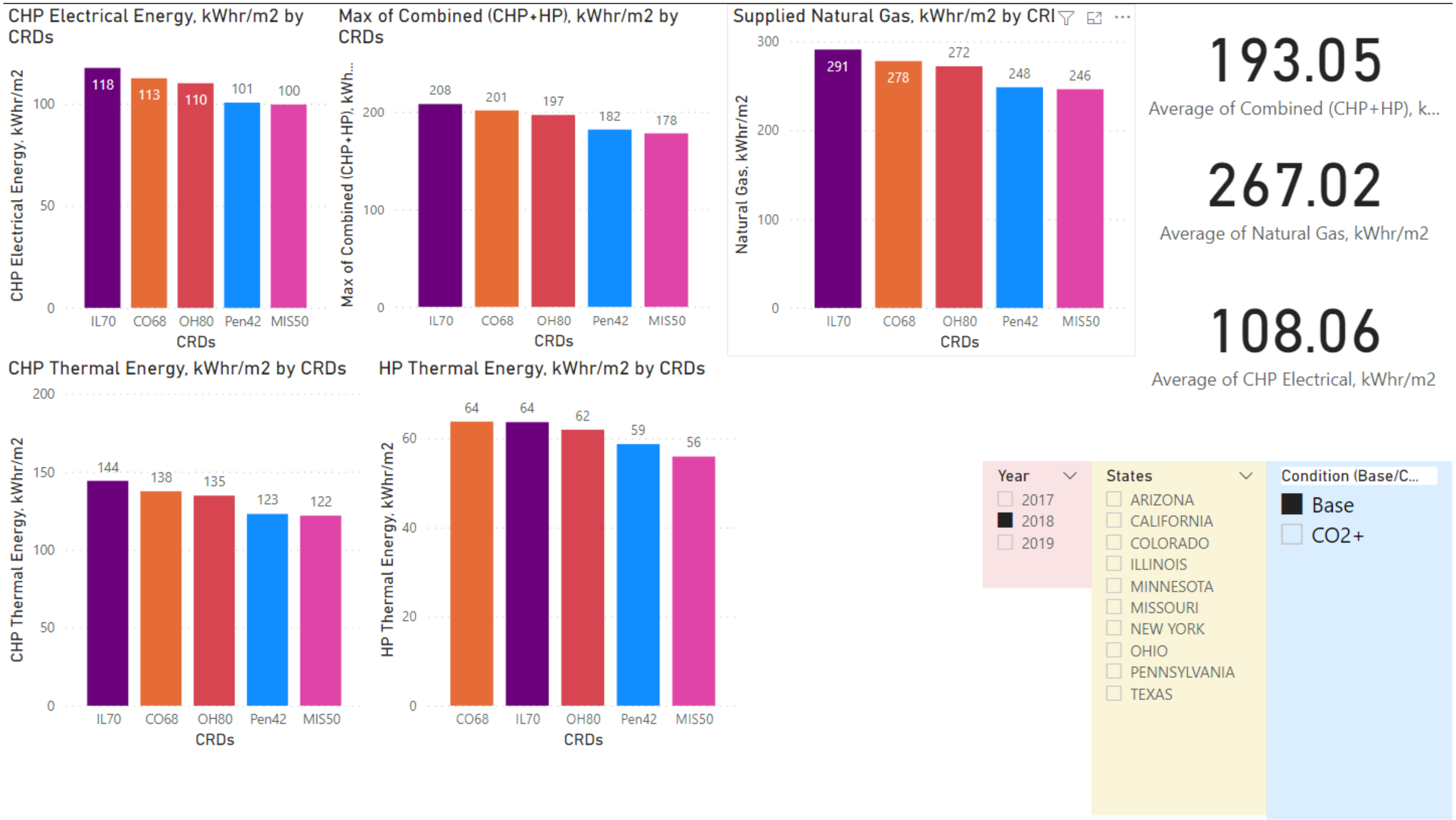
102.51
Average of CHP Electrical, kWhr/m2

Year
■ 2017
☐ 2018
☐ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
☐ Base
■ CO2+

Figure 22:Report of CHP and H.P. performance for 2017 with CO_2 Enrichment using Global System 2.



193.05

Average of Combined (CHP+HP), k...

267.02

Average of Natural Gas, kWhr/m2

108.06

Average of CHP Electrical, kWhr/m2

Year

☐ 2017
■ 2018
☐ 2019

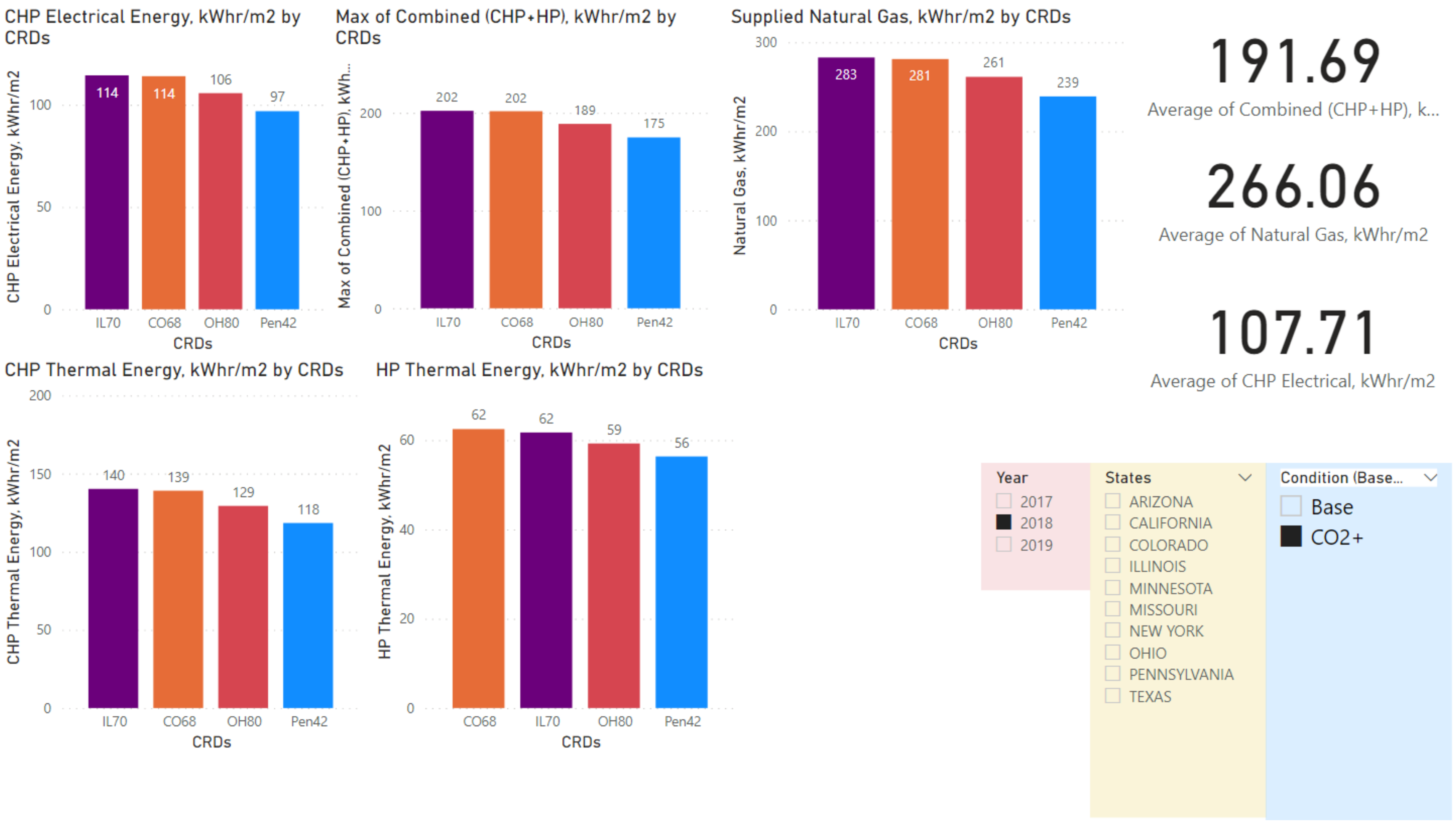
States

☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...

■ Base
☐ CO2+

Figure 23:Report on CHP and H.P. performance for 2018 for Base Case using Global System 2.



191.69
Average of Combined (CHP+HP), k...

266.06
Average of Natural Gas, kWhr/m2

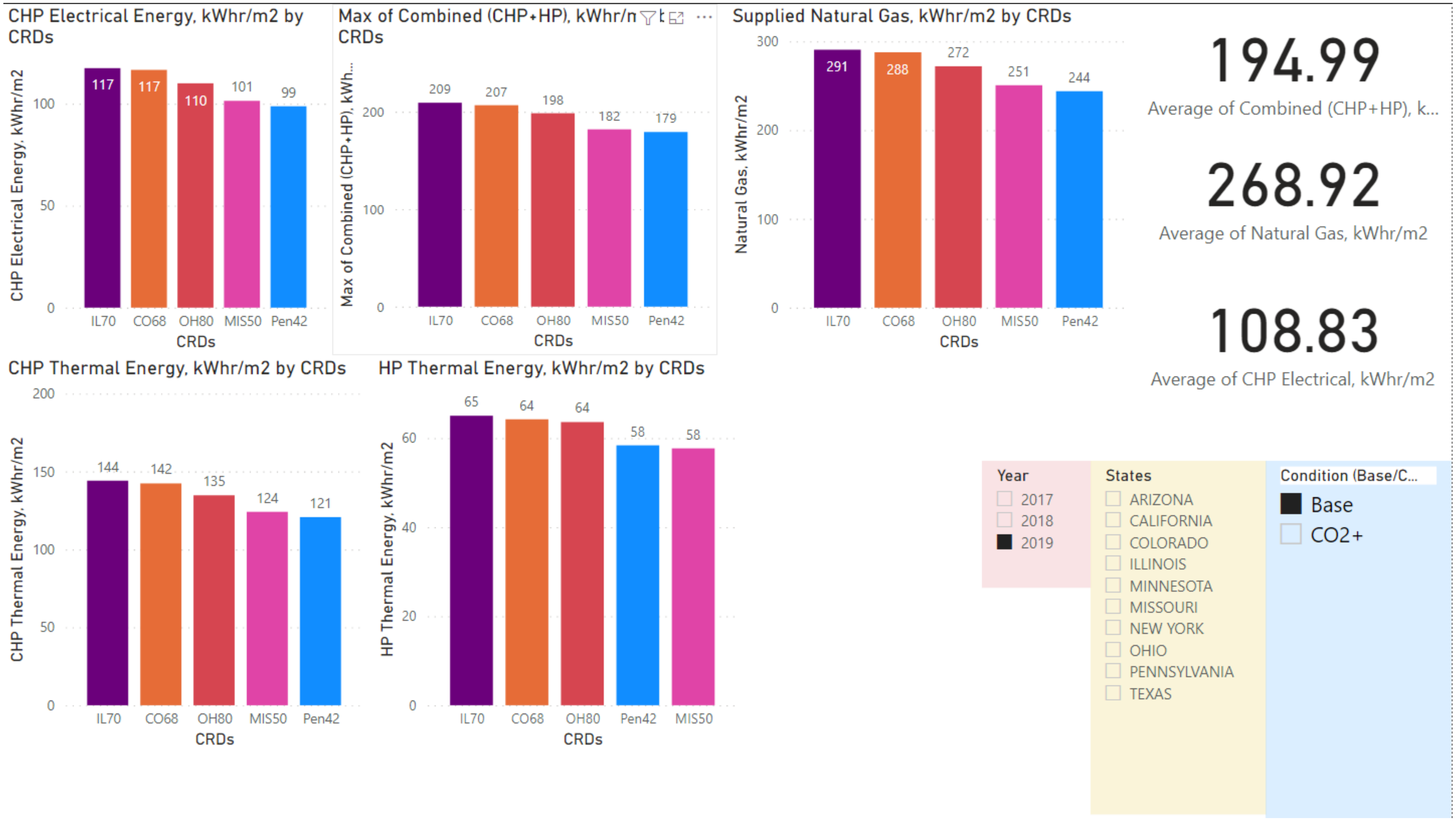
107.71
Average of CHP Electrical, kWhr/m2

Year
☐ 2017
■ 2018
☐ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base...
☐ Base
■ CO2+

Figure 24: Report on CHP and H.P. performance for 2018 with CO_2 Enrichment using Global System 2.



194.99
Average of Combined (CHP+HP), k...

268.92
Average of Natural Gas, kWhr/m2

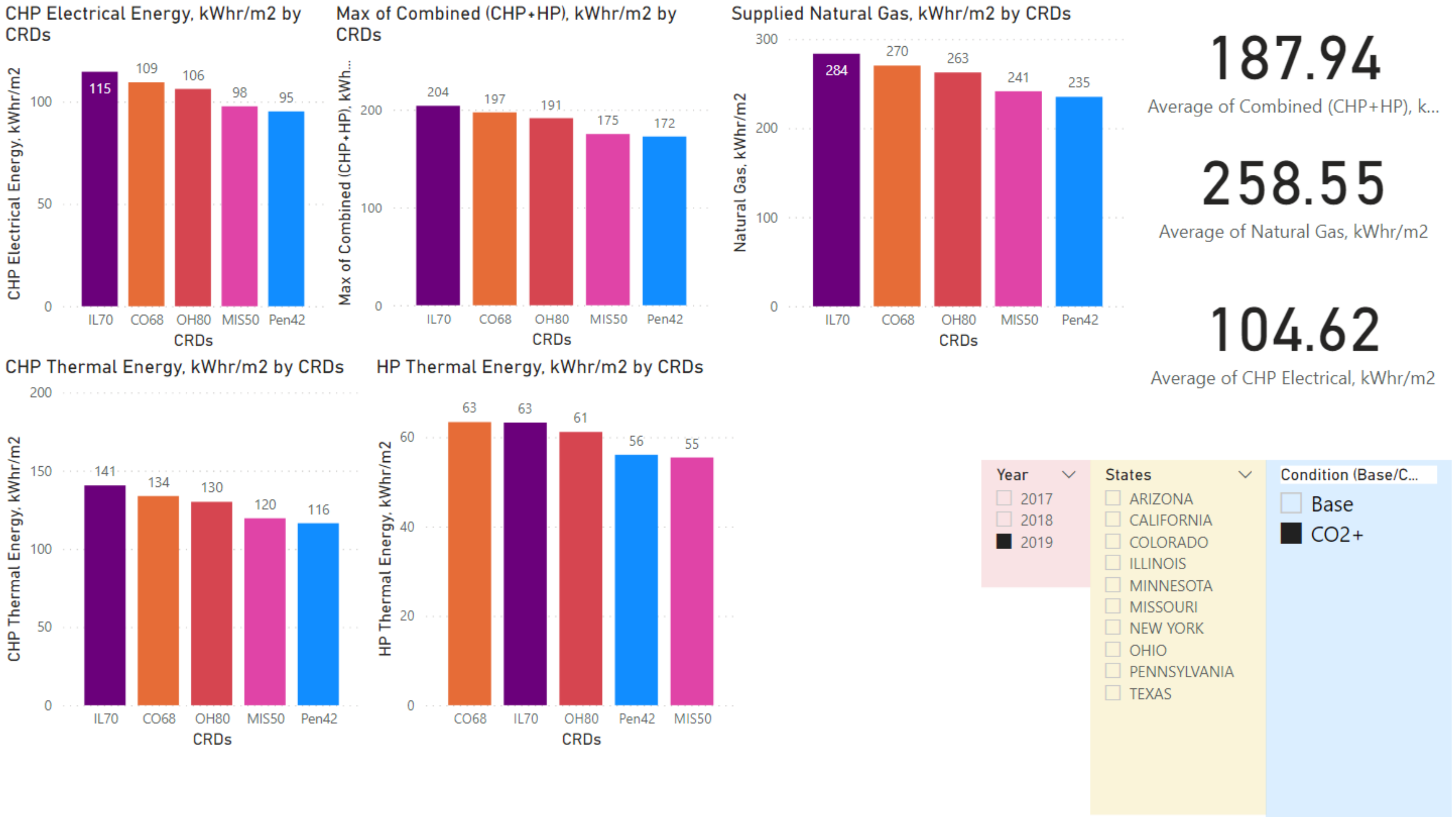
108.83
Average of CHP Electrical, kWhr/m2

Year
☐ 2017
☐ 2018
☒ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
☒ Base
☐ CO2+

Figure 25:Report on CHP and H.P. performance for 2019 for Base Case using Global System 2.



Year
☐ 2017
☐ 2018
■ 2019

States
☐ ARIZONA
☐ CALIFORNIA
☐ COLORADO
☐ ILLINOIS
☐ MINNESOTA
☐ MISSOURI
☐ NEW YORK
☐ OHIO
☐ PENNSYLVANIA
☐ TEXAS

Condition (Base/C...
☐ Base
■ CO2+

Figure 26: Report on CHP and H.P. performance for 2019 with CO_2 Enrichment using Global System 2.

Chapter 4: Discussion

Most CRDs produced yields within the expected range of 50-80 kg/m^2 except for TX30 and AZ80. These two states showed yields below the expected range. CO_2 enrichment positively impacts tomato production by increasing tomato yield. This increased yield productivity reduces the resources used such as electricity, thermal energy, natural gas, and water footprint needed for crop growth than in the Base Case. CO_2 enrichment makes the expansion of greenhouse technology for fruit and vegetable production attractive to all stakeholders in the sector as it makes the return on investment more apparent over time. There were a few locations, such as CA51, and AZ80 according to the data, that did not show improvements in yield, and resulted in lower yields than the Base Case. More work is needed to be done to explore why TX30, AZ80, and CA51 were outliers in the study. The observation suggested the need to adjust the controls in the model or examine other components not altered during the simulation.

The results from the model create a reference for further studies and decisions about greenhouses in the U.S. The data shown supports the use and research of agricultural greenhouse technology for fruit and vegetable production in response to persistent climate variability and change. Opportunities exist to expand greenhouse operations in the U.S. to maintain current and future food supply chains. Future works can include the manipulation of other settings included in the model to improve simulation performance. Tomato growth conditions can be altered to generate data for other tomato varieties not explored in this study, such as beefsteak tomatoes. Cultivation of different crops grown together can be explored and modeled to offset the investment cost of greenhouses. The CHP unit can be sized to produce most of the electricity needed and reduce reliance on the grid for power, further increasing the return on investment. CO_2 needed for carbon enrichment can be collected from the CHP unit instead of purchasing it from an external source. Installation of newer lighting technologies like LEDs can improve the energy efficiency of the system.

4.1 Validation of study

The observed PWU for a semi-closed greenhouse can range as low as 4 L/kg, but usually varies from 12.5-20 L/kg.[19] Greenhouse operations in hot arid regions cooled with pad-and-fan systems, and fogging can range from over 60 to 90 L/kg.[19] In this study, the PWU was as low as 12.18 L/kg and 9.37 L/kg and as high as 45.61L/kg and 57.38L/kg for the base case and CO_2+ case, respectively. On both extremes, the data fell in the expected range for PWU. The effect of a greenhouse on tomato cultivation can see production rates of 10-20 times higher than in an open field. Tomato yield in a greenhouse has been seen to range from 50-80 kg/m^2/yr, while in the open field, it is 3-8kg/m^2/yr.[19] In this study, the yield observed was as low as 35.57 kg/m^2 and 24.24 kg/$m^2$and as high as 81.42 kg/m^2 and 96.05 kg/m^2 for the base case and CO_2+ case, respectively. Data from a study analyzing the LCA for lighting technologies for greenhouse crop production was used to compare the electrical energy used by the simulation.[28] The authors found that the cumulative energy demand (CED) for a 1000W HPS system for 8 ft^2 yields 179722 kWhr (647000 MJ), which converts to 241886.9 kWhr/m^2. The average results for the base case and CO_2+ were 437.90 kWhr/m^2 and 437.84 kWhr/m^2, respectively .[28] Thermal energy provided to greenhouses can be supplied from a variety of fuel sources such as natural gas, and bunker oil, and depending on the design demand varies.[29] Heating demand can be as much as 587.3 kWhr/m^2, with 525.8 kWhr/m^2 being provided by burning of natural gas for an average greenhouse size of 60954 m^2.[29] The average heating demand was 206.62 kWhr/m^2 and 201.09 kWhr/m^2 for the base case and CO_2+, respectively. The average natural gas supplied was 265.43 kWhr/m^2 and 258.85 kWhr/m^2 for the base case and CO_2+, respectively. The simulated results of this study fall within this range of other works in the field, but more work is needed to reduce the heating demand. The CRDs with the lowest performance, such as AZ80 and TX30, need additional technologies to improve productivity which is outside the scope of the study.

4.2 Factors influencing the model

4.2.1 Temperature

The model creators had tested the greenhouse for mean temperature regimes of 14°C, 18°C, 22°C, and 26°C. The temperature regime is significant as anything above the mild temperature, 18°C-22°C, starts registering as non-optimal or extreme weather for tomato growth.[30] Underestimation of fruit growth also begins to appear at lower temperatures.[31] TX30, AZ80, and CA51 had average temperatures above 26°C and, in some years reaching as high as 35°C during the summer months. Greenhouse yield output is affected by the elevated temperature as it creates an inhospitable environment for tomato plants. Additional cooling systems may need to be installed, or current temperature controls need to be revised.

4.2.2 CO_2 enrichment

Externally supplied CO_2 to the tomato plants in the Greenhouse generally resulted in increased fresh weight yield and a reduction of resources for the simulated tomato production. The average natural gas and water supplied to the system for the 3 years saw a noticeable drop of 6.1 $kWhr/m^2$ and $126.11 L/m^2$, respectively. The electrical and thermal energy demands also saw small reductions over the years observed. In the case of TX30 and AZ80, additional work is needed to understand their lower yields which are outside the scope of this study. The locations likely had high day temperatures, and high PAR levels resulted in their diminished harvest. Photosynthesis was inhibited by the carbohydrate buffer saturation, which resulted in lower crop yield values.[30] This phenomenon was seen as the levels of carbon enrichment were elevated compared to the other CRDs in the study.

4.2.3 Supplemental Lighting

Most of the electrical energy consumed by the greenhouse models was for the supplemental lighting used. High Pressure Sodium (HPS) lamps have been the standard for years for greenhouses since the lamps are most efficient in the PAR spectrum range, with emissions highly concentrated between 500 and 650nm. HPS lighting is not designed for frequent cycling and needs to be set up to operate for extended periods.[11] However, with the evolution of lighting technologies, better alternatives are on the horizon with the use of LEDs which are more energy-efficient and have a longer life span. LEDs allow for better control of plant growth with reduced energy consumption. The light spectrum can be customized to specific colors to induce improved plant response.[32] The model creators have updated their approach to meet new environmentally friendly and energy-efficient requirements by modifying the system to combine HPS and LED lighting to produce similar results with reduced energy consumption.[33] Additional components were proposed to collect the wasted thermal energy from HPS lamps to heat the greenhouse.[33] The developers of the Modelica model should adopt these new improvements in their model, which will be more helpful with the current requirements.

Chapter 5: Limitations

The work presented in this study utilizing the Modelica Greenhouse Library provides a necessary insight into the viability of greenhouse cultivation across the continental U.S. However, there are a few limitations to the model. The greenhouse library aims to provide an open-source tool for modeling greenhouse climate with the possibility of energy integration with thermal and power systems.[18] Greenhouse climate modeling is complex and ongoing improvements are necessary to meet the demands of farmers and other interested parties. As a result, the work does not focus on climate setpoint optimization, so no innovative control is presented. Better controls are necessary for improved performance. The library is limited in the number of modeled HVAC systems, but allows for further integration with other thermal systems such as ThermoCycle and ThermoPower.[18]

The model creators have recently published a paper on improvements to their original model that the developers could integrate into the Greenhouse Library. These improvements range from combining HPS and LED lighting together to harvesting heat wasted from the lighting.[33] Redirecting wasted heat energy to keep the greenhouse heated would reduce the heat generated from the CHP unit and operation costs of the greenhouse. The use of LED lighting would significantly reduce the electricity used by the greenhouse and the number of replacement lamps needed during production. However, additional data affecting plant growth will be necessary to account for the shift from HPS lamps in the Greenhouse Library.

This study does not consider the energy required for CO2 enrichment, such as combustion from natural gas for a regulated flow of CO_2. The model assumes that CO_2 is continuously supplied externally and does not specify if it is collected from the CHP unit or purchased separately. The missing data would improve understanding of the environmental and energy cost of elevated CO_2 levels for improved tomato yield. The total energy input would likely increase, resulting in the energy intensity increased for the CO2+ scenario.

As previously noted in the results section, while simulating results for some CRDs, the model failed shortly after being initiated. The reason seems to be because of the extreme external weather that the greenhouse model had to overcome to heat the inside environment was high under current settings. This resulted in a switch from Global System_2 to Greenhouse_1 to generate results. This affected the comparison of the performance of the heating system. This behavior was noted for NY60, and MN90 for all three years, but only in 2018 for MIS50. Locations such as TX30, CA51, and AZ80 were not modeled with the heating system of Global System_2 because the outside weather was warmer than the other locations throughout the year, reducing the need for additional heat. Greenhouse_1 does output how much thermal energy is needed during plant growth. Those locations may need more cooling technology or controls to manage the outside temperature. For the CO2+ scenario, CA51 and AZ80 saw decreased yields below the expected range for greenhouse grown tomatoes. CO68 in 2017 also saw a yield decrease but for that year alone. Further investigation outside the scope of this study is needed to understand why CO_2 enrichment reduced the yield for those locations.

The water footprint reported is an estimation as the greenhouse model did not have an explicit component measuring water usage. The model assumed a non-limiting irrigation strategy for the growth period. To quantify the possible water used, plant transpiration data was converted using the trapezoidal rule to identify the water used. The same strategy was used to generate a possible carbon footprint. The model does not report any CO_2 emissions during the growth period, but relies on CO_2 to be externally supplied for CO_2 enrichment. The CO_2 supplied was used to identify the carbon footprint during operation. To generate the fresh weight tomato yield, it was assumed that the tomatoes had a 5% dry mass. This assumption allowed the conversion of the dry matter harvested data output from the simulation to quantify the fresh weight. Real grown tomatoes may have a different ratio than the assumption stated. Lastly, the model does not specify any assumption or component on fertilizer inputs or crop nutrient inputs, which would assist in quantifying needs for operation.

The results generated are for typical greenhouse grown tomatoes such as cherry and grape tomatoes. Future works can explore other tomato varieties to see their performance in the greenhouse model under those different growing conditions. The data would allow interested parties to select the best varieties to grow for consumers and their respective markets. The possibility of growing different varieties of tomatoes or other crops together might present a compelling case for fruit and vegetable production or identify unseen limitations. The CHP unit provides a portion of the electrical demand, and most of what is generated is sold back to the grid. Farmers would be more interested in a design that generates most of the electrical and thermal demand to reduce their expenses in the long term. The opportunity for harvesting CO_2 exhausted from the CHP unit might be another avenue of interest for researchers and investors.

Chapter 6: Conclusion

The results and findings presented in this study fill the gap of greenhouse tomato production in the U.S. The data from the 10 CRDs establishes a reference for future works in modeling tomato production and informing stakeholders in the fruit and vegetable supply chains on the opportunities that greenhouses offer for current and future food supply. The average fresh weigh yield for the Base Case from 2017-2019 was 63.67 kg/m2,55.68 kg/m2, and 59.10 kg/m2, respectively. The average fresh weight yield for the CO2+ from 2017-2019 was 67.19 kg/m2,61.37 kg/m2, and 65.30 kg/m2, separately. The CO68 and MN90 had the highest yields in both the Base and CO2+ scenarios, and AZ80 had the lowest. In both cases, the average yield falls within the expected range for greenhouse grown tomatoes. The average PWU for the Base Case from 2017-2019 was 19.91 L/kg, 21.68 L/kg, and 20.40 L/kg, respectively. The average PWU for the CO2+ from 2017-2019 was 18.76 L/kg, 18.86 L/kg, and 17.80 L/kg, separately. AZ80 had the highest PWU for both cases reaching as high as 57 L/kg, and MN90 had the lowest PWU in both cases, dropping as low as 9 L/kg. The PWU performance corresponds with the fresh weight yield. The Product Water Usage (PWU) falls within the expected range; even AZ80's results correspond with what has been observed with arid locations. PEN42 had the highest electricity usage of 439 kWhr/m2, and IL70 had the lowest of 437 kWhr/m2 in both scenarios. The average heating demand was 206.62 kWhr/m^2 and 201.09 kWhr/m^2 for the base case and CO_2+, respectively. MN90 had the highest heating demand reaching as high as 309 kWhr/m^{2,} and AZ80 had the lowest with 152 kWhr/m^2. The electrical and thermal energy used by the greenhouse model were lower than what has been used by other greenhouses.

The CHP unit can be sized to provide more of the electricity required to operate the greenhouse since the current design allows for a portion of the demand. CO_2 enrichment benefits the plants and farmers by reducing the water uptake, natural gas, electrical, and thermal energy demand. Temperature was the major factor affecting the simulation, with CO_2 enrichment and supplemental lighting also impacting production. Further research is needed to improve the Greenhouse Library to integrate the HPS and LED hybrid supplemental lighting, recycle exhausted CO_2 from the CHP to the canopy, capture thermal energy wasted by supplemental lighting for heating, and other developments for greenhouse cultivation. Some many opportunities and insights await with the use of greenhouses for the U.S. food supply.

References

(1) Mickelbart, M. V.; Hasegawa, P. M.; Bailey-serres, J. Genetic mechanisms of abiotic stress tolerance that translate to crop yield stability. *Nature Reviews. Genetics* **2015**, *16* (4), 237-251. DOI: http://dx.doi.org/10.1038/nrg3901 Medical Database; ProQuest Central.

(2) Baskins, S.; Bond, J. K.; Minor, T. *Unpacking the Growth in Per Capita Availabilty of Fresh Market Tomatoes*; United States Department of Agriculture, 2019. https://www.ers.usda.gov/publications/pub-details/?pubid=92441 (accessed March 2020).

(3) U.S. Department of Agriculture, U.; Economic Research Service, E. *Vegetable and Pulses Yearbook Tables, 1970-2018.* https://www.ers.usda.gov/webdocs/outlooks/92442/vgs-19c-01.pdf?v=3768.2 (accessed 2020/03/20).

(4) Lopez, J. A. A Demand Analysis for Fresh Tomatoes in the Dallas/Fort Worth Grocery Market. *Texas Journal of Agriculture and Natural Resources* **2017**, *30* (0). (acccessed 2020/03/20).

(5) Sturm, R.; An, R. Obesity and economic environments. *CA: A Cancer Journal for Clinicians* **2014**, *64* (5), 337-350. DOI: 10.3322/caac.21237.

(6) Dehghan, M.; Akhtar-Danesh, N.; Merchant, A. T. Factors associated with fruit and vegetable consumption among adults. *Journal of Human Nutrition and Dietetics* **2011**, *24* (2), 128-134. DOI: 10.1111/j.1365-277X.2010.01142.x.

(7) Onozaka, Y.; Nurse, G.; McFadden, D. T. Local Food Consumers: How Motivations and Perceptions Translate to Buying Behavior. *Choices* **2010**, *25* (1). (acccessed 2020/03/20).JSTOR.

(8) Calvin, L.; Thornsbury, S.; Cook, R. Protected-Culture Technology Transforms the Fresh-Tomato Market. *Amber Waves* **2013**, 63.

(9) U.S. Department of Agriculture, U.; Foreign Agricultural Service, F. Mexico Continues to Expand Greenhouse Tomato Production. 2016.

(10) Division, C. a. H.; Canada, A. a. A.-F. Statistical Overview of the Canadian Greenhouse Vegetable Industry. 2019.

(11) Altes-Buch, Q.; Lemort, V. Modeling framework for the simulation and control of greenhouse climate. In *Proceedings of the 10th International Conference on System Simulation in Buildings*, 2018; p 20.

(12) Bot, G. P. A. Greenhouse climate : from physical processes to a dynamic model. Bot, Wageningen, 1983. https://edepot.wur.nl/188427.

(13) Impron, I.; Hemming, S.; Bot, G. P. A. Simple greenhouse climate model as a design tool for greenhouses in tropical lowland. *Biosystems Engineering* **2007**, *98* (1), 79-89. DOI: https://doi.org/10.1016/j.biosystemseng.2007.03.028.

(14) Greenhouses Modelica Library Documentation. University of Liege: 2019; Vol. 2021.

(15) de Zwart, H. F. Analyzing Energy-Saving Options in Greenhouse Cultivation Using a Simulation Model. Ph.D., Wageningen University and Research, Ann Arbor, 1996. https://www.proquest.com/dissertations-theses/analyzing-energy-saving-options-greenhouse/docview/2465452813/se-2?accountid=8361.

(16) Vanthoor, B. H. E.; Stanghellini, C.; van Henten, E. J.; de Visser, P. H. B. A methodology for model-based greenhouse design: Part 1, a greenhouse climate model for a broad range of designs and climates. *Biosystems Engineering* **2011**, *110* (4), 363-377. DOI: https://doi.org/10.1016/j.biosystemseng.2011.06.001.

(17) Group, V. A. H. *Venlo Greenhouses: Economical Quality*. https://www.venloinc.com/greenhouses/venlo (accessed 2021 11/12/21).

(18) Altes-Buch, Q.; Quoilin, S.; Lemort, V. Greenhouses: A Modelica library for the simulation of greenhouse climate and energy systems. In *Proceedings of the 13th International Modelica Conference*, 2019.

(19) Nederhoff, E.; Stanghellini, C. *Water Use Efficiency of Tomatoes*; Casper Publications, 2010. DOI: 10.3316/informit.484066941930798.

(20) Altes-Bucha, Q.; Quoilina, S.; Lemort, V. Modeling and control of CHP generation for greenhouse cultivation including thermal energy storage. 2018.

(21) U.S. Department of Agriculture, U.; National Agricultural Statistics Service, N. Quick Stats. 2018.

(22) Sengupta, M.; Xie, Y.; Lopez, A.; Habte, A.; Maclaurin, G.; Shelby, J. The National Solar Radiation Data Base (NSRDB). *Renewable and Sustainable Energy Reviews* **2018**, *89*, 51-60. DOI: https://doi.org/10.1016/j.rser.2018.03.003.

(23) Fuentes, M. K. *A simplified thermal model for flat-plate photovoltaic arrays*; Sandia National Labs., Albuquerque, NM (USA), 1987.

(24) MathWorks. *MATLAB Live Editor*. 2021. https://www.mathworks.com/products/matlab/live-editor.html?s_tid=pl_ml_editor_banner02 (accessed 2020 10/16/20).

(25) Aaslyng, J. M.; Lund, J. B.; Ehler, N.; Rosenqvist, E. IntelliGrow: a greenhouse component-based climate control system. *Environmental Modelling & Software* **2003**, *18* (7), 657-666. DOI: https://doi.org/10.1016/S1364-8152(03)00052-5.

(26) Dieleman, J. A.; Marcelis, L. F. M.; Elings, A.; Dueck, T. A.; Meinen, E. ENERGY SAVING IN GREENHOUSES: OPTIMAL USE OF CLIMATE CONDITIONS AND CROP MANAGEMENT. 2006; International Society for Horticultural Science (ISHS), Leuven, Belgium: pp 203-210. DOI: 10.17660/ActaHortic.2006.718.22.

(27) Beck, H. E.; Zimmermann, N. E.; McVicar, T. R.; Vergopolan, N.; Berg, A.; Wood, E. F. Present and future Köppen-Geiger climate classification maps at 1-km resolution. *Scientific Data* **2018**, *5* (1), 180214. DOI: 10.1038/sdata.2018.214.

(28) Zhang, H.; Burr, J.; Zhao, F. A comparative life cycle assessment (LCA) of lighting technologies for greenhouse crop production. *Journal of Cleaner Production* **2017**, *140*, 705-713. DOI: https://doi.org/10.1016/j.jclepro.2016.01.014.

(29) Hendricks, P. Life cycle assessment of greenhouse tomato (Solanum lycopersicum L.) production in Southwestern Ontario. 2012. http://hdl.handle.net/10214/4052.

(30) Vanthoor, B.; De Visser, P.; Stanghellini, C.; Van Henten, E. A methodology for model-based greenhouse design: Part 2, description and validation of a tomato yield model. *Biosystems engineering* **2011**, *110* (4), 378-395.

(31) de Koning, A. N. M. Development and Dry Matter Distribution in Glasshouse Tomato : A Quantitative Approach. Ph.D., Wageningen University and Research, Ann Arbor, 1994. https://www.proquest.com/dissertations-theses/development-dry-matter-distribution-glasshouse/docview/2480353083/se-2?accountid=8361.

(32) Hao Zhang and John Burr and Fu, Z. A comparative life cycle assessment (LCA) of lighting technologies for greenhouse crop production. *Journal of Cleaner Production* **2017**, *140*, 705-713. DOI: https://doi.org/10.1016/j.jclepro.2016.01.014.

(33) Righini, I.; Vanthoor, B.; Verheul, M. J.; Naseer, M.; Maessen, H.; Persson, T.; Stanghellini, C. A greenhouse climate-yield model focussing on additional light, heat harvesting and its validation. *Biosystems Engineering* **2020**, *194*, 1-15. DOI: https://doi.org/10.1016/j.biosystemseng.2020.03.009.

(34) *Dlugokencky, E.; Tans, P. Trends in Atmospheric Carbon Dioxide*. NOAA/GML, 2021. https://gml.noaa.gov/ccgg/trends/gl_data.html (accessed 2021 10/05/21).

Appendix A:Adjusted Parameters

Table 122:Monthly Ambient CO_2 levels in ppm.[34]

Month/Year	2017	2018	2019
Jan	405.32	407.55	409.91
Feb	405.91	408.23	410.34
Mar	406.31	408.76	410.89
Apr	406.6	409.07	411.34
May	406.65	408.98	411.34
June	405.86	408.09	410.53
July	404.11	406.48	408.88
August	402.57	405.1	407.64
Sept	402.66	405.17	407.92
Oct	404.16	406.64	409.44
Nov	405.7	408.12	410.88
Dec	406.75	409.17	411.75
Average	405.22	407.61	410.07

Table 13: CO_2 Concentration Setpoint for Weather and Setpoint file.[14, 18]

Time, hr	CO2_air_sp, ppm	CO2_sp, ppm
0	800	1000
0.5	800	1000
1	800	400
1.5	800	400
2	800	400
2.5	800	400
3	800	400
3.5	800	400
4	800	400
4.5	800	400
5	800	400
5.5	800	400
6	800	400
6.5	800	1300
7	800	1300
7.5	800	1300
8	800	1300
8.5	800	1300
9	800	1300
9.5	700	1300
10	700	1300

Table 13 (cont.): CO_2 Concentration Setpoint for Weather and Setpoint file.[14, 18]

10.5	700	1300
11	700	1000
11.5	700	1000
12	700	1000
12.5	700	1000
13	700	1000
13.5	700	1000
14	700	1000
14.5	700	1000
15	700	1000
15.5	800	1000
16	800	1100
16.5	800	1100
17	800	1300
17.5	800	1300
18	800	1300
18.5	800	1300
19	800	1300
19.5	800	1300
20	800	1300
20.5	800	1300
21	800	1300

Table 13 (cont.): CO_2 Concentration Setpoint for Weather and Setpoint file.[14, 18]

Time, hr	Weather, ppm	Externally Supplied, ppm
21.5	800	1300
22	800	1300
22.5	800	1300
23	800	400
23.5	800	400
24	800	400

Appendix B: Greenhouses Library Source Code

“Greenhouse_1.mo”

```
within Greenhouses.Examples;
model Greenhouse_1
  "Simulation of a Venlo-type greenhouse for tomato crop cultivated from 10Dec-
22Nov (weather data from TMY)"
  extends Modelica.Icons.Example;
  Modelica.SIunits.HeatFlux q_low;
  Modelica.SIunits.HeatFlux q_up;
  Modelica.SIunits.HeatFlux q_tot;

  Real E_th_tot_kWhm2(unit="kW.h/m2");
  Real E_th_tot(unit="kW.h");

  Real DM_Har(unit="mg/m2") "Accumulated harvested tomato dry matter";

  Real W_el_illu(unit="kW.h/m2");
  Real E_el_tot_kWhm2(unit="kW.h/m2");
  Real E_el_tot(unit="kW.h");

  Components.Greenhouse.Cover cover(
    rho=2600,
```

```
  c_p=840,
  A=surface.k,
  steadystate=true,
  h_cov=1e-3,
  phi=0.43633231299858)
  annotation (Placement(transformation(extent={{22,112},{50,140}})));
Components.Greenhouse.Air air(
  A=surface.k,
  steadystate=true,
  steadystateVP=true,
  h_Air=h_Air.y)
  annotation (Placement(transformation(extent={{66,-58},{94,-30}})));
Components.Greenhouse.Canopy canopy(
  A=surface.k,
  steadystate=true,
  LAI=TYM.LAI)
  annotation (Placement(transformation(extent={{-78,-76},{-48,-48}})));
Flows.HeatTransfer.Radiation_T4 Q_rad_CanCov(
  A=surface.k,
  epsilon_a=1,
  epsilon_b=0.84,
  FFa=canopy.FF,
  FFb=1,
```

```
  FFab1=pipe_up.FF,
  FFab2=thScreen.FF_ij)
  annotation (Placement(transformation(extent={{-4,-82},{16,-62}})));
Components.Greenhouse.Floor floor(
  rho=1,
  c_p=2e6,
  A=surface.k,
  V=0.01*surface.k,
  steadystate=true)
  annotation (Placement(transformation(extent={{-182,-168},{-156,-142}})));

Flows.HeatTransfer.Radiation_T4 Q_rad_FlrCan(
  A=surface.k,
  epsilon_a=0.89,
  epsilon_b=1,
  FFa=1,
  FFb=canopy.FF,
  FFab1=pipe_low.FF) annotation (Placement(transformation(
      extent={{-10,-10},{10,10}},
      rotation=0,
      origin={-100,-84})));

Flows.HeatTransfer.CanopyFreeConvection Q_cnv_CanAir(A=surface.k, LAI=canopy.LAI)
```

```
  annotation (Placement(transformation(extent={{-4,-62},{16,-42}})));
Flows.HeatTransfer.FreeConvection Q_cnv_FlrAir(
  phi=0,
  A=surface.k,
  floor=true)
  annotation (Placement(transformation(extent={{50,-166},{70,-146}})));
Flows.HeatTransfer.Radiation_T4 Q_rad_CovSky(
  epsilon_a=0.84,
  epsilon_b=1,
  A=surface.k)
  annotation (Placement(transformation(extent={{76,146},{96,166}})));
Modelica.Thermal.HeatTransfer.Celsius.PrescribedTemperature out
  annotation (Placement(transformation(extent={{188,116},{176,128}})));
Flows.HeatTransfer.OutsideAirConvection Q_cnv_CovOut(
  A=surface.k,
  u=u_wind.y,
  phi=0.43633231299858) annotation (Placement(transformation(
      extent={{10,10},{-10,-10}},
      rotation=180,
      origin={86,140})));
Components.Greenhouse.Illumination illu(
  A=surface.k,
  power_input=true,
```

```
  LAI=TYM.LAI,
  P_el=500,
  p_el=100)
  annotation (Placement(transformation(extent={{-182,24},{-162,44}})));
Flows.HeatTransfer.Radiation_T4 Q_rad_FlrCov(
  A=surface.k,
  epsilon_a=0.89,
  FFa=1,
  FFab1=pipe_low.FF,
  FFab2=canopy.FF,
  epsilon_b=0.84,
  FFb=1,
  FFab3=pipe_up.FF,
  FFab4=thScreen.FF_ij) annotation (Placement(transformation(
      extent={{-10,-10},{10,10}},
      rotation=0,
      origin={6,-146})));

Modelica.Blocks.Sources.Constant surface(k=1.4e4)
  annotation (Placement(transformation(extent={{-216,106},{-204,118}})));
Flows.Sources.Vapour.PrescribedPressure prescribedVPout
  annotation (Placement(transformation(extent={{188,74},{176,86}})));
Flows.VapourMassTransfer.MV_CanopyTranspiration MV_CanAir(
```

```
  A=surface.k,
  LAI=canopy.LAI,
  CO2_ppm=CO2_air.CO2_ppm,
  R_can=solar_model.R_t_Glob + illu.R_PAR + illu.R_NIR,
  T_can=canopy.T)
  annotation (Placement(transformation(extent={{-4,-100},{16,-80}})));
Flows.HeatTransfer.SoilConduction Q_cd_Soil(
  N_c=2,
  N_s=5,
  lambda_c=1.7,
  lambda_s=0.85,
  A=surface.k,
  steadystate=false)
  annotation (Placement(transformation(extent={{-126,-176},{-106,-156}})));

Flows.HeatTransfer.Radiation_T4 Q_rad_CanScr(
  A=surface.k,
  epsilon_a=1,
  epsilon_b=1,
  FFa=canopy.FF,
  FFab1=pipe_up.FF,
  FFb=thScreen.FF_i)
  annotation (Placement(transformation(extent={{-90,-74},{-110,-54}})));
```

```
Flows.HeatTransfer.Radiation_T4 Q_rad_FlrScr(
  A=surface.k,
  epsilon_b=1,
  FFab1=canopy.FF,
  FFab2=pipe_up.FF,
  epsilon_a=0.89,
  FFa=1,
  FFab3=pipe_low.FF,
  FFb=thScreen.FF_i)
  annotation (Placement(transformation(extent={{-156,-130},{-136,-110}})));

Components.Greenhouse.ThermalScreen thScreen(
  A=surface.k,
  SC=SC.y,
  steadystate=false)
  annotation (Placement(transformation(extent={{-134,50},{-104,82}})));
Flows.HeatTransfer.Radiation_T4 Q_rad_ScrCov(
  A=surface.k,
  FFb=1,
  epsilon_a=1,
  epsilon_b=0.84,
  FFa=thScreen.FF_i)
```

```
  annotation (Placement(transformation(extent={{-40,124},{-20,144}})));
Components.Greenhouse.Air_Top air_Top(
  steadystate=true,
  steadystateVP=true,
  h_Top=0.4,
  A=surface.k)
  annotation (Placement(transformation(extent={{-56,86},{-26,114}})));
Components.Greenhouse.Solar_model solar_model(
  A=surface.k,
  LAI=TYM.LAI,
  SC=SC.y)
  annotation (Placement(transformation(extent={{-200,130},{-178,152}})));
Components.Greenhouse.HeatingPipe pipe_low(
  d=0.051,
  freePipe=false,
  A=surface.k,
  flow1DimInc(steadystate=false),
  N=5,
  N_p=625,
  l=50) annotation (Placement(transformation(extent={{-34,-140},{-64,-110}})));
Flows.HeatTransfer.Radiation_N Q_rad_LowFlr(
  A=surface.k,
  epsilon_a=0.88,
```

```
  FFa=pipe_low.FF,
  epsilon_b=0.89,
  FFb=1,
  N=pipe_low.N)
              annotation (Placement(transformation(
    extent={{-10,-10},{10,10}},
    rotation=180,
    origin={-100,-134})));
Flows.HeatTransfer.Radiation_N Q_rad_LowCan(
  A=surface.k,
  epsilon_b=1,
  FFb=canopy.FF,
  epsilon_a=0.88,
  FFa=pipe_low.FF,
  N=pipe_low.N)
              annotation (Placement(transformation(
    extent={{-10,-10},{10,10}},
    rotation=90,
    origin={-50,-98})));
Flows.HeatTransfer.Radiation_N Q_rad_LowCov(
  A=surface.k,
  epsilon_a=0.88,
  FFa=pipe_low.FF,
```

```
  epsilon_b=0.84,
  FFb=1,
  FFab1=canopy.FF,
  FFab2=pipe_up.FF,
  FFab3=thScreen.FF_ij,
  N=pipe_low.N)    annotation (Placement(transformation(
      extent={{-10,-10},{10,10}},
      rotation=0,
      origin={6,-134})));
Flows.HeatTransfer.PipeFreeConvection_N Q_cnv_LowAir(
  A=surface.k,
  d=pipe_low.d,
  freePipe=false,
  N_p=pipe_low.N_p,
  l=pipe_low.l,
  N=pipe_low.N)
  annotation (Placement(transformation(extent={{-4,-124},{16,-104}})));
Flows.HeatTransfer.Radiation_N Q_rad_LowScr(
  A=surface.k,
  epsilon_a=0.88,
  FFa=pipe_low.FF,
  epsilon_b=1,
  FFb=thScreen.FF_i,
```

```
  FFab1=canopy.FF,
  FFab2=pipe_up.FF,
  N=pipe_low.N)     annotation (Placement(transformation(
      extent={{-10,-10},{10,10}},
      rotation=180,
      origin={-100,-114})));
Components.Greenhouse.HeatingPipe pipe_up(
  A=surface.k,
  freePipe=true,
  d=0.025,
  l=44,
  flow1DimInc(steadystate=false),
  N=5,
  N_p=292) annotation (Placement(transformation(extent={{-58,-28},{-28,2}})));
Flows.HeatTransfer.Radiation_N Q_rad_UpFlr(
  A=surface.k,
  epsilon_a=0.88,
  epsilon_b=0.89,
  FFb=1,
  FFa=pipe_up.FF,
  FFab1=canopy.FF,
  FFab2=pipe_low.FF,
  N=pipe_up.N)  annotation (Placement(transformation(
```

```
      extent={{-10,-10},{10,10}},
      rotation=180,
      origin={-100,-24})));
Flows.HeatTransfer.Radiation_N Q_rad_UpCan(
  A=surface.k,
  epsilon_a=0.88,
  epsilon_b=1,
  FFa=pipe_up.FF,
  FFb=canopy.FF,
  N=pipe_up.N)   annotation (Placement(transformation(
      extent={{-10,-10},{10,10}},
      rotation=-90,
      origin={-42,-48})));
Flows.HeatTransfer.Radiation_N Q_rad_UpCov(
  A=surface.k,
  epsilon_a=0.88,
  epsilon_b=0.84,
  FFb=1,
  FFa=pipe_up.FF,
  FFab1=thScreen.FF_ij,
  N=pipe_up.N)     annotation (Placement(transformation(
      extent={{-10,-10},{10,10}},
      rotation=0,
```

```
    origin={6,-4})));
Flows.HeatTransfer.PipeFreeConvection_N Q_cnv_UpAir(
  A=surface.k,
  d=pipe_up.d,
  l=pipe_up.l,
  freePipe=true,
  N_p=pipe_up.N_p,
  N=pipe_up.N)
  annotation (Placement(transformation(extent={{-4,-34},{16,-14}})));
Flows.HeatTransfer.Radiation_N Q_rad_UpScr(
  A=surface.k,
  epsilon_a=0.88,
  FFa=pipe_low.FF,
  epsilon_b=1,
  FFb=thScreen.FF_i,
  N=pipe_up.N)  annotation (Placement(transformation(
    extent={{-10,-10},{10,10}},
    rotation=180,
    origin={-100,-4})));
Flows.HeatAndVapourTransfer.Convection_Condensation Q_cnv_AirScr(
  phi=0,
  A=surface.k,
  floor=false,
```

```
  thermalScreen=true,
  Air_Cov=false,
  SC=SC.y) annotation (Placement(transformation(extent={{-72,32},{-92,12}})));
Flows.HeatAndVapourTransfer.Convection_Condensation Q_cnv_AirCov(
  A=surface.k,
  floor=false,
  thermalScreen=true,
  Air_Cov=true,
  topAir=false,
  SC=SC.y,
  phi=0.43633231299858)
  annotation (Placement(transformation(extent={{70,78},{50,58}})));
Flows.HeatAndVapourTransfer.Convection_Condensation Q_cnv_TopCov(
  A=surface.k,
  floor=false,
  thermalScreen=true,
  Air_Cov=true,
  topAir=true,
  SC=SC.y,
  phi=0.43633231299858)
  annotation (Placement(transformation(extent={{-14,108},{6,128}})));
Flows.HeatAndVapourTransfer.Ventilation Q_ven_AirOut(
  A=surface.k,
```

```
  thermalScreen=true,
  topAir=false,
  u=u_wind.y,
  U_vents=U_vents.y,
  SC=SC.y) annotation (Placement(transformation(extent={{140,84},{160,104}})));
Flows.HeatAndVapourTransfer.Ventilation Q_ven_TopOut(
  A=surface.k,
  thermalScreen=true,
  u=u_wind.y,
  forcedVentilation=false,
  U_vents=U_vents.y,
  topAir=true,
  SC=SC.y)
  annotation (Placement(transformation(extent={{140,102},{160,122}})));
Flows.HeatAndVapourTransfer.AirThroughScreen Q_ven_AirTop(
  A=surface.k,
  K=0.2e-3,
  SC=SC.y,
  W=9.6) annotation (Placement(transformation(extent={{4,42},{-16,22}})));
Flows.HeatAndVapourTransfer.Convection_Evaporation Q_cnv_ScrTop(
  A=surface.k,
  SC=SC.y,
  MV_AirScr=Q_cnv_AirScr.MV_flow)
```

```
  annotation (Placement(transformation(extent={{-94,88},{-74,108}})));
Modelica.Thermal.HeatTransfer.Sensors.TemperatureSensor Tair_sensor
  annotation (Placement(transformation(extent={{136,-74},{146,-64}})));
ControlSystems.PID                              PID_Mdot(
  PVmin=18 + 273.15,
  PVmax=22 + 273.15,
  PVstart=0.5,
  CSstart=0.5,
  steadyStateInit=false,
  CSmin=0,
  Kp=0.7,
  Ti=600,
  CSmax=86.75)
        annotation (Placement(transformation(extent={{190,-108},{170,-88}})));
Modelica.Blocks.Sources.Constant Tsoil7(k=276.15)
  annotation (Placement(transformation(extent={{-54,-174},{-64,-164}})));
Flows.Sensors.RHSensor RH_out_sensor
  annotation (Placement(transformation(extent={{206,96},{218,108}})));
Modelica.Blocks.Sources.RealExpression Tout(y=TMY_and_control.y[2])
  annotation (Placement(transformation(extent={{222,112},{202,132}})));
Modelica.Blocks.Sources.RealExpression I_glob(y=TMY_and_control.y[5])
  annotation (Placement(transformation(extent={{-238,134},{-218,154}})));
Modelica.Blocks.Sources.RealExpression u_wind(y=TMY_and_control.y[6])
```

```
  annotation (Placement(transformation(extent={{222,128},{202,148}})));
Modelica.Blocks.Sources.RealExpression VPout(y=
    Greenhouses.Functions.WaterVapourPressure(
                                          TMY_and_control.y[2],
    TMY_and_control.y[3]))
  annotation (Placement(transformation(extent={{226,70},{202,90}})));
Modelica.Blocks.Sources.RealExpression OnOff(y=TMY_and_control.y[10])
  annotation (Placement(transformation(extent={{-220,32},{-200,52}})));

Modelica.Thermal.HeatTransfer.Celsius.PrescribedTemperature sky
  annotation (Placement(transformation(extent={{188,150},{176,162}})));
Modelica.Blocks.Sources.RealExpression Tsky(y=TMY_and_control.y[7])
  annotation (Placement(transformation(extent={{222,146},{202,166}})));
Modelica.Blocks.Sources.RealExpression Tair_setpoint(y=SP_new.y[2] +
    273.15)
  annotation (Placement(transformation(extent={{234,-50},{214,-30}})));
Components.CropYield.TomatoYieldModel TYM(
  T_canK=canopy.T,
  LAI(start=1.06),
  C_Leaf(start=40e3),
  C_Stem(start=30e3),
  CO2_air=CO2_air.CO2_ppm,
  R_PAR_can=solar_model.R_PAR_Can_umol + illu.R_PAR_Can_umol,
```

```
    LAI_MAX=2.7)
    annotation (Placement(transformation(extent={{90,-154},{130,-114}})));
  Flows.CO2MassTransfer.CO2_Air CO2_air(cap_CO2=h_Air.y)
    annotation (Placement(transformation(extent={{88,-10},{108,10}})));
  Flows.CO2MassTransfer.CO2_Air CO2_top(cap_CO2=0.4)
    annotation (Placement(transformation(extent={{88,58},{108,78}})));
  Flows.CO2MassTransfer.MC_ventilation2 MC_AirTop(f_vent=Q_ven_AirTop.f_AirTop)
    annotation (Placement(transformation(
        extent={{-10,-10},{10,10}},
        rotation=90,
        origin={98,28})));
  Flows.CO2MassTransfer.MC_ventilation2 MC_AirOut(f_vent=Q_ven_AirOut.f_vent_total)
    annotation (Placement(transformation(extent={{140,40},{160,60}})));
  Flows.CO2MassTransfer.MC_ventilation2 MC_TopOut(f_vent=Q_ven_TopOut.f_vent_total)
    annotation (Placement(transformation(extent={{140,66},{160,86}})));
  Flows.Sources.CO2.PrescribedConcentration CO2out
    annotation (Placement(transformation(extent={{188,46},{176,58}})));
  Modelica.Blocks.Sources.RealExpression CO2out_ppm_to_mgm3(y=340*1.94)
    annotation (Placement(transformation(extent={{222,42},{202,62}})));
  Flows.CO2MassTransfer.MC_AirCan MC_AirCan(MC_AirCan=TYM.MC_AirCan_mgCO2m
2s)
    annotation (Placement(transformation(extent={{88,-98},{108,-78}})));
  Flows.Sources.CO2.PrescribedCO2Flow MC_ExtAir(phi_ExtCO2=27)
```

```
  annotation (Placement(transformation(extent={{152,-10},{132,10}})));
ControlSystems.PID                         PID_CO2(
  PVstart=0.5,
  CSstart=0.5,
  steadyStateInit=false,
  PVmin=708.1,
  PVmax=1649,
  CSmin=0,
  CSmax=1,
  Kp=0.4,
  Ti=0.5)
  annotation (Placement(transformation(extent={{194,-6},{176,12}})));
Modelica.Blocks.Sources.RealExpression CO2_air_PV(y=CO2_air.CO2)
  annotation (Placement(transformation(extent={{234,-14},{214,6}})));
Flows.FluidFlow.Reservoirs.SourceMdot                sourceMdot_1ry(
  redeclare package Medium =
      Modelica.Media.Water.ConstantPropertyLiquidWater,
  Mdot_0=0.528,
  T_0=363.15)
  annotation (Placement(transformation(extent={{160,-114},{138,-92}})));
Flows.FluidFlow.Reservoirs.SinkP                sinkP_2ry(redeclare package
    Medium =        Modelica.Media.Water.ConstantPropertyLiquidWater, p0=
    1000000)
```

```
    annotation (Placement(transformation(extent={{-20,-20},{-10,-10}})));
  Modelica.Fluid.Sensors.Temperature T_ex_2ry(redeclare package Medium =
        Modelica.Media.Water.ConstantPropertyLiquidWater)
    annotation (Placement(transformation(extent={{-28,0},{-20,6}})));
  Modelica.Fluid.Sensors.Temperature T_su_1ry(redeclare package Medium =
        Modelica.Media.Water.ConstantPropertyLiquidWater)
    annotation (Placement(transformation(extent={{-40,-102},{-32,-96}})));
  Modelica.Fluid.Sensors.Temperature T_ex_1ry(redeclare package Medium =
        Modelica.Media.Water.ConstantPropertyLiquidWater)
    annotation (Placement(transformation(extent={{-68,-102},{-60,-96}})));
  Flows.Sensors.RHSensor RH_air_sensor
    annotation (Placement(transformation(extent={{136,-32},{148,-20}})));
  Modelica.Blocks.Sources.CombiTimeTable TMY_and_control(
    tableOnFile=true,
    tableName="tab",
    columns=1:10,
    fileName=Modelica.Utilities.Files.loadResource("modelica://Greenhouses/Resources/Data/10
Dec-22Nov.txt"))
    "Set-points for the climate"
    annotation (Placement(transformation(extent={{-152,152},{-132,172}})));

  Modelica.Blocks.Sources.RealExpression CO2_SP_var(y=SP_new.y[3]*1.94)
    annotation (Placement(transformation(extent={{234,2},{214,22}})));
```

```
  Modelica.Blocks.Sources.CombiTimeTable SC_usable(
    tableOnFile=true,
    tableName="tab",
    columns=1:2,
    fileName=Modelica.Utilities.Files.loadResource("modelica://Greenhouses/Resources/Data/SC
_usable_10Dec-22Nov.txt"))
    annotation (Placement(transformation(extent={{206,-62},{192,-48}})));
  ControlSystems.Climate.Control_ThScreen SC(R_Glob_can=I_glob.y,
      R_Glob_can_min=35)
    annotation (Placement(transformation(extent={{178,-48},{152,-28}})));
  Modelica.Blocks.Sources.RealExpression Tout_Kelvin(y=TMY_and_control.y[2] + 273.15)
    annotation (Placement(transformation(extent={{234,-34},{214,-14}})));
  ControlSystems.Climate.Uvents_RH_T_Mdot U_vents(
    T_air=air.T,
    T_air_sp=Tair_setpoint.y,
    Mdot=PID_Mdot.CS,
    RH_air_input=air.RH)
    annotation (Placement(transformation(extent={{80,108},{96,124}})));
  Modelica.Blocks.Sources.RealExpression h_Air(y=3.8 + (1 - SC.y)*0.4)
    "Height of main zone"
    annotation (Placement(transformation(extent={{-220,78},{-200,98}})));
  Modelica.Blocks.Sources.CombiTimeTable SP_new(
    tableOnFile=true,
```

```
    tableName="tab",
    columns=1:3,
    fileName=Modelica.Utilities.Files.loadResource("modelica://Greenhouses/Resources/Data/SP_10Dec-22Nov.txt"))
    "Climate set points 10Dec-22Nov: daily setpoints based on maximizing photosynthesis rate, minimum night temperature of 16, 24h mean temperature of 20"
    annotation (Placement(transformation(extent={{-118,152},{-98,172}})));
equation
  q_low = -pipe_low.flow1DimInc.Q_tot/surface.y;
  q_up = -pipe_up.flow1DimInc.Q_tot/surface.y;
  q_tot = -(pipe_low.flow1DimInc.Q_tot+pipe_up.flow1DimInc.Q_tot)/surface.y;
  max(q_tot,0) = der(E_th_tot_kWhm2*1e3*3600);
  E_th_tot = E_th_tot_kWhm2*surface.k;

  DM_Har = TYM.DM_Har;

  der(W_el_illu*1000*3600)=illu.W_el/surface.k;
  E_el_tot_kWhm2 = W_el_illu;
  E_el_tot = E_el_tot_kWhm2*surface.k;

  connect(canopy.heatPort, Q_rad_CanCov.port_a) annotation (Line(
      points={{-63,-72.36},{-36,-72.36},{-36,-72},{-4,-72}},
```

```
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(Q_rad_CanCov.port_b, cover.heatPort) annotation (Line(
    points={{16,-72},{36,-72},{36,-55.1777},{36,126}},
    color={191,0,0},
    smooth=Smooth.Bezier));

connect(canopy.heatPort, Q_cnv_CanAir.port_a) annotation (Line(
    points={{-63,-72.36},{-63,-72.36},{-30,-72.36},{-30,-52},{-4,-52}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_cnv_CanAir.port_b, air.heatPort) annotation (Line(
    points={{16,-52},{16,-52},{18,-52},{67.102,-52},{78,-52},{78,-44},{76.92,-44}},
    color={255,128,0},
    smooth=Smooth.Bezier));

connect(floor.heatPort, Q_cnv_FlrAir.port_a) annotation (Line(
    points={{-169,-155},{-169,-156},{50,-156}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_cnv_FlrAir.port_b, air.heatPort) annotation (Line(
    points={{70,-156},{78,-156},{78,-136.801},{78,-86},{78,-44},{76.92,-44}},
    color={255,128,0},
```

```
    smooth=Smooth.Bezier));
connect(cover.heatPort, Q_rad_CovSky.port_a) annotation (Line(
    points={{36,126},{36,126},{36,156},{76,156}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(canopy.heatPort, Q_rad_FlrCan.port_b) annotation (Line(
    points={{-63,-72.36},{-82,-72.36},{-82,-84},{-90,-84}},
    color={191,0,0},
    smooth=Smooth.Bezier));

connect(Q_rad_FlrCov.port_b, cover.heatPort) annotation (Line(
    points={{16,-146},{36,-146},{36,-131.871},{36,126}},
    color={191,0,0},
    smooth=Smooth.Bezier));

connect(floor.heatPort, Q_rad_FlrCan.port_a) annotation (Line(
    points={{-169,-155},{-169,-155},{-169,-91.2773},{-169,-84},{-155.575,-84},
        {-110,-84}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(floor.heatPort, Q_rad_FlrCov.port_a) annotation (Line(
    points={{-169,-155},{-113.937,-155},{-106,-155},{-106,-146},{-89.4648,-146},
        {-4,-146}},
```

```
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(MV_CanAir.port_b, air.massPort) annotation (Line(
    points={{16,-90},{16,-90},{67.3066,-90},{84,-90},{84,-68},{84,-44},{83.08,
        -44}},
    color={0,0,255},
    smooth=Smooth.Bezier));

connect(floor.heatPort, Q_cd_Soil.port_a) annotation (Line(
    points={{-169,-155},{-146,-155},{-146,-160},{-135.57,-160},{-116,-160},{-116,
        -158.4}},
    color={0,127,0},
    smooth=Smooth.Bezier));
connect(Q_rad_CanScr.port_a, canopy.heatPort) annotation (Line(
    points={{-90,-64},{-78,-64},{-78,-72.36},{-63,-72.36}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(Q_rad_CanScr.port_b, thScreen.heatPort) annotation (Line(
    points={{-110,-64},{-110,-64},{-122,-64},{-122,-38},{-122,66},{-122.6,66}},
    color={191,0,0},
    smooth=Smooth.Bezier));

connect(Q_rad_FlrScr.port_b, thScreen.heatPort) annotation (Line(
```

```
  points={{-136,-120},{-122,-120},{-122,-112.625},{-122,66},{-122.6,66}},
  color={191,0,0},
  smooth=Smooth.Bezier));
connect(thScreen.heatPort, Q_rad_ScrCov.port_a) annotation (Line(
  points={{-122.6,66},{-122,66},{-122,110},{-70,110},{-70,134},{-40,134},{-40,
    134}},
  color={191,0,0},
  smooth=Smooth.Bezier));
connect(Q_rad_ScrCov.port_b, cover.heatPort) annotation (Line(
  points={{-20,134},{-20,134},{2,134},{2,126},{10,126},{36,126}},
  color={191,0,0},
  smooth=Smooth.Bezier));

connect(floor.heatPort, Q_rad_FlrScr.port_a) annotation (Line(
  points={{-169,-155},{-169,-120},{-156,-120}},
  color={191,0,0},
  smooth=Smooth.Bezier));

connect(solar_model.R_SunCov_Glob, cover.R_SunCov_Glob) annotation (Line(
  points={{-176.9,146.5},{-22.8761,146.5},{8.8647,146.5},{26,146.5},{26,134},
    {26,132},{26.2,132},{26.2,131.6}},
  color={255,207,14},
  smooth=Smooth.Bezier));
```

```
connect(solar_model.R_SunFlr_Glob, floor.R_Flr_Glob[1]) annotation (Line(
    points={{-176.9,130},{-178,130},{-178,-130.064},{-178,-149.8},{-176.8,
        -149.8}},
    color={255,207,14},
    smooth=Smooth.Bezier));
connect(illu.R_IluFlr_Glob, floor.R_Flr_Glob[2]) annotation (Line(
    points={{-178,27},{-178,-130.011},{-178,-149.8},{-176.8,-149.8}},
    color={255,207,14},
    smooth=Smooth.Bezier));
connect(illu.R_IluAir_Glob, air.R_Air_Glob[2]) annotation (Line(
    points={{-166,27},{-166,27},{-166,10},{-156.015,10},{51.5469,10},{72,10},
        {72,-6},{72,-35.6},{73,-35.6}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(pipe_low.heatPorts, Q_rad_LowCov.heatPorts_a) annotation (Line(
    points={{-49,-119},{-30,-119},{-30,-134},{-3,-134}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(Q_rad_LowCan.heatPorts_a, pipe_low.heatPorts) annotation (Line(
    points={{-50,-107},{-50,-116.5},{-49,-116.5},{-49,-119}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(canopy.heatPort, Q_rad_LowCan.port_b) annotation (Line(
```

```
    points={{-63,-72.36},{-63,-72.18},{-50,-72.18},{-50,-88}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(Q_rad_LowFlr.heatPorts_a, pipe_low.heatPorts) annotation (Line(
    points={{-91,-134},{-72,-134},{-72,-119},{-49,-119}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(pipe_low.heatPorts, Q_cnv_LowAir.heatPorts_a) annotation (Line(
    points={{-49,-119},{-30,-119},{-30,-114},{-3,-114}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_cnv_LowAir.port_b, air.heatPort) annotation (Line(
    points={{16,-114},{16,-114},{62.3184,-114},{78,-114},{78,-76},{78,-44},{76.92,
        -44}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_rad_LowCov.port_b, cover.heatPort) annotation (Line(
    points={{16,-134},{36,-134},{36,-110.816},{36,126}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(pipe_low.heatPorts, Q_rad_LowScr.heatPorts_a) annotation (Line(
    points={{-49,-119},{-71.5,-119},{-71.5,-114},{-91,-114}},
    color={191,0,0},
```

```
    smooth=Smooth.Bezier));
connect(Q_rad_LowScr.port_b, thScreen.heatPort) annotation (Line(
    points={{-110,-114},{-122,-114},{-122,-98.531},{-122,66},{-122.6,66}},
    color={191,0,0},
    smooth=Smooth.Bezier));

connect(Q_rad_UpCov.port_b, cover.heatPort) annotation (Line(
    points={{16,-4},{36,-4},{36,0.90234},{36,126}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(pipe_up.heatPorts, Q_rad_UpCov.heatPorts_a) annotation (Line(
    points={{-43,-7},{-24,-7},{-24,-4},{-3,-4}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(floor.heatPort, Q_rad_UpFlr.port_b) annotation (Line(
    points={{-169,-155},{-169,-155},{-169,-49.8223},{-169,-24},{-145.031,-24},
        {-110,-24}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(pipe_up.heatPorts, Q_rad_UpCan.heatPorts_a) annotation (Line(
    points={{-43,-7},{-43,-25.65},{-42,-25.65},{-42,-39}},
    color={191,0,0},
    smooth=Smooth.Bezier));
```

```
connect(Q_rad_UpCan.port_b, canopy.heatPort) annotation (Line(
    points={{-42,-58},{-42,-58},{-42,-72},{-52,-72},{-52,-72.36},{-63,-72.36}},
    color={191,0,0},
    smooth=Smooth.Bezier));

connect(pipe_up.heatPorts, Q_cnv_UpAir.heatPorts_a) annotation (Line(
    points={{-43,-7},{-24,-7},{-24,-24},{-3,-24}},
    color={255,128,0},
    smooth=Smooth.Bezier));

connect(Q_cnv_UpAir.port_b, air.heatPort) annotation (Line(
    points={{16,-24},{16,-24},{38.998,-24},{52,-24},{52,-44},{64,-44},{76.92,-44}},
    color={255,128,0},
    smooth=Smooth.Bezier));

connect(Q_rad_UpScr.port_b, thScreen.heatPort) annotation (Line(
    points={{-110,-4},{-122,-4},{-122,0},{-122,66},{-122.6,66}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(solar_model.R_SunAir_Glob, air.R_Air_Glob[1]) annotation (Line(
    points={{-176.9,141},{-162,141},{-156,141},{-156,129.25},{-156,82},{-156,
        26.0195},{-156,14},{-142,14},{-146.408,14},{58,14},{72,14},{72,-12},{
        72,-35.6},{73,-35.6}},
```

```
    color={191,0,0},
    smooth=Smooth.Bezier));


connect(Q_cnv_AirScr.HeatPort_a, air.heatPort) annotation (Line(
    points={{-72,20},{-72,20},{52.2539,20},{78,20},{78,-4},{78,-44},{76.92,-44}},
    color={255,128,0},
    smooth=Smooth.Bezier));


connect(Q_cnv_AirScr.MassPort_a, air.massPort) annotation (Line(
    points={{-72,24},{-72,24},{66.7559,24},{84,24},{84,-4},{84,-44},{83.08,-44}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_cnv_AirScr.MassPort_b, thScreen.massPort) annotation (Line(
    points={{-92,24},{-92,24},{-116,24},{-116,66},{-115.4,66}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_cnv_AirScr.HeatPort_b, thScreen.heatPort) annotation (Line(
    points={{-92,20},{-122,20},{-122,66},{-122.6,66}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_cnv_AirCov.HeatPort_a, air.heatPort) annotation (Line(
    points={{70,66},{78,66},{78,51.6055},{78,-44},{76.92,-44}},
    color={255,128,0},
```

```
    smooth=Smooth.Bezier));
connect(Q_cnv_AirCov.HeatPort_b, cover.heatPort) annotation (Line(
    points={{50,66},{36,66},{36,126}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_cnv_AirCov.MassPort_b, cover.massPort) annotation (Line(
    points={{50,70},{42,70},{42,126},{41.6,126}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_cnv_AirCov.MassPort_a, air.massPort) annotation (Line(
    points={{70,70},{84,70},{84,56.6406},{84,-44},{83.08,-44}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(air_Top.heatPort, Q_cnv_TopCov.HeatPort_a) annotation (Line(
    points={{-44.3,100},{-44,100},{-44,120},{-14,120}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(air_Top.massPort, Q_cnv_TopCov.MassPort_a) annotation (Line(
    points={{-37.7,100},{-26,100},{-26,116},{-14,116}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_cnv_TopCov.MassPort_b, cover.massPort) annotation (Line(
    points={{6,116},{42,116},{42,126},{41.6,126}},
```

```
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_cnv_TopCov.HeatPort_b, cover.heatPort) annotation (Line(
    points={{6,120},{22,120},{36,120},{36,126}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_ven_AirOut.HeatPort_a, air.heatPort) annotation (Line(
    points={{140,96},{140,96},{87.928,96},{78,96},{78,83.8125},{78,-44},{76.92,
        -44}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_ven_AirOut.HeatPort_b, out.port) annotation (Line(
    points={{160,96},{168,96},{168,106},{168,122},{176,122}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_ven_AirOut.MassPort_b, prescribedVPout.port) annotation (Line(
    points={{160,92},{166,92},{166,80},{176,80}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_ven_AirOut.MassPort_a, air.massPort) annotation (Line(
    points={{140,92},{140,92},{84,92},{84,78.834},{84,-44},{83.08,-44}},
    color={0,0,255},
    smooth=Smooth.Bezier));
```

```
connect(Q_ven_TopOut.HeatPort_b, out.port) annotation (Line(
    points={{160,114},{168,114},{168,122},{176,122}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_ven_TopOut.MassPort_b, prescribedVPout.port) annotation (Line(
    points={{160,110},{166,110},{166,80},{176,80}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(air_Top.massPort, Q_ven_TopOut.MassPort_a) annotation (Line(
    points={{-37.7,100},{119.909,100},{128,100},{128,110},{140,110}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(air_Top.heatPort, Q_ven_TopOut.HeatPort_a) annotation (Line(
    points={{-44.3,100},{-44.3,104},{-19.7386,104},{110,104},{126,104},{126,114},
        {140,114}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_ven_AirTop.MassPort_a, air.massPort) annotation (Line(
    points={{4,34},{4,34},{72.8301,34},{84,34},{84,18.2324},{84,-44},{83.08,-44}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_ven_AirTop.MassPort_b, air_Top.massPort) annotation (Line(
```

```
    points={{-16,34},{-26,34},{-26,68},{-26,100},{-37.7,100}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_ven_AirTop.HeatPort_a, air.heatPort) annotation (Line(
    points={{4,30},{4,30},{69.4844,30},{78,30},{78,16.0527},{78,-44},{76.92,-44}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_ven_AirTop.HeatPort_b, air_Top.heatPort) annotation (Line(
    points={{-16,30},{-34.9766,30},{-44,30},{-44,42.6543},{-44,100},{-44.3,100}},
    color={255,128,0},
    smooth=Smooth.Bezier));

connect(canopy.massPort, MV_CanAir.port_a) annotation (Line(
    points={{-63,-76.84},{-63,-76.84},{-30,-76.84},{-30,-84},{-30,-90},{-4,-90}},
    color={0,0,255},
    smooth=Smooth.Bezier));

connect(Q_rad_UpFlr.heatPorts_a, pipe_up.heatPorts) annotation (Line(
    points={{-91,-24},{-70,-24},{-70,-7},{-43,-7}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(Q_rad_UpScr.heatPorts_a, pipe_up.heatPorts) annotation (Line(
    points={{-91,-4},{-70,-4},{-70,-7},{-43,-7}},
```

```
    color={191,0,0},
    smooth=Smooth.Bezier));
  connect(canopy.R_Can_Glob[2], illu.R_IluCan_Glob) annotation (Line(
    points={{-67.5,-50.8},{-67.5,-49.4},{-161.083,-49.4},{-172.75,-49.4},{
        -172.75,-27.0172},{-172.75,27},{-172,27}},
    color={255,207,14},
    smooth=Smooth.Bezier));
  connect(canopy.R_Can_Glob[1], solar_model.R_SunCan_Glob) annotation (Line(
    points={{-67.5,-50.8},{-67.5,-50},{-142.703,-50},{-160,-50},{-160,
        -31.2969},{-160,118},{-160,136},{-176.9,136},{-176.9,135.5}},
    color={255,207,14},
    smooth=Smooth.Bezier));
  connect(floor.heatPort, Q_rad_LowFlr.port_b) annotation (Line(
    points={{-169,-155},{-168,-155},{-168,-156},{-168,-146},{-168,-134},{-153.387,
        -134},{-110,-134}},
    color={191,0,0},
    smooth=Smooth.Bezier));
  connect(thScreen.heatPort, Q_cnv_ScrTop.HeatPort_a) annotation (Line(
    points={{-122.6,66},{-122,66},{-122,100},{-94,100}},
    color={255,128,0},
    smooth=Smooth.Bezier));
  connect(Q_cnv_ScrTop.HeatPort_b, air_Top.heatPort) annotation (Line(
    points={{-74,100},{-44.3,100}},
```

```
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(thScreen.massPort, Q_cnv_ScrTop.MassPort_a) annotation (Line(
    points={{-115.4,66},{-115.4,82},{-116,82},{-116,96},{-94,96}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(Q_cnv_ScrTop.MassPort_b, air_Top.massPort) annotation (Line(
    points={{-74,96},{-56,96},{-38,96},{-38,100},{-37.7,100}},
    color={0,0,255},
    smooth=Smooth.Bezier));
connect(air.heatPort, Tair_sensor.port) annotation (Line(
    points={{76.92,-44},{78,-44},{78,-69},{136,-69}},
    color={191,0,0},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));

connect(Tsoil7.y, Q_cd_Soil.T_layer_Nplus1) annotation (Line(
    points={{-64.5,-169},{-76.25,-169},{-76.25,-174},{-106,-174}},
    color={0,0,127},
    smooth=Smooth.Bezier));
connect(prescribedVPout.port, RH_out_sensor.massPort) annotation (Line(
    points={{176,80},{196,80},{196,99.6},{206,99.6}},
    color={0,0,255},
```

```
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(out.port, RH_out_sensor.heatPort) annotation (Line(
    points={{176,122},{194,122},{194,104.4},{206,104.4}},
    color={191,0,0},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(Tout.y, out.T) annotation (Line(
    points={{201,122},{201,122},{189.2,122}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(I_glob.y, solar_model.I_glob) annotation (Line(
    points={{-217,144},{-210,144},{-210,141},{-202.2,141}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));

connect(VPout.y, prescribedVPout.VP) annotation (Line(
    points={{200.8,80},{200.8,80},{189.2,80}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
```

```
connect(OnOff.y, illu.switch) annotation (Line(
    points={{-199,42},{-192,42},{-192,40.4},{-174.2,40.4}},
    color={0,0,127},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(cover.heatPort, Q_cnv_CovOut.port_a) annotation (Line(
    points={{36,126},{36,126},{36,140},{76,140}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Q_cnv_CovOut.port_b, out.port) annotation (Line(
    points={{96,140},{136,140},{136,122},{176,122}},
    color={255,128,0},
    smooth=Smooth.Bezier));
connect(Tsky.y, sky.T) annotation (Line(
    points={{201,156},{189.2,156}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(sky.port, Q_rad_CovSky.port_b) annotation (Line(
    points={{176,156},{138,156},{96,156}},
    color={191,0,0},
    smooth=Smooth.Bezier));
connect(Tair_sensor.T, PID_Mdot.PV) annotation (Line(
```

```
    points={{146,-69},{146,-69},{206,-69},{206,-102},{190,-102}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(Tair_setpoint.y, PID_Mdot.SP) annotation (Line(
    points={{213,-40},{213,-94},{190,-94}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(CO2out_ppm_to_mgm3.y,CO2out. CO2) annotation (Line(
    points={{201,52},{206,52},{189.2,52}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(CO2_air.port, MC_AirOut.port_a) annotation (Line(
    points={{98,0},{112,0},{112,13.166},{112,42},{112,68.793},{112,80},{120,80},
        {132,80},{132,70},{132,50},{140,50}},
    color={95,95,95},
    smooth=Smooth.Bezier));
connect(MC_AirOut.port_b, CO2out.port) annotation (Line(
    points={{160,50},{176,50},{176,52}},
    color={95,95,95},
    smooth=Smooth.Bezier));
```

```
connect(CO2_air.port, MC_AirTop.port_a) annotation (Line(
    points={{98,0},{98,18}},
    color={95,95,95},
    smooth=Smooth.Bezier));
connect(MC_AirTop.port_b, CO2_top.port) annotation (Line(
    points={{98,38},{98,68}},
    color={95,95,95},
    smooth=Smooth.Bezier));
connect(CO2_top.port, MC_TopOut.port_a) annotation (Line(
    points={{98,68},{100,68},{100,88},{117.598,88},{134,88},{134,76},{140,76}},
    color={95,95,95},
    smooth=Smooth.Bezier));
connect(MC_TopOut.port_b, CO2out.port) annotation (Line(
    points={{160,76},{160,72},{170,72},{170,52},{176,52}},
    color={95,95,95},
    smooth=Smooth.Bezier));
connect(CO2_air.port, MC_AirCan.port) annotation (Line(
    points={{98,0},{98,-79}},
    color={95,95,95},
    smooth=Smooth.Bezier));
connect(MC_ExtAir.port, CO2_air.port) annotation (Line(
    points={{132,0},{98,0}},
    color={95,95,95},
```

```
    smooth=Smooth.None));
connect(CO2_air_PV.y, PID_CO2.PV) annotation (Line(
    points={{213,-4},{204,-4},{204,0},{196,0},{194,0},{194,-0.6}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));

connect(sourceMdot_1ry.flangeB, pipe_low.pipe_in) annotation (Line(
    points={{139.1,-103},{139.1,-104},{114,-104},{-15.4707,-104},{-37,-104},{-37,
        -125}},
    color={0,128,255},
    smooth=Smooth.Bezier));
connect(PID_Mdot.CS, sourceMdot_1ry.in_Mdot) annotation (Line(
    points={{169.4,-98},{152,-98},{152,-96.4},{155.6,-96.4}},
    color={0,0,127},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(pipe_up.pipe_out, sinkP_2ry.flangeB) annotation (Line(
    points={{-31,-13},{-25.5,-13},{-25.5,-15},{-19.2,-15}},
    color={0,0,255},
    smooth=Smooth.None));
connect(T_ex_2ry.port, sinkP_2ry.flangeB) annotation (Line(
    points={{-24,0},{-24,-16},{-19.2,-15}},
```

```
    color={0,127,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(T_su_1ry.port, pipe_low.pipe_in) annotation (Line(
    points={{-36,-102},{-36,-116},{-37,-116},{-37,-125}},
    color={0,127,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(T_ex_1ry.port, pipe_low.pipe_out) annotation (Line(
    points={{-64,-102},{-64,-125},{-61,-125}},
    color={0,127,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(pipe_low.pipe_out, pipe_up.pipe_in) annotation (Line(
    points={{-61,-125},{-61,-124},{-78,-124},{-78,-86.8398},{-78,-38},{-64,-38},
        {-64,-14},{-56,-14},{-56,-13},{-55,-13}},
    color={0,128,255},
    smooth=Smooth.Bezier));

connect(air.heatPort, RH_air_sensor.heatPort) annotation (Line(
    points={{76.92,-44},{90,-44},{90,-23.6},{136,-23.6}},
    color={191,0,0},
    smooth=Smooth.Bezier,
```

```
    pattern=LinePattern.Dash));
connect(air.massPort, RH_air_sensor.massPort) annotation (Line(
    points={{83.08,-44},{91.54,-44},{91.54,-28.4},{136,-28.4}},
    color={0,0,255},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(CO2_SP_var.y, PID_CO2.SP) annotation (Line(
    points={{213,12},{202,12},{202,6.6},{194,6.6}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(SC_usable.y[2], SC.SC_usable) annotation (Line(
    points={{191.3,-55},{186,-55},{186,-48},{182,-48},{182,-47},{179.3,-47}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(Tair_setpoint.y, SC.T_air_sp) annotation (Line(
    points={{213,-40},{204,-40},{180,-40},{179.3,-40},{179.3,-41}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(Tout_Kelvin.y, SC.T_out) annotation (Line(
    points={{213,-24},{213,-24},{200,-24},{200,-34.8},{179.3,-34.8}},
```

```
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(RH_air_sensor.RH, SC.RH_air) annotation (Line(
    points={{148,-26},{148,-26},{186,-26},{186,-29},{179.3,-29}},
    color={0,0,127},
    smooth=Smooth.Bezier,
    pattern=LinePattern.Dash));
connect(PID_CO2.CS, MC_ExtAir.U_MCext) annotation (Line(
    points={{175.46,3},{169.73,3},{169.73,0.2},{154.2,0.2}},
    color={0,0,127},
    smooth=Smooth.Bezier));
annotation (
    Diagram(coordinateSystem(preserveAspectRatio=false, extent={{-240,-180},{240,
          180}}), graphics={
      Polygon(
        points={{-190,-158},{130,-158},{130,66},{-30,166},{-190,66},{-190,-158}},
        lineColor={135,135,135},
        smooth=Smooth.None,
        lineThickness=1),
      Line(
        points={{-70,124}},
        color={135,135,135},
```

```
  thickness=1,
  smooth=Smooth.None),
Line(
  points={{50,116},{114,76}},
  color={255,255,255},
  thickness=1,
  smooth=Smooth.None),
Line(
  points={{50,116},{122,116}},
  color={135,135,135},
  thickness=1,
  smooth=Smooth.None),
Line(
  points={{146,-138},{166,-138}},
  color={255,128,0},
  smooth=Smooth.Bezier),
Line(
  points={{146,-146},{166,-146}},
  color={191,0,0},
  smooth=Smooth.Bezier),
Line(
  points={{146,-154},{166,-154}},
  color={255,207,14},
```

```
          smooth=Smooth.Bezier),
        Text(
          extent={{170,-136},{230,-172}},
          lineColor={0,0,0},
          fillColor={0,0,255},
          fillPattern=FillPattern.Solid,
          horizontalAlignment=TextAlignment.Left,
          textString="Convection
Long-wave radiation
Short-wave radiation
Conduction
Vapour transfer"),
        Line(
          points={{146,-162},{166,-162}},
          color={0,127,0},
          smooth=Smooth.Bezier),
        Line(
          points={{146,-170},{166,-170}},
          color={0,0,255},
          smooth=Smooth.Bezier),
        Line(
          points={{-190,52},{130,52}},
          color={135,135,135},
```

```
  smooth=Smooth.None,
  thickness=0.5),
Line(
  points={{-190,44},{130,44}},
  color={135,135,135},
  smooth=Smooth.None,
  thickness=0.5),
Line(
  points={{-190,52},{-180,44},{-170,52},{-160,44},{-150,52},{-140,44},{-130,
      52},{-120,44},{-110,52},{-100,44},{-90,52},{-80,44},{-70,52},{-60,
      44},{-50,52},{-40,44},{-30,52},{-20,44},{-10,52},{0,44},{10,52},{20,
      44},{30,52},{40,44},{50,52},{60,44},{70,52},{80,44},{90,52},{100,44},
      {110,52},{120,44},{130,52}},
  color={135,135,135},
  thickness=0.5,
  smooth=Smooth.None),
Line(
  points={{-190,62},{-186,66},{-182,58},{-178,66},{-174,58},{-170,66},{-166,
      58},{-162,66},{-158,58},{-154,66},{-150,58},{-146,66},{-142,58},{-138,
      66},{-134,58},{-130,66},{-126,58},{-122,66},{-118,58},{-114,66},{-110,
      58},{-106,66},{-102,58},{-98,66},{-94,58},{-90,66},{-86,58},{-82,66},
      {-78,58},{-74,66},{-70,58},{-66,66},{-62,58},{-58,66},{-54,58},{-50,
      66},{-46,58},{-42,66},{-38,58},{-34,66},{-30,58},{-26,66},{-22,58},
```

```
       {-18,66},{-14,58},{-10,66},{-6,58},{-2,66},{2,58},{6,66},{10,58},{
       14,66},{18,58},{22,66},{26,58},{28,62},{32,62}},
     color={95,95,95},
     thickness=0.5,
     smooth=Smooth.Bezier),
    Rectangle(
     extent={{32,64},{34,60}},
     lineColor={95,95,95},
     lineThickness=0.5,
     fillColor={215,215,215},
     fillPattern=FillPattern.Solid)}),
                 Icon(coordinateSystem(preserveAspectRatio=false,
     extent={{-100,-100},{100,100}})),
  Documentation(info="<html>
<p><b></font><font style=\"font-size: 12pt; \">Simulation of greenhouse climate</b></p>
<p></font><font style=\"font-
size: 10pt; \">This example intends to illustrate the simulation of a greenhouse climate. The gree
nhouse is built by interconnecting all of the energy and mass <a href=\"modelica://Greenhouses.
Flows\">Flows</a> presents in a greenhouse to their related <a href=\"modelica://Greenhouses.
Components.Greenhouse\">Components</a>. As it can be distinguished, the greenhouse modele
d in this example consists of two levels of heating circuits, roof windows (but not side vents), nat
ural ventilation (no forced ventilation) and a movable thermal screen. It should be noted that, wh
en the screen is drawn, the air of the greenhouse is divided in two zones, i.e. below and above the
```

screen. These zones are modeled separately (models air and air_Top) and their climate is assumed to be homogeneous. The models parameters have been set to typical values for Venlo-type greenhouse construction design dedicated to tomato crop cultivation. The greenhouse floor area and the mean greenhouse height are set in two individual block sources.</p>

<p><big>The simulated greenhouse is located in Belgium and the simulation period is from December 10th to November 22nd. Two data files are required: </p>

<big><i>Weather data</i>: The input weather data for the simulation period is extracted from a TMY for Brussels and can be found in ‘Greenhouses/Resources/Data/10Dec-22Nov.txt’. The file contains data for the outside air temperature, air pressure, wind speed and global irradiation. The sky temperature, previously computed in a Python script, is also included in this file.

<big><i>Climate control set-points</i>: The temperature and CO2 set-points for the simulation period are calculated according to the strategy presented in the online documentation and can be found in ‘Greenhouses/Resources/Data/SP_10Dec-22Nov.txt’.

<p><big> These '.txt' files are accessed by means of <i>TMY_and_control</i> and <i>SP_new</i>, which are two CombiTimeTables models from the Modelica Standard Library.</p>

<p><big>The goal of this example is to show the energy flows interacting in a greenhouse. Thus, no generation units are included. Instead, the heating pipes are connected to a water source and sink model. The model includes the following controls:</p>

```
<ul>
<li><big><i>PID_Mdot</i>: A PI controller adjusts the output mass flow rate of the water sourc
e connected to the heating pipes by comparing the air temperature set-
point and present value.</li>
<li><big><i>PID_CO2</i>: A PI controller adjusts the output of the CO2 external source by co
mparing the actual CO2 concentration of the air to its set-point.</li>
<li><big><i>Ctrl_SC</i>: A state graph adjusts the screen closure (SC) according to the strateg
y presented in <a href=\"modelica://Greenhouses.ControlSystems.Climate.Control_ThScreen\">
Control_ThScreen</a>. The real inputs must be connected to the air relative humidity, the outdo
or temperature, the indoor air temperature set-
point and the usable hours of the screen. The usable hours are 1h30 before dusk, 1h30 after dawn
 and during night.</li>
<li><big><i>vents</i>: A PI controller adjusts the opening of the windows according to the strat
egy presented in <a href=\"modelica://Greenhouses.ControlSystems.Climate.Uvents_RH_T_Md
ot\">Uvents_RH_T_Mdot</a>. The opening depends mainly on the indoor air relative humidity
and temperature.</li>
<li><big><i>OnOff</i>: controls the ON/OFF operation of the supplementary lighting accordin
g to the strategy presented in Control Systems. The control output, previously computed in a Pyt
hon script, is input as a .txt file by means of the TMY_and_control CombiTimeTable.</li>
</ul>
</html>"));
end Greenhouse_1;
```

“GlobalSystem_2.mo”

```
within Greenhouses.Examples;
model GlobalSystem_2
  "Greenhouse connected to a CHP, a heat pump and thermal energy storage"
  extends Modelica.Icons.Example;
  Real Mdot_2ry(unit="kg/s",start=0.528);
  Real Mdot_air(unit="kg/s",start=0.528);
  Real E_gas_CHP(unit="kW.h");
  Real E_el_CHP(unit="kW.h");
  Real E_el_HP(unit="kW.h");
  Real E_th_CHP(unit="kW.h");
  Real E_th_HP(unit="kW.h");
  Real E_th_total(unit="kW.h");
  Real E_th_G(unit="kW.h");
  Real E_amb_TES(unit="kW.h");
  Real E_el_sell(unit="kW.h");
  Real E_el_buy(unit="kW.h");
  Real Pi_buy(unit="1/(kW.h)")=0.1415 "50euro/MWh";
  Real Pi_sell(unit="1/(kW.h)")=0.0472;
  Real Pi_gas(unit="1/(kW.h)")=0.0355;
  Real C_sell;
```

```
Real C_buy;
Real C_gas;
Real E_gas_CHP_kWhm2(unit="kW.h/m2");
Real E_el_CHP_kWhm2(unit="kW.h/m2");
Real E_th_CHP_kWhm2(unit="kW.h/m2");
Real E_th_HP_kWhm2(unit="kW.h/m2");
Real E_el_HP_kWhm2(unit="kW.h/m2");
Real E_th_total_kWhm2(unit="kW.h/m2");
Real E_th_G_kWhm2(unit="kW.h/m2");
Real E_amb_TES_kWhm2(unit="kW.h/m2");
Real E_el_sell_kWhm2(unit="kW.h/m2");
Real E_el_buy_kWhm2(unit="kW.h/m2");
Real W_CHP_net(unit="W");
Real W_sell(unit="W");
Real W_buy(unit="W");
Real W_residual(unit="W");
Greenhouses.Components.HVAC.CHP CHP(
  redeclare package Medium = Modelica.Media.Water.ConstantPropertyLiquidWater,
  Tmax=373.15,
  Th_nom=773.15)
  annotation (Placement(transformation(extent={{-20,-20},{10,10}})));
Modelica.Fluid.Sensors.Temperature T_ex_CHP(redeclare package Medium =
      Modelica.Media.Water.ConstantPropertyLiquidWater)
```

```
  annotation (Placement(transformation(extent={{-32,2},{-24,8}})));
Modelica.Fluid.Sensors.Temperature T_su_CHP(redeclare package Medium =
      Modelica.Media.Water.ConstantPropertyLiquidWater)
  annotation (Placement(transformation(extent={{-36,-14},{-28,-8}})));
Modelica.Thermal.HeatTransfer.Sources.PrescribedHeatFlow Qdot_nom_gas_CHP
  annotation (Placement(transformation(extent={{26,-4},{18,4}})));
Modelica.Blocks.Sources.Constant set_Qdot_nom_gas_CHP(k=1750e3)
  annotation (Placement(transformation(extent={{40,-2},{36,2}})));
Greenhouses.Components.HVAC.HeatStorageWaterHeater.Heat_storage_hx_R TES(
  h_T=0.6,
  U_amb=2,
  steadystate_hx=false,
  Unom_hx=1000,
  steadystate_tank=false,
  redeclare package MainFluid =
      Modelica.Media.Water.ConstantPropertyLiquidWater,
  h2=1,
  h1=0.01,
  N1=1,
  N2=15,
  Wdot_res=115500,
  redeclare package SecondaryFluid =
      Modelica.Media.Water.ConstantPropertyLiquidWater,
```

```
  V_hx=0.005*10,
  A_hx=700,
  Mdot_nom=5,
  V_tank=313,
  Tmax=373.15,
  Tstart_inlet_tank=303.15,
  Tstart_outlet_tank=323.15,
  Tstart_inlet_hx=333.15,
  Tstart_outlet_hx=313.15)
  annotation (Placement(transformation(extent={{-10,26},{-40,56}})));
Modelica.Fluid.Sensors.Temperature T_ex_TES(redeclare package Medium =
     Modelica.Media.Water.ConstantPropertyLiquidWater)
  annotation (Placement(transformation(extent={{-44,30},{-52,36}})));
Modelica.Fluid.Sensors.Temperature T_su_G(redeclare package Medium =
     Modelica.Media.Water.ConstantPropertyLiquidWater)
  annotation (Placement(transformation(extent={{-22,66},{-14,72}})));
Modelica.Fluid.Sensors.Temperature T_ex_G(redeclare package Medium =
     Modelica.Media.Water.ConstantPropertyLiquidWater)
  annotation (Placement(transformation(extent={{-16,34},{-8,40}})));
Greenhouses.Flows.FluidFlow.Pump_Mdot pump_2ry(redeclare package Medium =
     Modelica.Media.Water.ConstantPropertyLiquidWater)
  annotation (Placement(transformation(extent={{-60,-64},{-48,-52}})));
Greenhouses.Flows.FluidFlow.Reservoirs.SinkP sinkP_2ry(redeclare package
```

```
    Medium = Modelica.Media.Water.ConstantPropertyLiquidWater, p0=1000000)
  annotation (Placement(transformation(extent={{-76,-48},{-88,-36}})));
Greenhouses.Flows.FluidFlow.Pdrop pdrop_2ry(
  Mdot_max=0.5,
  redeclare package Medium = Modelica.Media.Water.ConstantPropertyLiquidWater,
  DELTAp_max=1100)
  annotation (Placement(transformation(extent={{-88,-64},{-76,-52}})));
Greenhouses.Components.Greenhouse.Unit.Greenhouse G
  annotation (Placement(transformation(extent={{46,26},{94,62}})));
ControlSystems.HVAC.Control_2 controller(
  Mdot_max=86,
  T_max=343.15,
  T_min=313.15,
  Mdot_1ry=pump_1ry.flow_in)
  annotation (Placement(transformation(extent={{-74,-2},{-54,18}})));
Greenhouses.Flows.FluidFlow.Pump_Mdot pump_1ry(redeclare package Medium =
      Modelica.Media.Water.ConstantPropertyLiquidWater, Mdot_0=0.528)
  annotation (Placement(transformation(extent={{24,44},{36,56}})));
Greenhouses.Flows.FluidFlow.Pdrop pdrop_1ry(
  redeclare package Medium = Modelica.Media.Water.ConstantPropertyLiquidWater,
  Mdot_max=0.5,
  DELTAp_max=1100)
  annotation (Placement(transformation(extent={{-6,48},{6,60}})));
```

```
Greenhouses.Flows.FluidFlow.Reservoirs.SinkP sinkP_1ry(redeclare package
    Medium = Modelica.Media.Water.ConstantPropertyLiquidWater, p0=1000000)
  annotation (Placement(transformation(extent={{6,62},{-6,74}})));
Modelica.Blocks.Sources.RealExpression set_Mdot_2ry(y=Mdot_2ry)
  annotation (Placement(transformation(extent={{-88,-36},{-74,-24}})));
Modelica.Fluid.Sensors.Temperature T_su_HP(redeclare package Medium =
      Modelica.Media.Water.ConstantPropertyLiquidWater)
  annotation (Placement(transformation(extent={{-36,-52},{-28,-46}})));
Modelica.Blocks.Sources.RealExpression T_out(y=BruTMY.y[2] + 273.15)
  annotation (Placement(transformation(extent={{32,-32},{22,-22}})));
Greenhouses.Components.HVAC.HeatPump_ConsoClim HP(
  COP_n=3.5,
  Q_dot_cd_n=490e3,
  redeclare package Medium1 =
      Modelica.Media.Water.ConstantPropertyLiquidWater,
  redeclare package Medium2 = Modelica.Media.Air.SimpleAir,
  T_su_ev_n=280.15,
  T_ex_cd_n=308.15)
  annotation (Placement(transformation(extent={{0,-56},{-20,-36}})));
Greenhouses.Flows.FluidFlow.Reservoirs.SinkP sinkP_air(redeclare package
    Medium = Modelica.Media.Air.SimpleAir, p0=100000)
  annotation (Placement(transformation(extent={{14,-58},{26,-46}})));
Greenhouses.Flows.FluidFlow.Reservoirs.SourceMdot sourceMdot(redeclare
```

```
      package Medium = Modelica.Media.Air.SimpleAir, Mdot_0=1)
    annotation (Placement(transformation(extent={{26,-44},{14,-32}})));
  Modelica.Blocks.Sources.RealExpression set_Mdot_air(y=Mdot_air)
    annotation (Placement(transformation(extent={{42,-40},{28,-28}})));
  Modelica.Blocks.Sources.CombiTimeTable BruTMY(
    tableOnFile=true,
    tableName="tab",
    columns=1:10,
    fileName=Modelica.Utilities.Files.loadResource(
        "modelica://Greenhouses/Resources/Data/10Dec-22Nov.txt"))
    "TMY of Brussels for the period of 10Dec to 22Nov"
    annotation (Placement(transformation(extent={{-76,80},{-62,94}})));
equation
  Mdot_2ry = if time<1e4 then 10 else (if controller.CHP then 5 else 0);
  Mdot_air = if time<1e4 then 1 else (if controller.CHP then 1 else 0);
  der(E_gas_CHP*1e3*3600) = CHP.Qdot_gas;
  der(E_el_CHP*1e3*3600) = CHP.Wdot_el;
  der(E_th_CHP*1e3*3600) = CHP.prescribedHeatFlow.Q_flow;
  der(E_th_HP*1e3*3600) = HP.Q_dot_cd;
  der(E_el_HP*1e3*3600) = HP.W_dot_cp;
  E_th_total = E_th_CHP + E_th_HP;
  E_th_G = G.E_th_tot;
  der(E_amb_TES*1e3*3600) = sum(TES.cell1DimInc_hx.Q_tot);
```

```
W_CHP_net = CHP.Wdot_el - HP.W_dot_cp;
W_sell = max(0,W_CHP_net-G.illu.W_el);
W_buy = max(0,G.illu.W_el-W_CHP_net);
W_residual = G.illu.W_el-W_CHP_net
  "Positive: residual load (buy), negative: too much (sell)";
der(E_el_sell*1e3*3600) = max(0,W_CHP_net-G.illu.W_el);
der(E_el_buy*1e3*3600) = max(0,G.illu.W_el-W_CHP_net);
C_sell = Pi_sell*E_el_sell;
C_buy = Pi_buy*E_el_buy;
C_gas = Pi_gas*E_gas_CHP;
E_gas_CHP_kWhm2=E_gas_CHP/G.surface.k;
E_el_CHP_kWhm2=E_el_CHP/G.surface.k;
E_el_HP_kWhm2=E_el_HP/G.surface.k;
E_th_CHP_kWhm2=E_th_CHP/G.surface.k;
E_th_HP_kWhm2=E_th_HP/G.surface.k;
E_th_total_kWhm2=E_th_total/G.surface.k;
E_th_G_kWhm2=E_th_G/G.surface.k;
E_amb_TES_kWhm2=E_amb_TES/G.surface.k;
E_el_sell_kWhm2=E_el_sell/G.surface.k;
E_el_buy_kWhm2=E_el_buy/G.surface.k;
connect(T_ex_CHP.port, CHP.OutFlow) annotation (Line(
    points={{-28,2},{-28,-2.6},{-20,-2.6}},
    color={0,127,255},
```

```
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(T_su_CHP.port, CHP.InFlow) annotation (Line(
    points={{-32,-14},{-26,-14},{-26,-17.3},{-20,-17.3}},
    color={0,127,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(Qdot_nom_gas_CHP.port, CHP.HeatSource) annotation (Line(
    points={{18,0},{18,1.15},{-0.05,1.15}},
    color={191,0,0},
    smooth=Smooth.None));
connect(set_Qdot_nom_gas_CHP.y, Qdot_nom_gas_CHP.Q_flow) annotation (Line(
    points={{35.8,0},{26,0}},
    color={0,0,127},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(T_su_G.port, TES.MainFluid_ex) annotation (Line(
    points={{-18,66},{-18,64},{-25.2,64},{-25.2,53.9},{-25,53.9}},
    color={0,127,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(T_ex_G.port, TES.MainFluid_su) annotation (Line(
    points={{-12,34},{-12,29.15},{-19.45,29.15}},
```

```
    color={0,127,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(CHP.OutFlow, TES.SecondaryFluid_ex) annotation (Line(
    points={{-20,-2.6},{-36,-2.6},{-36,36},{-31,36},{-31,36.5}},
    color={0,0,255},
    smooth=Smooth.None,
    thickness=0.5));
connect(pdrop_2ry.OutFlow, pump_2ry.inlet) annotation (Line(
    points={{-77.5,-58},{-74,-58},{-74,-57.7},{-58.32,-57.7}},
    color={0,0,255},
    smooth=Smooth.None,
    thickness=0.5));
connect(sinkP_2ry.flangeB, pump_2ry.inlet) annotation (Line(
    points={{-76.96,-42},{-68,-42},{-68,-57.7},{-58.32,-57.7}},
    color={0,0,255},
    smooth=Smooth.None));
connect(T_ex_TES.port, TES.SecondaryFluid_su) annotation (Line(
    points={{-48,30},{-42,30},{-42,46.1},{-31,46.1}},
    color={0,127,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(pdrop_2ry.InFlow, TES.SecondaryFluid_su) annotation (Line(
```

```
    points={{-86.5,-58},{-94,-58},{-94,46},{-31,46},{-31,46.1}},
    color={0,0,255},
    smooth=Smooth.None,
    thickness=0.5));
connect(TES.Temperature, controller.T_tank) annotation (Line(
    points={{-18.85,45.35},{-22,45.35},{-22,42},{-82,42},{-82,4},{-75,4}},
    color={0,0,127},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(controller.CHP, CHP.on_off) annotation (Line(
    points={{-53.5,14},{-42,14},{-42,-22},{-4,-22},{-4,-18.8},{-4.7,-18.8}},
    color={255,0,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(pdrop_1ry.OutFlow, pump_1ry.inlet) annotation (Line(
    points={{4.5,54},{12,54},{12,50.3},{25.68,50.3}},
    color={0,0,255},
    smooth=Smooth.None,
    thickness=0.5));
connect(sinkP_1ry.flangeB, pump_1ry.inlet) annotation (Line(
    points={{5.04,68},{12,68},{12,50.3},{25.68,50.3}},
    color={0,0,255},
    smooth=Smooth.None));
```

```
connect(G.flangeB, TES.MainFluid_su) annotation (Line(
  points={{53.4,29.2},{20.5,29.2},{20.5,29.15},{-19.45,29.15}},
  color={0,0,255},
  smooth=Smooth.None,
  thickness=0.5));
connect(pump_1ry.outlet, G.flangeA) annotation (Line(
  points={{33.36,54.44},{33.36,53.7},{53.4,53.7},{53.4,39}},
  color={0,0,255},
  smooth=Smooth.None,
  thickness=0.5));
connect(pdrop_1ry.InFlow, TES.MainFluid_ex) annotation (Line(
  points={{-4.5,54},{-8,54},{-8,53.9},{-25,53.9}},
  color={0,0,255},
  smooth=Smooth.None,
  thickness=0.5));
connect(G.PID_Mdot_CS, pump_1ry.flow_in) annotation (Line(
  points={{86.6,34.6},{96,34.6},{96,68},{28,68},{28,54},{28.08,54},{28.08,
    54.8}},
  color={0,0,127},
  smooth=Smooth.None,
  pattern=LinePattern.Dash));
connect(T_ex_TES.T, controller.T_low_TES) annotation (Line(
  points={{-50.8,33},{-50.8,32},{-80,32},{-80,8},{-75,8}},
```

```
    color={0,0,127},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(set_Mdot_2ry.y, pump_2ry.flow_in) annotation (Line(
    points={{-73.3,-30},{-56,-30},{-56,-53.2},{-55.92,-53.2}},
    color={0,0,127},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(pump_2ry.outlet, HP.Supply_cd) annotation (Line(
    points={{-50.64,-53.56},{-34.32,-53.56},{-34.32,-53},{-19,-53}},
    color={0,0,255},
    smooth=Smooth.None,
    thickness=0.5));
connect(T_su_HP.port, HP.Supply_cd) annotation (Line(
    points={{-32,-52},{-26,-52},{-26,-53},{-19,-53}},
    color={0,127,255},
    smooth=Smooth.None,
    pattern=LinePattern.Dash));
connect(HP.Exhaust_cd, CHP.InFlow) annotation (Line(
    points={{-19,-39},{-19,-28.5},{-20,-28.5},{-20,-17.3}},
    color={0,0,255},
    smooth=Smooth.None,
    thickness=0.5));
```

```
connect(sourceMdot.flangeB, HP.Supply_ev) annotation (Line(
   points={{14.6,-38},{6,-38},{6,-39},{-1,-39}},
   color={135,135,135},
   smooth=Smooth.None));
connect(T_out.y, sourceMdot.in_T) annotation (Line(
   points={{21.5,-27},{20.75,-27},{20.75,-34.4},{20.12,-34.4}},
   color={0,0,127},
   smooth=Smooth.None));
connect(HP.Exhaust_ev, sinkP_air.flangeB) annotation (Line(
   points={{-1,-53},{6.5,-53},{6.5,-52},{14.96,-52}},
   color={135,135,135},
   smooth=Smooth.None));
connect(CHP.Wdot_el, HP.W_dot_set) annotation (Line(
   points={{11.5,-12.5},{11.5,-12},{44,-12},{44,-64},{-10,-64},{-10,-57}},
   color={255,128,0},
   smooth=Smooth.None,
   thickness=0.5));
connect(T_ex_CHP.T, controller.T_su_hx) annotation (Line(
   points={{-25.2,5},{-22,5},{-22,20},{-78,20},{-78,12},{-75,12}},
   color={0,0,127},
   smooth=Smooth.None,
   pattern=LinePattern.Dash));
connect(set_Mdot_air.y, sourceMdot.in_Mdot) annotation (Line(
```

```
   points={{27.3,-34},{26,-34},{26,-34.4},{23.6,-34.4}},
   color={0,0,127},
   smooth=Smooth.None));
 connect(controller.CHP, HP.on_off) annotation (Line(
   points={{-53.5,14},{-42,14},{-42,-56.6},{-14,-56.6}},
   color={255,0,255},
   smooth=Smooth.None,
   pattern=LinePattern.Dash));
 annotation (Diagram(coordinateSystem(preserveAspectRatio=false, extent={{-100,
      -100},{100,100}}), graphics={
    Text(
     extent={{-64,60},{-28,50}},
     lineColor={0,0,0},
     pattern=LinePattern.Dash,
     fillColor={215,215,215},
     fillPattern=FillPattern.Solid,
     textString="Thermal Energy
Storage"),
    Text(
     extent={{56,26},{82,20}},
     lineColor={0,0,0},
     pattern=LinePattern.Dash,
     fillColor={215,215,215},
```

```
  fillPattern=FillPattern.Solid,
  textString="Greenhouse"),
Text(
  extent={{-18,-22},{8,-28}},
  lineColor={0,0,0},
  pattern=LinePattern.Dash,
  fillColor={215,215,215},
  fillPattern=FillPattern.Solid,
  textString="CHP"),
Text(
  extent={{-30,-60},{-4,-66}},
  lineColor={0,0,0},
  pattern=LinePattern.Dash,
  fillColor={215,215,215},
  fillPattern=FillPattern.Solid,
  textString="HP")}), Icon(graphics={
Ellipse(lineColor = {75,138,73},
        fillColor={255,255,255},
        fillPattern = FillPattern.Solid,
        extent={{-100,-100},{100,100}}),
Polygon(lineColor = {0,0,255},
        fillColor = {75,138,73},
        pattern = LinePattern.None,
```

```
            fillPattern = FillPattern.Solid,
            points={{-36,60},{64,0},{-36,-60},{-36,60}})}),
  Documentation(info="<html>
  <p><big>This is a second example aiming at illustrating the energy flows interacting between
the greenhouse and generation and storage units. In the previous example <a href=\"modelica://G
reenhouses.Examples.GlobalSystem_1\">GlobalSystem_1</a>, a considerable part of the produc
ed electricity is sold back to the grid. This electricity, in the absence of subsidies, is remunerated
at a price close to the wholesale price of electricity. Because the retail price of electricity is signif
icantly higher than the wholesale price, prosumers have a clear advantage at maximizing their le
vel of self-consumption. </p>
  <p><big>In order to evaluate the potential of such activity, we propose a new case study in w
hich we maximize the self-
consumption rate through the use of a heat pump. To that end, the heat pump model from the Gre
enhouses library is used and is connected in series with the CHP. The excess of electricity that in
itially was being fed back to the grid is now used to power the heat pump. The heat pump is size
d so that its nominal electrical capacity is equal to the excess of electricity of the CHP in nominal
 conditions. A heat-
driven control decides when to run the CHP. The heat pump is powered only by the CHP, and th
erefore never running independently. Electricity excess not consumed by the heat pump is sold to
 the grid. The greenhouse electrical demand not covered by the CHP is covered by the grid. The
electricity and gas prices are the same than in <a href=\"modelica://Greenhouses.Examples.Glob
alSystem_1\">GlobalSystem_1</a>. </p>
  <h4><big>Results</h4>
```

```
    <p><big>The results obtained from this simulation are discussed in the online documentation:
 <a href=\"https://greenhouses-library.readthedocs.io/en/latest\">https://greenhouses-
library.readthedocs.io/en/latest</a>. A more detailed discussion including a comparison between
 this example and <a href=\"modelica://Greenhouses.Examples.GlobalSystem_1\">GlobalSyste
m_1</a> is presented in the following article:</p>
    <p><big>Altes-
Buch Q., Quoilin S., Lemort V.. Modeling and control of CHP generation for greenhouse cultivat
ion including thermal energy storage. In <i>Proceedings of the 31st international conference on e
fficiency, cost, optimization, simulation and environmental impact of energy systems</i>, Guim
araes, Portugal, June 2018.</p>
</html>"));
end GlobalSystem_2;
```

Appendix C: Supplemental Source Code

“Waterconverter.m”

```
%Program converts transpiration data to waterfootprint.
clear all
format long


%------------------------------------------------------------------------
%constants
Area_floor=1.4*10^4; % m^2 area of greenhouse floor


time = table2array(readtable('time.xlsx','sheet', 2,... % reads time
   'Range','A1:A17520',...
   'ReadVariableNames',false));
% if an error comes up check to see if the amount of data you have is
% really at the year mark
%------------------------------------------------------------------------
% importing data from model for water calculation
transp_2017=table2array(readtable('Water and Co2 data.xlsx','sheet', 'Missouri',... % reads transpiration data from 2017
   'Range','K2:K17521',...
   'ReadVariableNames',false));
```

```
transp_2018=table2array(readtable('Water and Co2 data.xlsx','sheet', 'Missouri', ... % reads
transpiration data from 2018
    'Range','L2:L17521',...
    'ReadVariableNames',false));
transp_2019=table2array(readtable('Water and Co2 data.xlsx','sheet', 'Missouri', ... % reads
transpiration data
    'Range','M2:M17521',...                                   %from 2019
    'ReadVariableNames',false));



%-----------------------------------------------------------------------
%Integration
water2017=trapz(time,transp_2017); %applies trapezoidal rule %kg/m2
waterkg2017=(water2017*Area_floor);%kg of water

water2018=trapz(time,transp_2018); %applies trapezoidal rule %kg/m2
waterkg2018=(water2018*Area_floor);%kg of water

water2019=trapz(time,transp_2019);%applies trapezoidal rule %kg/m2
waterkg2019=(water2019*Area_floor);%kg of water
```

“DM_harvested.m”

```
% Program converts harvested dry matter to fresh weight.
 clear all
DM_har_2017=3257116.3;%mg/m2 simulated harvested dry matter
DM_har_2018=2729535.3;%mg/m2 simulated harvested dry matter
DM_har_2019=3116320.5;%mg/m2 simulated harvested dry matter
%coverts mg/m2 to kg
DM_h_2017=(DM_har_2017/(1*10^6));%kg/m2
DM_h_2018=(DM_har_2018/(1*10^6));%kg/m2
DM_h_2019=(DM_har_2019/(1*10^6));%kg/m2

Fresh_2017=DM_h_2017/0.05; %converts dry matter to fresh weight Kg/m2
Fresh_2018=DM_h_2018/0.05; %converts dry matter to fresh weight Kg/m2
Fresh_2019=DM_h_2019/0.05; %converts dry matter to fresh weight Kg/m2
```

“Co2_enrichment.m”

```
%Program to convert co2 supplied from external source to total mass used


time = table2array(readtable('time.xlsx','sheet', 2,... % reads time
    'Range','A1:A17520',...
    'ReadVariableNames',false));
%------------------------------------------------------------------------
 %importing data from model for CO2 supplied calculation
```

```
massco2_2017=table2array(readtable('Water and Co2 data.xlsx','sheet', 4,... % reads mass conc,
mg/m2 s
   'Range','B2:B17521',...
   'ReadVariableNames',false));
massco2_2018=table2array(readtable('Water and Co2 data.xlsx','sheet', 4,... % reads mass conc,
mg/m2 s
   'Range','C2:C17521',...
   'ReadVariableNames',false));
massco2_2019=table2array(readtable('Water and Co2 data.xlsx','sheet', 4,... % reads mass conc,
mg/m2 s
   'Range','D2:D17521',...
   'ReadVariableNames',false));

%--------------------------------------------------------------------------------------------------
%%conversion calculations and integration

CO2_2017=massco2_2017.*Area_floor; % converts mg/m2 s to mg/s
CO2footprint_2017=trapz(time,CO2_2017); % mg of CO2
A=co2footprint_2017/(1*10^6); % converts mg to kg
%
CO2_2018=massco2_2018.*Area_floor; % converts mg/m2 s to mg/s
CO2footprint_2018=trapz(time,CO2_2018); % mg of CO2
B=co2footprint_2018/(1*10^6); % converts mg to kg
```

```
%

CO2_2019=massco2_2019.*Area_floor; % converts mg/m2 s to mg/s

CO2footprint_2019=trapz(time,CO2_2019); % mg of CO2

C=co2footprint_2019/(1*10^6); % converts mg to kg
```

“Tsky.m”

```
%Program calculate fictitious sky temperature.

%assuming average cloudy day for U.S. to be 0.68



%-----------------------------------------------------------------------

%temperature input


T=table2array(readtable('fresno_2019.xlsx',... % reads data external temperature of location

    'Range','B2:B17521',...

    'ReadVariableNames',false));

%-----------------------------------------------------------------------


Tamb=T+273.15; % converting temperature to K

T_sky_eqn=0.037536.*Tamb.^1.5+0.32.*Tamb;


T_sky=273.15-T_sky_eqn;% converts temperature back to C

%T_sky is the input for sky temperature
```

“dusk_dawn.m”

```
%Program uses sunrise and sunset times to create a column of 1s and 0s.
%These inputs will be made for every 30mins and will let the model know
%when there is sunlight present during the day.

%Please put a 0 at the start of the data file for ilu_sp.
clear,clc
%------------------------------------------------------------------------
%Reads sunrise, sunset and time data

Trig_Da=table2array(readtable('dawn_dusk_Texas.xlsx',... % reads data for sunrise only
    'Range','A1:A365',...
    'ReadVariableNames',false));
Trig_Du=table2array(readtable('dawn_dusk_Texas.xlsx',... %reads data for sunset only
    'Range','B1:B365',...
    'ReadVariableNames',false));

time = table2array(readtable('time.xlsx','sheet',1)); %reads time

%------------------------------------------------------------------------
```

```
for i=1:365
    target(i) = i*86400;
end

TIME = zeros(48,365);
% day 1
[xa,~] = find(time(:,1)== target(1,1));
TIME(:,1) = time(2:xa,1);



for i=2:365
[x1,~] = find(time(:,1)== target(1,i-1));
[x2,~] = find(time(:,1)== target(1,i));
TIME(:,i) = time(x1+1:x2,1);
end

fixtriggerDa = zeros(365,1);
fixtriggerDa(1,1) = Trig_Da(1,1);

fixtriggerDu = zeros(365,1);
fixtriggerDu(1,1) = Trig_Du(1,1);
```

```
for i= 2:365
    fixtriggerDa(i) = Trig_Da(i,1) + TIME(1,i);
    fixtriggerDu(i) = Trig_Du(i,1) + TIME(1,i);
end


for i = 1:365
   Tg1 = fixtriggerDa(i);
    Tg2 = fixtriggerDu(i);
    for j = 1:48
        t = TIME(j,i);
        if t < Tg1
            results(j,i) = 0;
        elseif t > Tg2
            results(j,i) = 0;
        else
            results(j,i) = 1;


        end
    end
end
A=results;
B=reshape(A,[],1);% combines the 365 columns into one column
filename= 'inputdawndusk_Texas.xlsx';%creates file to store data
```

```
writematrix(B,filename,'Sheet',1,'Range','A1');%inputs column into file
```

“thermalscrn.m”

```
%Program uses sunrise and sunset times to create a column of 1s and 0s.
%These inputs will be made for every 30mins and will let the model know
%when to deploy the thermal screen and when to withdraw it during the day.

%Please put a 1 at the start of the data file for SC_usable
clear,clc

%-----------------------------------------------------------------------
%Reads sunrise, sunset and time data

Trig_Da=table2array(readtable('thermalscreen_Texas.xlsx',... % reads data for sunrise only
    'Range','A1:A365',...
    'ReadVariableNames',false));
Trig_Du=table2array(readtable('thermalscreen_Texas.xlsx',... %reads data for sunset only
    'Range','B1:B365',...
    'ReadVariableNames',false));

time = table2array(readtable('time.xlsx','sheet',1));
```

```
%------------------------------------------------------------------------


for i=1:365
    target(i) = i*86400;
end


TIME = zeros(48,365);
% day 1
[xa,~] = find(time(:,1)== target(1,1));
TIME(:,1) = time(2:xa,1);



for i=2:365
[x1,~] = find(time(:,1)== target(1,i-1));
[x2,~] = find(time(:,1)== target(1,i));
TIME(:,i) = time(x1+1:x2,1);
end


fixtriggerDa = zeros(365,1);
fixtriggerDa(1,1) = Trig_Da(1,1);


fixtriggerDu = zeros(365,1);
fixtriggerDu(1,1) = Trig_Du(1,1);
```

```
for i= 2:365

    fixtriggerDa(i) = Trig_Da(i,1) + TIME(1,i);

    fixtriggerDu(i) = Trig_Du(i,1) + TIME(1,i);

end


for i = 1:365

   Tg1 = fixtriggerDa(i);

    Tg2 = fixtriggerDu(i);

    for j = 1:48

        t = TIME(j,i);

        if t < Tg1

            results(j,i) = 1;

        elseif t > Tg2

            results(j,i) = 1;

        else

            results(j,i) = 0;


        end

    end

end

A=results;

B=reshape(A,[],1);% combines the 365 columns into on column
```

```
filename= 'inputthermalscreen_Texas.xlsx';%creates file to store data
writematrix(B,filename,'Sheet',1,'Range','A1');%inputs column into file
```

“sunsrisesunset.m”

```
%Program calculates the sunrise and sunset times for a specific location
%within the continental U.S.
%This will identify sunlight supplied setpoints as well as the conditions
%needed to thermal screen deployment.

%----------------------------------------------------------------------
%input latitude and logitude data points

longitude =-97.82;
latitude  = 32.77;

days = 1:1826;
%Selection of Time Zones for Continental US:

%Pacific Time,
PT = -8; %UTC
%Mountain Time
MT =-7; %UTC
```

```
%Central Time
CT = -6; %UTC
%Eastern Time
ET = -5; %UTC
UTCoff=CT;% selected time zone to be calculated.
%------------------------------------------------------------------------
%Equations
solarCorr = 4*(longitude - 15*UTCoff) + equationoftime(days);
delta = asind(sind(23.45)*sind(360*(days - 81)/365));
sunrise = 12 - acosd(-tand(latitude)*tand(delta))/15 - solarCorr/60;
sunset  = 12 + acosd(-tand(latitude)*tand(delta))/15 - solarCorr/60;
%------------------------------------------------------------------------
%plot figure for the year

f = figure;
f.Position = f.Position.*[1 1 1 0.7];
clf
plot(days, sunrise, days, sunset, 'LineWidth', 4)
axis([1 365 0 24])
title('sunrise and sunset')
xlabel('Day of Year') ; ylabel('Time of Day')
hold on
patch([days flip(days)], [sunrise flip(sunset)], [0.97 0.69 0.34])
```

```
%-----------------------------------------------------------------------

%Converts time from hours to seconds

A=sunrise*3600;

B=sunset*3600;


%Applies conditions for thermal screen deployment limits


C=(sunrise+1.5)*3600; % screen stays 1.5hr after sunrise

D=(sunset-1.5)*3600; %screen deploys at 1.5hr before sunset

%-----------------------------------------------------------------------

%data for sunlight supplied


filename= 'dawn_dusk_Texas.xlsx'; %creates spreadsheet to store data

writematrix(A',filename,'Sheet',1,'Range','A1'); %inputs sunrise time

writematrix(B',filename,'Sheet',1,'Range','B1'); %inputs sunset time


%data for thermal screen deployment


filename='thermalscreen_Texas.xlsx';%creates spreadsheet to store data

writematrix(C',filename,'Sheet',1,'Range','A1');%inputs sunrise limit time

writematrix(D',filename,'Sheet',1,'Range','B1');%inputs sunset limit time
```

“equationoftime.m”

```
function eot = equationoftime(range)


B = 360*(range - 81)/365;

eot = 9.87*sind(2*B) - 7.53*cosd(B) - 1.5*sind(B);


end
```

Appendix D: Preparing Input Files

Sample input files for Fresno, California for 2017 for 24hrs. The full file would be quite lengthy to insert so data for 24hrs is used to illustrate the files.

“Fresno_2017.txt”

```
#1
double tab(17520,10)  #Time T_out  RH_out       P_out  I_glob u_wind        T_sky
        T_air_spCO2_air_sp  ilu_sp
0       4       94.04   99000   0       1.3     11.27299592     20      405.32 0
1800    4       94.03   99000   0       1.2     11.27299592     20      405.32 0
3600    4       93.33   99000   0       1.2     11.27299592     20      405.32 0
5400    3       93.32   99000   0       1.2     12.5294889      20      405.32 0
7200    3       93.42   99000   0       1.2     12.5294889      20      405.32 0
9000    3       93.4    99000   0       1.1     12.5294889      20      405.32 0
10800   3       93.88   99000   0       1       12.5294889      20      405.32 0
12600   3       100     99000   0       1       12.5294889      20      405.32 0
14400   3       100     99000   0       1       12.5294889      20      405.32 0
16200   3       100     99000   0       1       12.5294889      20      405.32 0
18000   3       100     99000   0       1.1     12.5294889      20      405.32 0
19800   3       100     99000   0       1       12.5294889      20      405.32 0
21600   3       100     99000   0       0.9     12.5294889      20      405.32 0
23400   3       100     99000   0       0.7     12.5294889      20      405.32 0
25200   3       96.92   99000   0       0.6     12.5294889      20      405.32 0
```

27000	4	90.29	99000	24	0.6	11.27299592	20	405.32	1
28800	5	92.5	99000	93	0.7	10.0148119	20	405.32	1
30600	6	92.53	99000	174	1	8.754939895	20	405.32	1
32400	7	88.64	99000	111	1.3	7.493382923	20	405.32	1
34200	7	88.66	99000	144	1.3	7.493382923	20	405.32	1
36000	8	85.62	99000	300	1.4	6.230143996	20	405.32	1
37800	8	85.6	99000	333	1.2	6.230143996	20	405.32	1
39600	8	80.33	99000	361	1.2	6.230143996	20	405.32	1
41400	8	80.3	99000	510	1.2	6.230143996	20	405.32	1
43200	9	73.81	99000	522	1.2	4.965226109	20	405.32	1
45000	9	73.79	99000	516	1.2	4.965226109	20	405.32	1
46800	9	72.31	99000	493	1.3	4.965226109	20	405.32	1
48600	9	72.3	99000	456	1.2	4.965226109	20	405.32	1
50400	10	71.25	99000	15	1.2	3.698632239	20	405.32	1
52200	9	71.25	99000	13	1.1	4.965226109	20	405.32	1
54000	9	72.76	99000	10	1	4.965226109	20	405.32	1
55800	8	77.89	99000	7	0.8	6.230143996	20	405.32	1
57600	7	85.04	99000	4	0.6	7.493382923	20	405.32	1
59400	6	91.16	99000	1	0.6	8.754939895	20	405.32	1
61200	6	85.67	99000	0	0.7	8.754939895	20	405.32	0
63000	5	91.88	99000	0	0.9	10.0148119	20	405.32	0
64800	5	88.77	99000	0	1.1	10.0148119	20	405.32	0
66600	5	95.25	99000	0	1.2	10.0148119	20	405.32	0

```
68400  5        92.02   99000  0        1.3      10.0148119     20       405.32 0
70200  4        92.04   99000  0        1.3      11.27299592    20       405.32 0
72000  4        89.44   99000  0        1.4      11.27299592    20       405.32 0
73800  4        89.46   99000  0        1.4      11.27299592    20       405.32 0
75600  4        86.87   99000  0        1.4      11.27299592    20       405.32 0
77400  4        86.88   99000  0        1.4      11.27299592    20       405.32 0
79200  4        84.57   99000  0        1.5      11.27299592    20       405.32 0
81000  4        84.57   99000  0        1.5      11.27299592    20       405.32 0
82800  4        83.25   99000  0        1.5      11.27299592    20       405.32 0
84600  4        89.36   99000  0        1.4      11.27299592    20       405.32 0
86400  4        88.16   99000  0        1.4      11.27299592    20       405.32 0
```

“SC_usable_1Jan-31Decfresno.txt”

```
#1
double tab(17520,2)
0       1
1800    1
3600    1
5400    1
7200    1
9000    1
10800  1
```

12600	1
14400	1
16200	1
18000	1
19800	1
21600	1
23400	1
25200	1
27000	1
28800	1
30600	1
32400	0
34200	0
36000	0
37800	0
39600	0
41400	0
43200	0
45000	0
46800	0
48600	0
50400	0
52200	0

```
54000  0
55800  1
57600  1
59400  1
61200  1
63000  1
64800  1
66600  1
68400  1
70200  1
72000  1
73800  1
75600  1
77400  1
79200  1
81000  1
82800  1
84600  1
86400  1
```

“SP_1Jan-31Dec_2017.txt”

```
#1
```

```
double tab (17520,3) #Time   T_sp    CO2_air_sp

0       20      405.32

1800    20      405.32

3600    20      405.32

5400    20      405.32

7200    20      405.32

9000    20      405.32

10800   20      405.32

12600   20      405.32

14400   20      405.32

16200   20      405.32

18000   20      405.32

19800   20      405.32

21600   20      405.32

23400   20      405.32

25200   20      405.32

27000   20      405.32

28800   20      405.32

30600   20      405.32

32400   20      405.32

34200   20      405.32

36000   20      405.32

37800   20      405.32
```

39600	20	405.32
41400	20	405.32
43200	20	405.32
45000	20	405.32
46800	20	405.32
48600	20	405.32
50400	20	405.32
52200	20	405.32
54000	20	405.32
55800	20	405.32
57600	20	405.32
59400	20	405.32
61200	20	405.32
63000	20	405.32
64800	20	405.32
66600	20	405.32
68400	20	405.32
70200	20	405.32
72000	20	405.32
73800	20	405.32
75600	20	405.32
77400	20	405.32
79200	20	405.32

```
81000  20       405.32
82800  20       405.32
84600  20       405.32
86400  20       405.32
```

“SP_1Jan-31Dec_2017_Co+.txt”

```
#1
double tab (17520,3) #Time   T_sp    CO2_air_sp
0        20       1000
1800     20       1000
3600     20       400
5400     20       400
7200     20       400
9000     20       400
10800  20       400
12600  20       400
14400  20       400
16200  20       400
18000  20       400
19800  20       400
21600  20       400
23400  20       1300
25200  20       1300
```

27000	20	1300
28800	20	1300
30600	20	1300
32400	20	1300
34200	20	1300
36000	20	1300
37800	20	1300
39600	20	1000
41400	20	1000
43200	20	1000
45000	20	1000
46800	20	1000
48600	20	1000
50400	20	1000
52200	20	1000
54000	20	1000
55800	20	1000
57600	20	1100
59400	20	1100
61200	20	1300
63000	20	1300
64800	20	1300
66600	20	1300

68400	20	1300
70200	20	1300
72000	20	1300
73800	20	1300
75600	20	1300
77400	20	1300
79200	20	1300
81000	20	1300
82800	20	400
84600	20	400
86400	20	400

Appendix E:Data from Greenhouse Model Simulation

Table 134: Electrical Energy for Base Case and CO_2 enrichment.

Year	CRDs	Condition (Base/CO2+)	Total Electrical, kWhr/m^2	Buy Electrical, kWhr/m^2	Sell Electrical, kWhr/m^2	CHP Electrical, kWhr/m^2	HP Electrical, kWhr/m^2	Net Electrical, kWhr/m^2
2017	CA51	Base	438.40	438.40				
2018	CA51	Base	438.42	438.42				
2019	CA51	Base	438.43	438.43				
2017	CA51	CO2+	438.35	438.35				
2018	CA51	CO2+	438.37	438.37				
2019	CA51	CO2+	438.37	438.37				
2017	AZ80	Base	437.87	437.87				
2018	AZ80	Base	437.90	437.90				
2019	AZ80	Base	437.88	437.88				
2017	AZ80	CO2+	437.82	437.82				
2018	AZ80	CO2+	437.84	437.84				
2019	AZ80	CO2+	437.82	437.82				
2017	TX30	Base	437.71	437.71				
2018	TX30	Base	437.74	437.74				
2019	TX30	Base	437.73	437.73				
2017	TX30	CO2+	437.65	437.65				

Table 14 (cont.): Electrical Energy for Base Case and CO_2 enrichment.

Year	CRDs	Condition (Base/CO2+)	Total Electrical, kWhr/ m^2	Buy Electrical, kWhr/ m^2	Sell Electrical, kWhr/ m^2	CHP Electrical, kWhr/ m^2	HP Electrical, kWhr/ m^2	Net Electrical, kWhr/ m^2
2018	TX30	CO2+	437.63	437.63				
2019	TX30	CO2+	437.64	437.64				
2017	NY60	Base	438.35	438.35				
2018	NY60	Base	438.36	438.36				
2019	NY60	Base	438.35	438.35				
2017	NY60	CO2+	438.31	438.31				
2018	NY60	CO2+	438.32	438.32				
2019	NY60	CO2+	438.32	438.32				
2017	CO68	Base	437.33	433.21	81.24	112.42	27.05	351.97
2018	CO68	Base	437.31	432.24	80.19	112.50	27.24	352.04
2019	CO68	Base	437.34	432.17	83.49	116.58	27.92	348.68
2017	CO68	CO2+	437.26	433.25	79.70	110.26	26.54	353.55
2018	CO68	CO2+	437.28	433.25	82.81	113.96	27.11	350.44
2019	CO68	CO2+	437.28	434.23	80.21	109.44	26.18	354.02
2017	Pen42	Base	438.59	432.71	69.44	99.51	24.19	363.26
2018	Pen42	Base	438.58	431.06	68.54	100.57	24.51	362.52
2019	Pen42	Base	438.58	431.67	67.74	98.73	24.09	363.94

Table 14 (cont.): Electrical Energy for Base Case and CO_2 enrichment.

Year	CRDs	Condition (Base/CO2+)	Total Electrical, kWhr/m^2	Buy Electrical, kWhr/m^2	Sell Electrical, kWhr/m^2	CHP Electrical, kWhr/m^2	HP Electrical, kWhr/m^2	Net Electrical, kWhr/m^2
2017	Pen42	CO2+	438.54	434.64	69.02	96.23	23.31	365.62
2018	Pen42	CO2+	438.53	433.63	68.43	96.85	23.53	365.20
2019	Pen42	CO2+	438.52	433.99	67.55	95.21	23.13	366.44
2017	OH80	Base	438.39	431.03	74.00	107.65	26.30	357.03
2018	OH80	Base	438.37	429.46	74.33	110.05	26.81	355.13
2019	OH80	Base	438.38	429.90	74.63	110.00	26.89	355.27
2017	OH80	CO2+	438.35	433.22	73.62	104.09	25.34	359.60
2018	OH80	CO2+	438.32	432.41	74.09	105.67	25.67	358.32
2019	OH80	CO2+	438.32	432.48	74.49	106.19	25.86	357.99
2017	MN90	Base	437.21	437.21				
2018	MN90	Base	437.21	437.21				
2019	MN90	Base	437.21	437.21				
2017	MN90	CO2+	437.17	437.17				
2018	MN90	CO2+	437.17	437.17				
2019	MN90	CO2+	437.18	437.18				
2017	IL70	Base	436.90	428.46	75.25	111.22	27.54	353.22
2018	IL70	Base	436.93	427.42	79.31	117.55	28.73	348.11

Table 14 (cont.): Electrical Energy for Base Case and CO_2 enrichment.

Year	CRDs	Condition (Base/CO2+)	Total Electrical, kWhr/m^2	Buy Electrical, kWhr/m^2	Sell Electrical, kWhr/m^2	CHP Electrical, kWhr/m^2	HP Electrical, kWhr/m^2	Net Electrical, kWhr/m^2
2019	IL70	Base	436.90	426.86	78.54	117.42	28.85	348.32
2017	IL70	CO2+	436.84	430.08	75.16	108.79	26.88	354.92
2018	IL70	CO2+	436.85	429.43	79.06	114.35	27.86	350.37
2019	IL70	CO2+	436.84	428.93	78.59	114.58	28.08	350.34
2017	MIS50	Base	438.18	433.06	67.30	95.72	23.30	365.76
2018	MIS50	Base	438.20	431.43	68.69	99.63	24.17	362.74
2019	MIS50	Base	438.19	431.58	70.16	101.40	24.63	361.42
2017	MIS50	CO2+	438.12	434.34	66.79	93.17	22.60	367.55
2018	MIS50	CO2+	438.14	438.14				
2019	MIS50	CO2+	438.13	433.73	69.64	97.70	23.66	364.09

Table 15: Thermal Energy for Base Case and CO_2 enrichment.

Year	CRDs	Condition (Base/CO2+)	Total Thermal, $kWhr/m^2$	CHP Thermal, $kWhr/m^2$	HP Thermal, $kWhr/m^2$	Combined (CHP+HP), $kWhr/m^2$	Net Thermal, $kWhr/m^2$	Natural Gas, $kWhr/m^2$
2017	CA51	Base	182.03					
2018	CA51	Base	181.13					
2019	CA51	Base	182.04					
2017	CA51	CO2+	175.72					
2018	CA51	CO2+	175.29					
2019	CA51	CO2+	176.24					
2017	AZ80	Base	160.11					
2018	AZ80	Base	156.08					
2019	AZ80	Base	162.58					
2017	AZ80	CO2+	155.19					
2018	AZ80	CO2+	152.13					
2019	AZ80	CO2+	157.79					
2017	TX30	Base	187.88					
2018	TX30	Base	191.49					
2019	TX30	Base	200.11					
2017	TX30	CO2+	183.55					

Table 15 (cont.): Thermal Energy for Base Case and CO_2 enrichment.

Year	CRDs	Condition (Base/CO2+)	Total Thermal, kWhr/m^2	CHP Thermal, kWhr/m^2	HP Thermal, kWhr/m^2	Combined (CHP+HP), kWhr/m^2	Net Thermal, kWhr/m^2	Natural Gas, kWhr/m^2
2018	TX30	CO2+	186.52					
2019	TX30	CO2+	191.40					
2017	NY60	Base	222.49					
2018	NY60	Base	223.96					
2019	NY60	Base	223.26					
2017	NY60	CO2+	210.02					
2018	NY60	CO2+	212.61					
2019	NY60	CO2+	212.59					
2017	CO68	Base	210.13	137.52	65.31	202.83	-7.30	277.71
2018	CO68	Base	209.17	137.51	63.77	201.28	-7.89	277.80
2019	CO68	Base	215.18	142.35	64.22	206.57	-8.61	287.70
2017	CO68	CO2+	206.51	134.72	62.23	196.95	-9.56	272.20
2018	CO68	CO2+	211.86	139.06	62.47	201.53	-10.33	281.12
2019	CO68	CO2+	206.15	133.78	63.37	197.16	-8.99	270.24
2017	Pen42	Base	188.75	121.78	60.14	181.92	-6.82	245.88
2018	Pen42	Base	188.45	122.97	58.71	181.68	-6.77	248.37

Table 15 (cont.): Thermal Energy for Base Case and CO_2 enrichment.

Year	CRDs	Condition (Base/CO2+)	Total Thermal, kWhr/m^2	CHP Thermal, kWhr/m^2	HP Thermal, kWhr/m^2	Combined (CHP+HP), kWhr/m^2	Net Thermal, kWhr/m^2	Natural Gas, kWhr/m^2
2019	Pen42	Base	186.60	120.72	58.37	179.09	-7.50	243.83
2017	Pen42	CO2+	183.87	117.73	58.01	175.74	-8.13	237.73
2018	Pen42	CO2+	182.59	118.39	56.34	174.73	-7.87	239.16
2019	Pen42	CO2+	181.36	116.39	56.06	172.45	-8.91	235.11
2017	OH80	Base	201.99	131.90	64.66	196.57	-5.42	266.18
2018	OH80	Base	203.73	134.72	61.95	196.66	-7.07	271.97
2019	OH80	Base	204.79	134.74	63.60	198.34	-6.45	271.93
2017	OH80	CO2+	196.55	127.51	62.28	189.80	-6.76	257.33
2018	OH80	CO2+	196.73	129.33	59.25	188.57	-8.16	261.10
2019	OH80	CO2+	198.98	130.07	61.17	191.23	-7.74	262.51
2017	MN90	Base	305.58					
2018	MN90	Base	300.67					
2019	MN90	Base	309.11					
2017	MN90	CO2+	295.24					
2018	MN90	CO2+	290.52					
2019	MN90	CO2+	298.80					

Table 15 (cont.): Thermal Energy for Base Case and CO_2 enrichment.

Year	CRDs	Condition (Base/CO2+)	Total Thermal, kWhr/m^2	CHP Thermal, kWhr/m^2	HP Thermal, kWhr/m^2	Combined (CHP+HP), kWhr/m^2	Net Thermal, kWhr/m^2	Natural Gas, kWhr/m^2
2017	IL70	Base	208.06	136.65	65.40	202.05	-6.01	275.41
2018	IL70	Base	215.34	144.19	63.64	207.82	-7.51	290.82
2019	IL70	Base	215.32	144.10	65.02	209.12	-6.20	290.58
2017	IL70	CO2+	204.66	133.66	63.82	197.48	-7.19	269.39
2018	IL70	CO2+	210.40	140.22	61.71	201.92	-8.48	282.86
2019	IL70	CO2+	210.98	140.58	63.26	203.84	-7.14	283.51
2017	MIS50	Base	184.02	117.25	58.32	175.57	-8.45	236.64
2018	MIS50	Base	187.79	121.87	55.91	177.79	-10.01	246.12
2019	MIS50	Base	190.64	124.12	57.69	181.81	-8.83	250.57
2017	MIS50	CO2+	180.58	114.10	56.60	170.70	-9.88	230.30
2018	MIS50	CO2+	202.78					
2019	MIS50	CO2+	185.03	119.56	55.45	175.01	-10.02	241.40

Table16 : Yield, Water Usage and CO_2 externally supplied for Base Case and CO_2+.

Year	CRDs	Condition (Base/CO 2+)	Water, L/m^2	PWU, L/kg	Dry Matter, kg/m^2	Fresh Weight kg/m^2	CO2, kg/m^2
2017	CA51	Base	1,390.14	25.67	2.71	54.16	
2018	CA51	Base	1,457.33	29.76	2.45	48.97	
2019	CA51	Base	1,396.67	26.75	2.61	52.20	
2017	CA51	CO2+	1,209.75	23.49	2.58	51.51	505.52
2018	CA51	CO2+	1,267.64	27.07	2.34	46.82	449.89
2019	CA51	CO2+	1,220.41	23.35	2.61	52.26	419.77
2017	AZ80	Base	1,625.78	45.61	1.78	35.64	
2018	AZ80	Base	1,618.00	45.48	1.78	35.57	
2019	AZ80	Base	1,618.31	41.73	1.94	38.78	
2017	AZ80	CO2+	1,390.87	57.38	1.21	24.24	598.97
2018	AZ80	CO2+	1,392.86	48.94	1.42	28.46	608.61
2019	AZ80	CO2+	1,389.76	46.97	1.48	29.59	600.05
2017	TX30	Base	1,227.19	22.39	2.74	54.81	
2018	TX30	Base	1,167.83	26.86	2.17	43.48	
2019	TX30	Base	1,141.02	27.35	2.09	41.73	
2017	TX30	CO2+	1,075.21	20.35	2.64	52.84	690.79
2018	TX30	CO2+	1,023.99	22.58	2.27	45.35	659.77
2019	TX30	CO2+	1,010.76	22.38	2.26	45.17	654.95
2017	NY60	Base	934.73	15.44	3.03	60.56	

Table 16 (cont.): Yield, Water Usage and CO_2 externally supplied for Base Case and CO_2+.

Year	CRDs	Condition (Base/CO2+)	Water, L/m^2	PWU, L/kg	Dry Matter, kg/m^2	Fresh Weight kg/m^2	CO2, kg/m^2
2018	NY60	Base	919.09	16.04	2.87	57.31	
2019	NY60	Base	947.47	15.04	3.15	63.01	
2017	NY60	CO2+	849.48	11.96	3.55	71.02	163.09
2018	NY60	CO2+	838.34	11.96	3.50	70.09	151.77
2019	NY60	CO2+	838.31	11.96	3.50	70.10	151.24
2017	CO68	Base	1,327.15	16.30	4.07	81.42	
2018	CO68	Base	1,326.06	17.50	3.79	75.75	
2019	CO68	Base	1,297.40	17.44	3.72	74.41	
2017	CO68	CO2+	1,181.37	14.95	3.95	79.01	409.30
2018	CO68	CO2+	1,158.11	14.80	3.91	78.26	364.88
2019	CO68	CO2+	1,186.57	14.04	4.23	84.54	376.22
2017	Pen42	Base	993.83	14.63	3.40	67.92	
2018	Pen42	Base	889.63	14.78	3.01	60.17	
2019	Pen42	Base	985.08	16.12	3.06	61.12	
2017	Pen42	CO2+	898.38	11.55	3.89	77.76	253.52
2018	Pen42	CO2+	807.48	11.36	3.55	71.09	247.76
2019	Pen42	CO2+	888.80	12.79	3.48	69.51	247.39
2017	OH80	Base	1,008.26	13.99	3.60	72.06	
2018	OH80	Base	909.15	15.76	2.88	57.68	

Table 16 (cont.): Yield, Water Usage and CO_2 externally supplied for Base Case and CO_2+.

Year	CRDs	Condition (Base/CO2+)	Water, L/m^2	PWU, L/kg	Dry Matter, kg/m^2	Fresh Weight kg/m^2	CO2, kg/m^2
2019	OH80	Base	964.97	14.55	3.32	66.31	
2017	OH80	CO2+	904.25	11.43	3.95	79.09	247.76
2018	OH80	CO2+	816.63	12.24	3.34	66.70	247.39
2019	OH80	CO2+	865.91	11.62	3.73	74.53	352.22
2017	MN90	Base	984.90	12.18	4.04	80.87	
2018	MN90	Base	980.25	13.62	3.60	71.98	
2019	MN90	Base	950.95	12.43	3.83	76.50	
2017	MN90	CO2+	900.05	9.37	4.80	96.05	343.87
2018	MN90	CO2+	891.60	10.34	4.31	86.19	355.98
2019	MN90	CO2+	869.93	9.42	4.62	92.33	316.96
2017	IL70	Base	1,055.16	15.35	3.44	68.72	
2018	IL70	Base	948.60	16.64	2.85	57.01	
2019	IL70	Base	946.46	14.95	3.17	63.32	
2017	IL70	CO2+	947.11	12.59	3.76	75.21	453.88
2018	IL70	CO2+	855.20	12.93	3.31	66.16	443.73
2019	IL70	CO2+	857.57	11.81	3.63	72.59	438.83
2017	MIS50	Base	1,063.04	17.57	3.03	60.50	
2018	MIS50	Base	992.94	20.33	2.44	48.84	
2019	MIS50	Base	948.43	17.70	2.68	53.60	

Table 16 (cont.): Yield, Water Usage and CO_2 externally supplied for Base Case and CO_2+.

Year	CRDs	Condition (Base/CO2+)	Water, L/m^2	PWU, L/kg	Dry Matter, kg/m^2	Fresh Weight kg/m^2	CO2, kg/m^2
2017	MIS50	CO2+	947.29	14.54	3.26	65.14	355.34
2018	MIS50	CO2+	896.86	16.43	2.73	54.59	313.14
2019	MIS50	CO2+	852.08	13.67	3.12	62.33	318.57