Design and Optimization of Linear Concentrating Photovoltaic System for Combined Power Heat and Desalination

Mohammad Karimzadeh Kolamroudi

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Mechanical Engineering

Eastern Mediterranean University
January 2016
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Mustafa Tümer
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Mechanical Engineering.

Assoc. Prof. Dr. Hasan Hacışevki
Chair, Department of Mechanical Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Mechanical Engineering.

Asst. Prof. Dr. Naser Kordani
Co-Supervisor

Prof. Dr. Uğur Atikol
Supervisor

Examining Committee

1. Prof. Dr. Uğur Atikol
2. Prof. Dr. Fuat Egelioğlu
3. Assoc. Prof. Dr. Mustafa Dağbaşı

ABSTRACT

Currently, the world is being faced with an impending energy supply crisis. In other to provide solution to this dilemma, there is need to expand and effectively maintain our energy resource and this has to be done in such a way that it takes into account the surrounding environment; as a matter of fact, this is incredibly vital for progress to be made with renewable energy technologies. However, regardless of this urgent demand, it is crucial to appraise the bigger environmental impact that is associated with the use of these renewable energy resources. The purpose of this study is to provide a better understanding of Linear Concentrating Photovoltaics system (LCPVs) from an engineering point of view; this technology is as a result of the recent developments and numerous researches in this field; and so, this has led to the development of this system.

Python software was used to simulate the performance of LCPVs, and the results were compared with results obtained by other authors, who used the Engineering Equation Solver (EES). Moreover, the effect of a cooling fluid in the system and its effect on the productivity of multi-junction cells were also evaluated.

The results acquired from the data simulation painted a picture of LCPVs from an environmental perspective, electricity plus water heating. In addition, this system utilised renewable energy which helped in reducing CO_2 emission. The comparison of the current study results with those obtained previously, indicated that a great similarity exist with a minimal error of 0.002 (kWh). Also, when the design of the

LCPVs was changed, the average efficiency increased to 0.598 % during the simulation; this is in contrast to the old design model.

Keywords: Linear Concentrating Photovoltaics System (LCPVs), Solar Energy, Thermal Energy, Multi-junction Cell

ÖZ

Dünya şu anda, yaklaşmakta olan bir enerji arzı kriziyle karşı karşıyadır. Bu ikilemi çözmek için, enerji kaynağımızı genişletmek ve etkin bir şekilde koruma ve kullanmak aynı anda bunun bölgesel çevreyi hesaba katacak şekilde yapmak gerekmektedir. Nitekim, yenilenebilir enerji teknolojileri ile ilerleme kaydedilmesi son derece hayati önem taşımaktadır. Bu acil talebe bakılmaksızın, yenilenebilir enerji kaynaklarının kullanımı ile alakalı daha büyük ve geniş kapsamda araştırma ve geliştirmeler ciddi önem taşımaktadır.

Bu çalışmanın amacı, Lineer Konsantre Fotovoltaik (LCPV) 'yi mühendislik açısından daha iyi anlamamızı sağlamaktır; Lineer Konsantre Fotovoltaik teknolojisi, son gelişmelerin ve bu alandaki sayısız araştırmanın sonucunda yenilenebilir enerji üretimine ve gelişimine yol açmaya devam ediyor.

LCPV'nin performansını simüle etmek için Python yazılımı kullanılmış ve sonuçlar, Mühendislik Denklem Çözücü'sünü (EES) kullanan diğer yazarlar tarafından elde edilen sonuçlarla karşılaştırılmıştır. Ayrıca, soğutma sıvısının sistem üzerindeki etkisi ve bunun çoklu bağlantı hücrelerinin verimliliği etkisi de değerlendirilmiştir.

Veri simülasyonundan elde edilen sonuçlar, çevresel açıdan LCPV'lerin elektrik ve su ısıtması üzerindeki durumunu lanse etti. Buna ek olarak, bu sistemde CO_2 emisyonunun azaltılmasına yardımcı olan yenilenebilir enerji üretildi. Mevcut çalışma sonuçlarının daha önce elde edilen bilgiler ile karşılaştırılması sonucu 0.002 (kWh) minimum hata ile büyük bir benzerliğin var olduğunu kanıtladı. Ayrıca,

LCPV'lerin eski tasarım modelinin aksine tasarımı değiştirildiğinde, simülasyon sırasında ortalama verimlilik% 0.598'e yükseldi.

Anahtar Kelimeler: Doğrusal Konsantre Fotovoltaik Sistem (LCPV), Güneş Enerjisi, Isıl Enerji, Çok Kontaktlı Hücre

ACKNOWLEDGMENT

First and foremost, I want to give special gratitude to almighty Allah for giving me the required strength, opportunity and good health to complete my Master's Thesis in the prestigious Mechanical Engineering department of Eastern Mediterranean University which is a great citadel of learning.

Also worthy of mention are the persons of Prof. Dr. Ugur Atikol who was my project supervisor and Co-supervisor Assist. Prof. Dr. Naser Kordani whose expertise and guidance helped me gain a lot of vital information and knowledge for my thesis. In addition, I humbly appreciate the general Academic staff in the Mechanical engineering department of Eastern Mediterranean University for the enormous knowledge they have impacted on me which was very helpful during the duration of my Masters program and also my classmates whom I met at the Mechanical engineering department for their friendliness and good team work.

Special thanks goes to my family, for their immense support which that gave me throughout the course of my Master's program and least I forget to my wonderful friend Poorya Ghafoorpoor Yazdi thank you so much for been helpful.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

$A_{concentrator}$	Area of Concentrated Solar Radiation [m^2]
η_{cell}	Efficiency
k	Thermal Conductivity [$kW/m*\ K$]
μ	Dynamic Viscosity [$kg/m*s$]
ν	Kinematic Viscosity [m^2/s]
ρ	Density [kg/m^3]
B_o	Boiling Number
Co	Convection Number
Cp	Specific Heat [$kJ/kg*K$]
D_h	Hydraulic Diameter [m]
$E_{thermal}$	Thermal Energy of the Fluid in the Flow Channel [kWh]
$Electricity$	Average Electricity Production over the LCPV Array [kWh]
f	Friction Factor
Fr_{liquid}	Froude Number of Fluid in Liquid State
$Flowrate$	Volumetric Flowrate [gal/min]
G	Mass Flux [kg/m^2*s]
$gravity$	Earth's Gravitational Pull [m/s^2]
ht	Heat Transfer Coefficient [$kW/m^2*\ K$]
$Insulation$	Insulation Value for the Flow Channel [$kW/m^2*\ K$]
$Length$	Module Length [m]

$\dot{m}$	Mass Flowrate $[kg/s]$
Nu	Nusselt Number
p	Perimeter of the Flow Channel Cross Section $[m]$
P_{cell}	LCPV System Power $[kW]$
Pr	Prandtl Number
q_{total}	Heat Entering the Flow Channel $[kW]$
$\ddot{q}_{rad}$	Solar Radiation $[kW/m^2]$
$R_{channel}$	Thermal Resistance of the Flow Channel Insulation $[kW/m^2 * K]$
Re	Reynolds Number
$Rows$	Number of Module Rows in the LCPV Array
R_{factor}	Insulation Value for the Storage Tank $[kW/m^2 * K]$
T_{air}	Outdoor Air Temperature $[K]$
T_{bulk}	Temperature of the Bulk Fluid Flow in the Channel $[K]$
$\bar{T}_{surface}$	Average Channel Surface Temperature $[K]$
U_{liquid}	Velocity of the Fluid in Liquid State $[m/s]$
V_{use}	Volume of Fluid that Leaves System Due to Use $[m^2]$
$w_{channel}$	Width of the Flow Channel $[m]$

Chapter 1

INTRODUCTION

1.1 Importance of Energy

Energy plays an important role in the development of a country. As a matter of fact, adequate and authentic energy resources are essential for the economic development of a country. Thus, alternative sources of energy are sought and needed everywhere because the fossil fuels reservoirs of our planet are rapidly diminishing.

Pollutants from crude oil exploration and bituminous coal industries have contaminated the environment to a large extent and this is as a result of their heavy use and consumption. Actually, these energy sources produce high amounts of energy although the fuel sources are steadily becoming depleted [1]. Therefore, supplementary options should be put in place to replace the present energy sectors. The world at present has numerous sustainable energy sources such as hydro, solar, wind and geothermal power. However, in order to have a better understanding of these energy sources, the exemplified energy found in the other energy sectors must be further investigated in terms of their sustainability.

As a result of solar energy technology, there has been a significant increase in power plants that utilize solar energy for power generation. Various thermal systems such as Parabolic Trough Solar Collectors (PTSCs), solar dishes, and solar towers can be used to generate power. The parabolic trough solar collectors are the most frequently used solar technology when the sun's energy is utilized as the source of power [2].

1.2 Motivation of study

As mentioned earlier, energy plays an important role in today's world and also, fossil fuels are used extensively in the industries. However, using fossil fuels is not without drawbacks as they impact the environment negatively; consequently resulting to unnatural side effects such as acid rain, environmental hazards and health problems.

Hence, the option available for eradicating such hazardous effects would be to fully take advantage of renewable energy sources such as wind, solar, and hydraulic energy. Solar energy is believed to be the most sustainable and abundant energy resource in the world. The earth absorbs approximately 26,000 TW of solar energy, while the earth's population used approximately 18 TW in the year 2005 [2]. Linear Concentration Photovoltaic System (LCPVs) can play a major role in the reduction of the negative effects that emanates from fossil fuels; moreover, this technology provides a cost-effective solution since it operates by solar energy. In LCPVs, solar rays are focused onto a little zone of photovoltaic surface where electricity and heat is generated. LCPVs can have three generations at the same time namely electricity, heat and fresh water.

Therefore, in this era of global warming, ozone layer depletion, and ice melting of the South and North poles, the development of LCPVs is a necessity in our world today and due to its various applications in different sectors, this research will help reinforce the fundamental understanding of this technology and the viability of LCPVs as an alternative to traditional electricity and thermal energy production.

1.3 Objectives

The aim of this study is to improve the performance of the LCPVs by modifying the design that was proposed by Tony Krezmann [2]. The objective is to produce electricity and thermal energy for heating water and purifying well water.

1.4 Organization of the Thesis

The background information on the photovoltaic literature review, solar concentrators and LCPV cooling system are presented in the literature review in chapter 2. Chapter three also introduces and describes the system while chapter four discusses on the simulation conducted with Python programming language. In Chapter 5, the data accumulated by comparing the experimental and numerical results was reviewed. And finally, the conclusion of the study together with the recommendations for the future work were presented in chapter 6.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

Alexandre-Edmond Becquerel in 1839 discovered the photovoltaic effect, but 44 years later, Charles Fritts designed the first PV cell using selenium [3, 4]. Since the invention of PV cells, scientists have sought to increase cell efficiencies and decrease cell costs. They were familiar with the photovoltaic effect for over 170 years, however, no one embarked on using the system widely until the energy crisis in the 1970s where the PV cells became very popular on a massive scale. Immediately after this energy crisis and till the mid-1990s, the enthusiasm for PV reduced due to an increase in global warming; this generated great concern as well as prompted the spending of huge amounts of money on researches within the alternative energy circle.

A lot of attempts have been made to expand the effectiveness of solar cells; furthermore, advances in multi-junction cells have also helped enhance the effectiveness of PV cells. Of the diverse solar cell innovations, the multi-junction concentrator cells have exhibited the highest efficiency [5]. There are three main categories of the solar cell technology: the first is the original innovation-this is comprised of large areas that have single layer p-n (positive-negative) diode cells that are made up of many silicon cells. The second generation cells are delivered by utilizing the discharge of tiny sheets of wafer embedded in a lattice matched material. The innovation is centered on creating highly efficient models; however, this raises the manufacturing cost. And so, multi-junction cell innovation has the lion's share of this generation technology [6].

In 1954, the Bell Laboratories invented a silicon solar based cell with 6% efficiency [7]. As a result, cells that are silicon based have peaked to higher efficient levels of up to 24.7%. This great feat was achieved by a university in the United states in 1999 [8]. Solar based silicon cells are a widely recognized type of photovoltaic cells. Presently, cells which are silicon dependent are reaching a peak hypothetical level of 33% [9]. However, there are some problems associated with the silicon cells, for example, the silicon wafers used to make silicon cells are expensive and solar grade silicon is hard to find [10]. Another drawback of the silicon cells is the low technology improvement potentials compared to a 36 layer multi-junction cell which has a theoretical maximum efficiency of 72% [11].

In the field of multi-junction cell, the objective is to make highly efficient cells. As a matter of fact, the efficiencies of cells as high as 58% are not irrational; this is according to Bett et al [12]. According to Bett et al, Efficiencies above 60% may turn into a reality with respect to the rapid advancement in multi-junction cell technology. The scholar also stated that Multi-junction cells can be considered as one of the advanced solar energy technologies as their efficiency has been on the rise ever since. Multi-junction cells standout amongst most of the present day advancements in technologies which provide energy that are solar reliant. Examination and assembling in this area has further improved drastically; and so, when the efficiency of the cell increases, the multi-junction cell expense, in effect, will reduce.

2.2 LCPV System

A significant part of the examination relating to LCPV is coordinated towards developing a higher effective cell in order to expand the application possibility. Lately, LCPV cell fabricating procedures have enhanced; this has helped raise the bar on the material's purity during manufacturing as well as lead to the reduction in the material's deformities. With the modifications in solar based technologies, multi-junction aggregating cells have become more and more popular [13].

2.3 Solar Concentrators

The ability to harness solar energy is a vital aspect of any established solar technology that hopes to be of great promise. Majority of previously exhibited plants that are solar powered use a form that retains the suns radiation through a concentrator. One of the primary reasons for this is to center the sun radiation onto a particular area which produces a relatively large amount of energy flux. This energy is responsible for producing highly confined temperatures, and, at times, for example, the main source of electricity production from the sun's radiation, while the LCPVs makes use of thermal energy as its auxiliary source. The use of solar concentration has helped to protect the environment and has also decreased the cost of installation to a very large extent.

2.3.1 Concentrating Solar Power Plants

Solar power plants can be categorized into parabolic trough, solar power tower and dish Stirling engine plants (see Figure 2.1) [14]. Collectively, solar power plants usually require large investment, high operational and repair costs, and a vast landmass. However, the benefit of this technology is that they can retain thermal energy that can be further utilized to generate electricity unlike the photovoltaic system that makes use of costly electric storage technique. Solar power plants have been designed in such a way to make use of the suns radiation as its primary source of energy; also, the plant has to be capable to generating electricity at night times and during times where the level of solar intensity is not high enough. Two techniques are employed in most solar power generation centers which include; the design of a plant to generate more heat that is requires in producing a substantial amount of electricity during the day, then the excess energy is stored and used up when there is no or low solar intensity energy. Another technique can be the use of use of fossil fueled heat generation [15].

Figure 2.1. Solar Power Plants [15]

2.3.2 Power Tower

Usually, the power towers is comprised of a receiver that is centralized; this is because it is the point of convergence for large group of reflective mirrors which follows the path of the sun rays and then focuses the suns' radiation onto the receptor. These systems have solely been utilized for large scale power plants [16].

2.3.3 Parabolic Dish

In the parabolic dish, the sun radiation is concentrated on the receiver which is usually placed at the focal point of the dish. The dish is made of a material with reflective properties and which can detect the path of the sun rays using two axes. These systems are usually equipped with a stirling engine for creating thermal energy by the aggregated solar radiation which is then converted to electrical power; also, the parabolic dish is used to focus the rays from the sun onto a little zone of PV cells [16].

2.3.4 Fresnel Lens

Fresnel focal lens are gaining wide popularity amongst the solar concentrating lens. These lenses operate with the same principle as the convex lenses with respect to their light concentration mechanism. Moreover, the light rays can be refracted by adding prisms. In Figure 2.2 a Fresnel lens which is on the left and a convex lens which is on the right are illustrated.

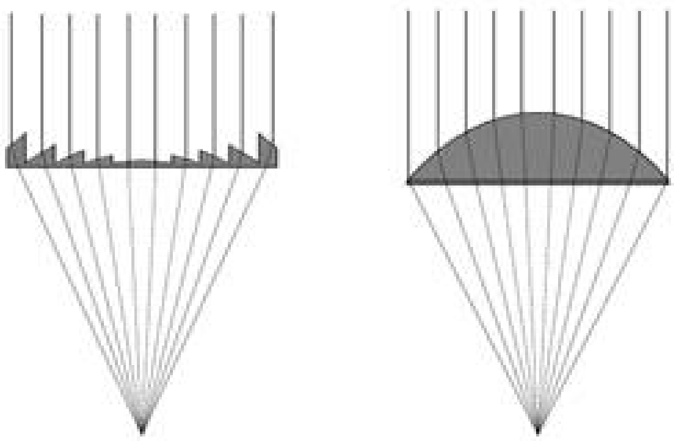

Figure 2.2. Fresnel Lens (left) and Convex Lens (right) [14]

However, concentrating Fresnel lens are not without faults either as they sometimes lose radiation because of reflection, absorption and refraction effects [14]. The effectiveness of the Fresnel lens is likewise exceptionally reliant on the precision of the system that helps it to track sun rays [15].

The world is confronted by an unavoidable energy supply crisis. Keep in mind, however, that the end goal is to support and build our energy producing systems to comply with the environment; and so, it is important to carry out research on renewable energy advancements. The LCPVs serves two noteworthy purposes: it produces electricity and the waste heat that is gathered can be utilized for warming purposes. In this regard, numerous studies have been conducted to discover the answers for increasing its efficiency as well as advancement of LCPV; however, as far as the researcher knows, no study has benefited from Python programing language in this regard. Therefore, in this study Python programing language was used to examine the efficiency, electricity and thermal energy of LCPVs in our new design.

Chapter 3

SYSTEM DESCRIPTION AND MATHEMATICAL MODEL

3.1 System Description

As indicated in figure 3.1, the LCPVs is made up of the receiver, well water tank, Frensel lens, multi-junction cell, clean water tank, hot water tank, pump, glass wall, tracking system. The designed system is intended for a 6 person’s family. In order to deliver the required energy, the simulation made use of five rows of five meter long modules, or the sum of 25 meters of LCPV modules (figure 3.3). A 0.02 m high flow channel and a width of 0.04m, corresponded to a 1:2 height to width ratio (figures 3.4, 3.5). The linear solar concentrator had a concentration width of 0.01 m as dictated by the Emcore CTJ triple junction cell's aperture area [17, 18]. It should be mentioned here that there is a pipe with 0.01 in diameter inside the channel which cools down the surface temperature.

Subsequently after sunshine, as depicted by the picture, pumps I and II begin to work, with pump I transferring the well water to well water channel and after which, a circulation in the channel around the glass wall enters the well water tank in the LCPVs. Pump II circulates the pipe water from the city water tank to the receiver and vice versa. The water is heated in the receiver and then this heat is transferred through the pipe into the city’s water tank; hence, increasing the temperature of city’s water tank. Two forms of energy are produced by the LCPVs which include; electric and thermal energy. The lens captures the solar energy and converts it into the above mentioned forms of energy through the multi-junction cells. The thermal energy that is generated is then utilized for clean and hot water applications with the system. Although, a fraction of the electric energy produced is usually used for the system pumps and another part is used by household equipment. The greater the quantity of flow, the more likely pumps will be in consuming more energy and this reduces the available quantity for other energy uses.

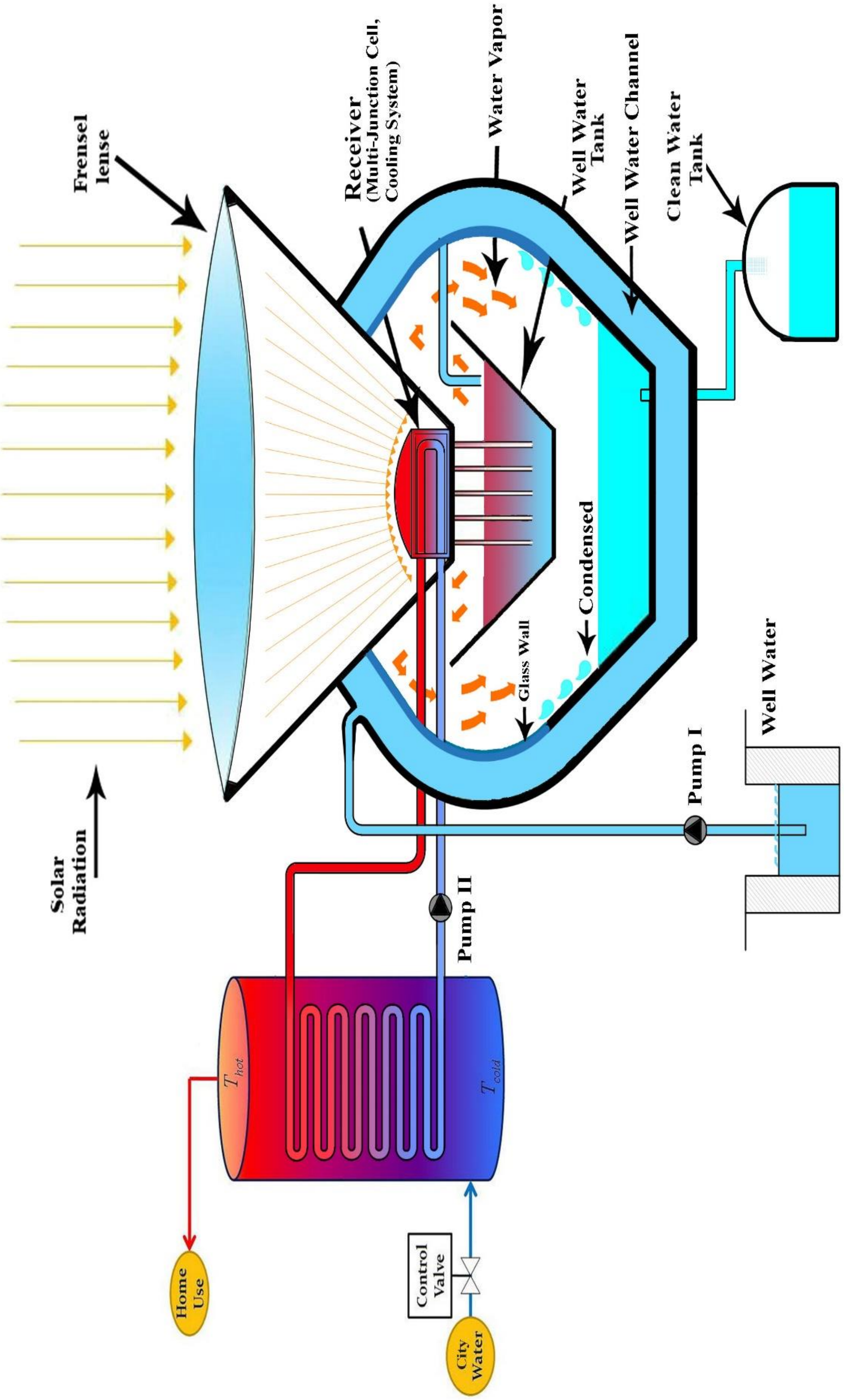

Figure 3.1. LCPV System

The generated thermal energy heats the available well water in the tank through some installed components located at the bottom of the receiver. This heat makes the well water vaporize and then, the glass wall, having a lower temperature condenses the water. The refined or distilled water then flows into the clean water tank and afterward is utilized for household purposes.

In order to increase the well water temperature before it flows into the well water tank it has to be pre-heated as this process helps in making the evaporation procedure faster. It additionally reduces the temperature of the glass wall. At some points after sunshine and after the well water circulation in well water channel, there is an increment in the temperature of the glass wall and in the event that it is not cooled down properly, the refining procedure takes longer period of time. Consequently, since the temperature of the circulating well water around the glass wall is equivalent with the water temperature in the environment temperature, it helps us to reach our target.

The PV module receives highly localized solar intensity and this energy creates a high temperature in the multi-junction cells. With a concentration of about 80 times in the LCPV modules, the Fresnel lenses were estimated to have a transmissivity of 85%. The efficiency of the cells decreased when there was an increase in the temperature of the multi-junction cells, this further leads to a decrease in the electricity output. Keeping in mind the end goal is to create a balanced and an ideal operating efficient system, a system that can cool the LCPVs efficiently must be considered and provided, in other to ensure optimum performance.

Figure 3.2 shows that after solar radiation reaches the lens, the lens concentrate the radiation on the receiver. The cooling system in the receiver prevents an increase in temperature on the surface of the receiver that consists of multi-junction cells; hence this increases the average efficiency. Furthermore, a portion of the solar radiation received by the lenses is reflected (decreased) and the remaining 85 percent is concentrated on PV module. About 15 percent of the transmitted 85 percent solar radiation is converted into heat and used to produce electricity and the rest is used to produce hot water and clean water.

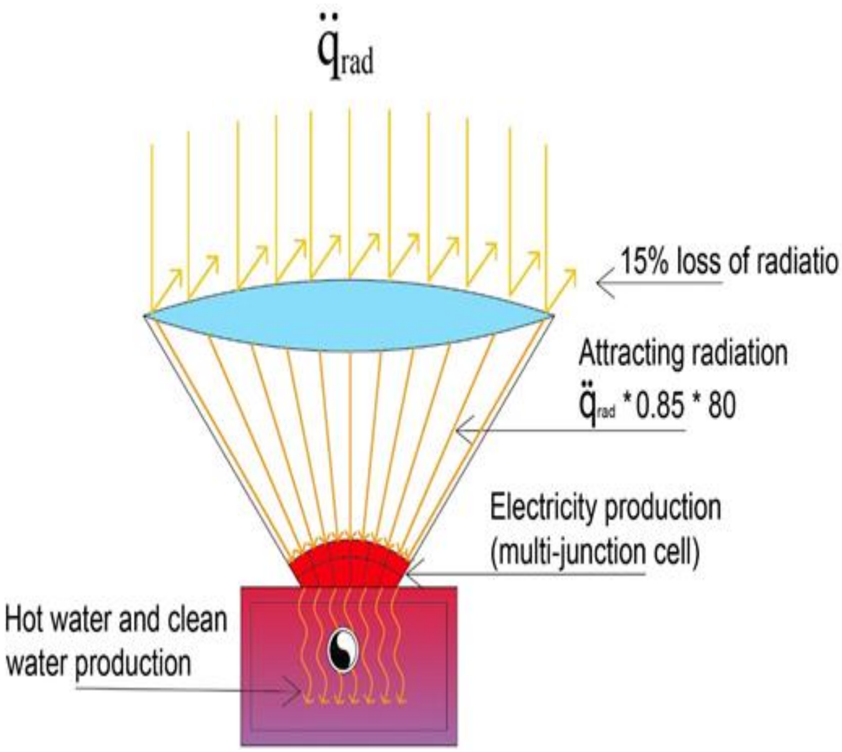

Figure 3.2. Cross section of the Receiver in LCPVs

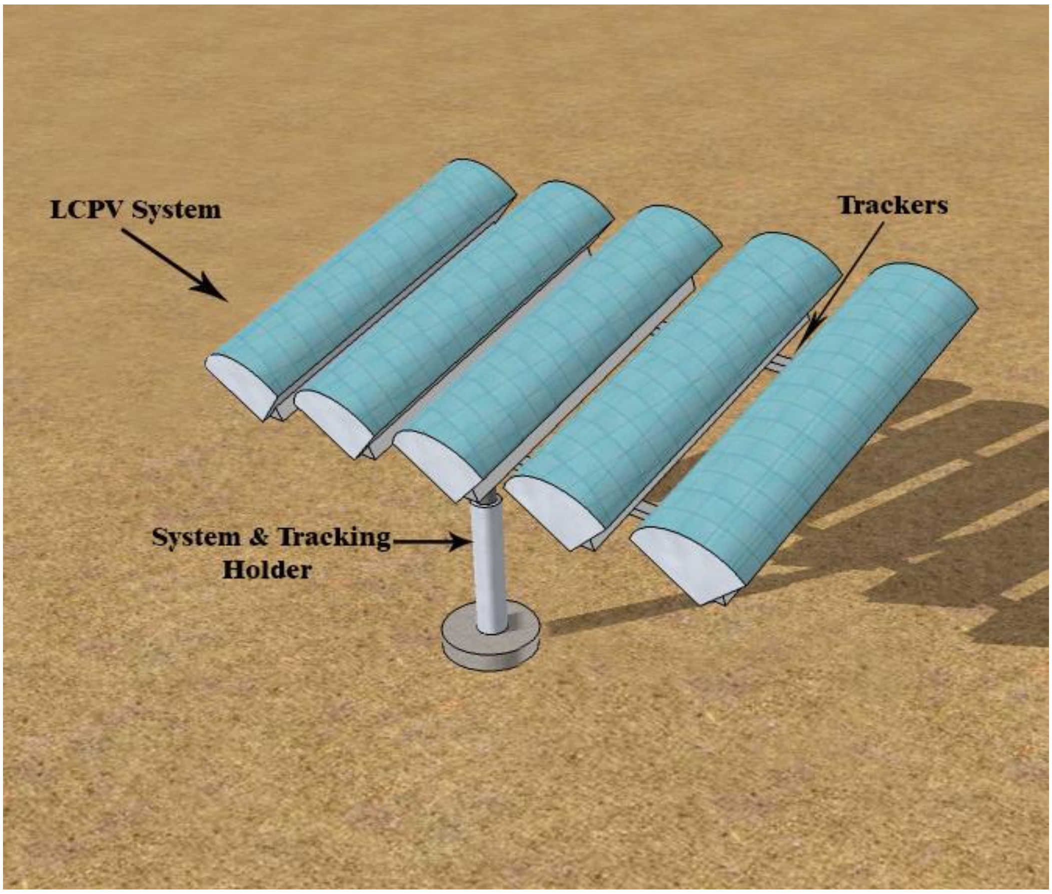

Figure 3.3. 3D Rendering of a LCPVs with SketchUp

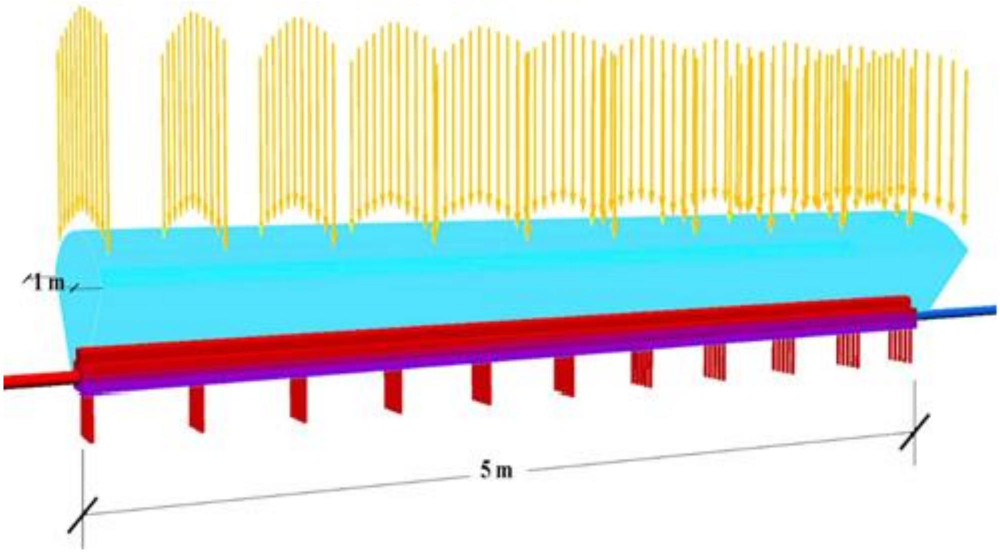

Figure 3.4. 3D Rendering of a LCPVs side view

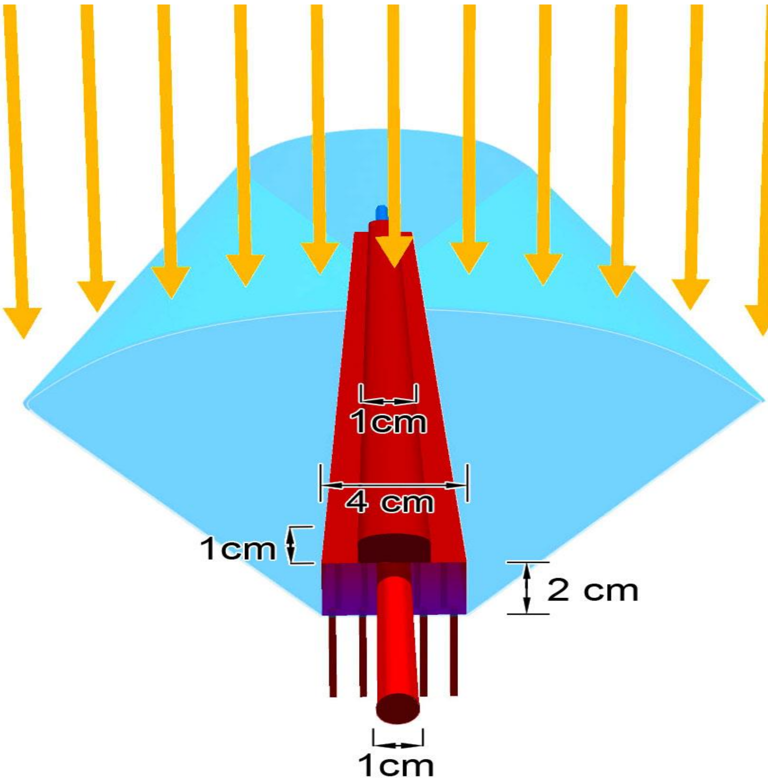

Figure 3.5. 3D Rendering of a LCPVs front view

3.2 Mathematical Model

In each portion, the following equations were used for calculation purposes; this will be explained subsequently in more details below. Equation 3.1 is used to calculate the heat that enters the fluid (q_{total}) and is equal to the solar heat entering the flow channel. In order to determine the heat that is entering the fluid, the solar flux ($\ddot{q}_{heat}$) must be calculated using the input solar radiation ($\ddot{q}_{rad}$) and the temperature dependent cell efficiency (η_{cell}), in which T_{bulk} represents the temperature of the bulk fluid flow in the channel.

$$q_{total} = \ddot{q}_{heat}.(A_{concentrator}) - (A_{surface}.(T_{bulk} - T_{air})) \tag{3.1}$$

Where,

$$\ddot{q}_{heat} = \ddot{q}_{rad}.(1 - \eta_{cell}) * 0.85 * 80 \tag{3.2}$$

The heat energy which gets into the channel and the fluid's enthalpy are calculated by using Equations 3.2 and 3.3. $\ddot{q}_{heat}$ is the heat entering the fluid from the solar radiation, assuming that the heat transfer through the thin aluminum channel is negligible. The calculation accounts for the concentrator's optical losses (assumed to be 15%) and this is made up of 80 times of the solar concentration. The h_{bulk} for each segment is calculated using the previous segment's bulk flow enthalpy ($h_{bulk,i-1}$) plus the segmented enthalpy due to the incoming thermal energy h_{heat}:

$$h_{bulk} = h_{bulk,i-1} + h_{heat} \tag{3.3}$$

Where,

$$h_{heat} = \frac{q_{total}}{\dot{m}} \tag{3.4}$$

After the bulk fluid enthalpy for the segment is calculated, the heat transfer coefficient must be calculated in order to determine the surface temperature. The first step in executing this calculation is to determine whether the fluid flow is liquid, two-phase, or steam. The bulk fluid enthalpy is compared to the saturated liquid enthalpy, and if it is higher, then the fluid is either two-phase or steam. The Kandlikar correlation is used to estimate the heat transfer coefficient for both steam and two phase flow [20].

For two-phase flow in horizontal and vertical tubes, the Kandlikar correlation was performed. According to the correlation, the LCPV is not considered as horizontal and the flow is regarded as vertical fluid flow, especially in winter season, for smaller altitude angle, tilting the system closer to vertical. Equations 3.5 through 3.15 show different steps in computing the Kandlikar coefficient in a vertical tube, where Co is considered to be the convection number, ht represents the heat transfer coefficient, x shows the quality, f represents the friction factor, Fr showing the Froud number, B_o representing boiling number, and G is the mass flux [21]. The first two equations calculate whether the two-phase flow is convective-boiling-dominant (CBD) or nucleate boiling- dominant (NBD) in which the heat transfer coefficient is the larger of the two solutions:

$$ht_{CBD} = 1.136.(\mathrm{Co}^{-0.9}).((1-x)^{0.8}).fFr_{liquid}.ht_{liquid} + 667.2.(Bo^{0.7}).(1-x)^{0.8}).ht_{liquid} \quad (3.5)$$

$$ht_{NBD} = 0.6683.(Co^{-0.2}).((1-x)^{0.8}).fFr_{liquid}.ht_{liquid} + 1058.(Bo^{0.7}).(1-x)^{0.8}).ht_{liquid} \quad (3.6)$$

Where,

$$Co = \left(\left(\frac{\rho_{gas}}{\rho_{liquid}}\right)^{0.5}\right).\left(\left(\frac{1-x}{x}\right)^{0.8}\right) \quad (3.7)$$

$$fF\mathrm{r}_{\mathrm{liquid}} = 1 \text{ for vertical tubes} \quad (3.8)$$

ht_{liquid} can be found using the Gnielinski correlation, which is valid for liquid flows within the range $0.5 \leq Pr_{liquid} \leq 2000$ and $2300 \leq Re_{liquid} \leq 10{,}000$ [22].

$$ht_{liquid} = \frac{(Re_{liquid}-1000) * Pr_{liquid} * \left(\frac{f}{2}\right).(K_{liquid}/D_h)}{1+12.7*\left(\left(Pr^{\frac{2}{3}}{}_{liquid}\right)-1\right)*\left(\left(f/_2\right)^{0.5}\right)} \quad (3.9)$$

$$B_o = \frac{q_{total}}{G.h_{fg}} \quad (3.10)$$

$$G = \rho.U_{liquid} \quad (3.11)$$

$$h_{fg} = h_{gas} - h_{liquid} \quad (3.12)$$

$$f = \sqrt{\left((1.58.\ln(Re_{liquid})) - 3.28\right)} \quad (3.13)$$

$$Re_{liquid} = \frac{U_{liquid}.D_h}{\nu} \quad (3.14)$$

$$D_h = \frac{4.A_{Cross-section}}{p} \quad (3.15)$$

If the flow is liquid, then the first step in calculating the heat transfer coefficient is to determine whether the flow is a laminar or turbulent flow regime by using the flow velocity, hydraulic diameter, and fluid viscosity, as seen in Equation 3.16. From this information, the Nusselt number is known for laminar flows, shown in Equation 3.17, where the channel width to height ratio is two [23]. Equation 3.18, the Dittus-Boelter correlation, is used to calculate the Nusselt number for turbulent flows [24]. After calculating the Nusselt number, the convective heat transfer coefficient can be calculated using Equation 3.20. The heat transfer coefficient is necessary to calculate the surface temperature, which is extremely important because it affects the cell's efficiency.

$$Re = \frac{U_m.D_h}{\nu} \tag{3.16}$$

$$Nu = 4.12 \tag{3.17}$$

$$Nu = 0.023Re^{0.8}pr^{0.4} \tag{3.18}$$

Where,

$$Pr = \frac{Cp.\mu}{k} \tag{3.19}$$

$$ht = \frac{Nu.k}{D_h} \tag{3.20}$$

By using the information accumulated from the flow channel model, the simulation becomes capable of calculating the PV cell temperature and assumes that the temperature is the same at the top surface of the channel. This assumption is made because the cell is welded to the channel and the heat transfer through the metal weld is much higher than the heat transfer through the insulation to the surrounding area. In order to calculate the average cell efficiency along the length of the LCPVs, the average surface temperature must be calculated. The average surface temperature is dependent on the average bulk flow temperature, the thermal energy entering the channel, and then the average heat transfer coefficient can be gotten by using Equation 3.21.

$$\bar{T}_{surface} = \bar{T}_{balk} + \frac{q_{total}}{\overline{ht}} \tag{3.21}$$

Where,

$$\overline{ht} = \sum_1^i \frac{ht_i}{i} = \frac{ht_1 + ht_2 \ldots + ht_{i-1} + ht_i}{i} \tag{3.22}$$

$$\bar{T}_{balk} = \sum_1^i \frac{T_{bulk,i}}{i} = \frac{T_{bulk,1} + T_{bulk,2} \ldots + T_{bulk,i-1} + T_{bulk,i}}{i} \tag{3.23}$$

The average cell efficiency ($\bar{\eta}_{cell}$) can now be deduced from Equation 3.24, where the average efficiency at room temperature (293.15 ˚K) is 36.5% and the change in efficiency with respect to temperature is $-0.06\%/^{\circ}K$ for the Emcore Corporation CTJ photovoltaic cells. These cell parameters were determined through experimental characterization and are found on the CTJ cell specification sheet [25]. The system electrical power P_{cell} can be calculated by using Equation 3.25, with respect to a solar concentration of 80 and optical transmissivity of 85%.

$$\bar{\eta}_{cell} = 36.5\% - \left(\bar{T}_{surface} - 293.15k\right) * 0.06\% \tag{3.24}$$

$$P_{cell} = \bar{\eta}_{cell}.Rows.Length.Width_{concentration}.\ddot{q}_{rad} * 80 * 85\% \tag{3.25}$$

The cell power is in unit of kW and this unit simply needs to be multiplied by the number of hours that the system is under the specified conditions in order to obtain the energy output in kWh. The simulation runs for each hour of the day, and so therefore, the cell power is multiplied by 1 hour to convert to the unit of energy.

Thereafter, it is necessary to compute the variables that have been used for the LCPV heat storage system which is done after the flow and cell conditions have been simulated. The storage tank for hot water is insulated (R_{tank}). In order to calculate the energy within the hot storage tank, an energy balance must be completed. Equation 3.26 provides the balanced energy for the storage tank which stores the heat content; E_{tank} is the energy in kJ of the tank. Figure 3.6 provides a better understanding for the balance of the storage tank [26]:

$$E_{tank} = E_{tank,i-1} + E_{in} + E_{citywater} - E_{use} - E_{out} - E_{loss} \tag{3.26}$$

Where,

$$E_{tank,i-1} = \text{Tank energy from the previous hour iteration} \tag{3.27}$$

$$E_{in} = h_{bulk}.\dot{m}.Time \text{ (Note: Time= 1 hour or 3600s)} \tag{3.28}$$

$$E_{citywater} = V_{use}.\rho_{citywater}.h_{citywater} \tag{3.29}$$

$$E_{use} = V_{use}.\rho_{tank.i-1}.h_{tank,i-1} \tag{3.30}$$

$$E_{out} = h_{tank,i-1}.\dot{m}.Time \tag{3.31}$$

$$E_{loss} = Surface\ Area_{\ tank}.R_{tank}.(T_{tank} - T_{room}).Time \tag{3.32}$$

Now that the tank energy has been calculated, the enthalpy in the tank can also be deduced by simply dividing the tank energy by the mass of the fluid in the tank, as seen in Equation 3.33. In order to calculate the tanks fluid temperature, the heat content in the tank is inputted as the temperature function. This temperature is used in the next hourly iteration as the initial bulk flow temperature for the fluid entering the flow channel.

$$h_{tank} = \frac{E_{tank}}{Mass_{tank}} \tag{3.33}$$

The surface temperature, bulk temperature, tank temperature, cell power, cell efficiency and the tank energy are stored by the simulation on a spreadsheet, as shown in Appendix A.3.

Chapter 4

SIMULATION WITH PYTHON PROGRAMMING LANGUAGE

4.1 Introduction to Python Programming Language

During the final phases of the year 1989, Guido van Rossum, a young Dutch scientist, developed Python programming language. Later on, he identified the following goals for Python as a programming language; this includes:

- A straightforward and instinctive programming language, while their rivals in the professional world have the power.
- Open Source; every man could help in the advancement of this project.
- The code is straightforward and simple to peruse an English content.
- Suitable for every day work and for quick and easy design of a program with a very short time.

The programming language has met some of these needs. As at this time, Python as a scripting language has been further developed and popularized in the virtual world known as the internet. Python programming models, (for example, object-based and imperative programming and axis function) help bolster, and decide the sort of a variable a dynamic system will take.

Python is a powerful scripting language which is used on a large scale and in influential works almost everywhere in the present day world for example in NASA, due to its simplicity and capacity to manage smaller projects and modify it to manage even larger projects. It is also one of the favorite Google programing language. Moreover, Autodesk as the world's biggest programming organization manufactures computer-aided design (CAD) and visualization which makes use of Python scripting functionality.

Learning the language is not difficult as it is one of the easiest languages to learn. Python can be utilized in system programming - user interface - Internet programming - numerical and computational applications - database programs - image processing - artificial intelligence - distributed objects - simulation - Robotics - Mobile Programming - Security and network and etc.

4.2 Method of Computing

The research data is arranged into columns which can be found in Appendix 3.B. Column 1 takes into account the hours ranging from the 9th and 10th of July between 6 AM to 8 PM. Column 7 captures data from the direct sun rays which comes in contact with the LCPVs surface each time, in kW/m2 (before concentration). This radiation is based on an active two axis tracking system which traces the location of the sun in the sky and has a minimum error of degree of about 1 in the azimuth and horizontal directions. The average air temperature is recorded in degree Kelvin in column 10.

The solar radiation data used in this study is based on the data received from National Solar Radiation Database (NSRDB). The NSRDB has a large collection of solar data from various locations throughout the US [19]. The next step involves inputting the flow rate which is in volumes and the units is in gal/min. The pumps are terminated from working after the information obtained about the sun’s radiation parametric table reads zero, which means there is no sunshine. It should be noted that the flow rate volume influences a lot of sections in the system which is responsible for the quantity of parasitic electricity used up during the process of pumping. During the change in the volume of flow, the heat transfer coefficient, thermal energy produced, and the channel surface temperature all change. This is due to the fact that the LCPVs utilizes a coupled solar thermal energy and a photovoltaic; moreover, the rate of flow has an impact on condition of the cell and the systems production of electricity.

Furthermore, Python software is very similar to other coding software. First, the parameters were defined, and then the required data were substituted. The formulas for the calculation of the required parameters were also written. Then, each available parameter in the main equation is written down. After the equations were written down, the software ran and the value of each parameter were obtained by pressing the calculate button.

4.3 Air temperature and solar radiation during the 9th and 10th of July

Figure 4.1 and 4.2 shows the temperature and the solar radiation during 9th and 10th of July. The available data in the figure 4.1 was extracted from NSRDB which is included in Appendix 3. A. Table 1 and Table 2 shows NSRDB data for solar radiation and air temperatures.

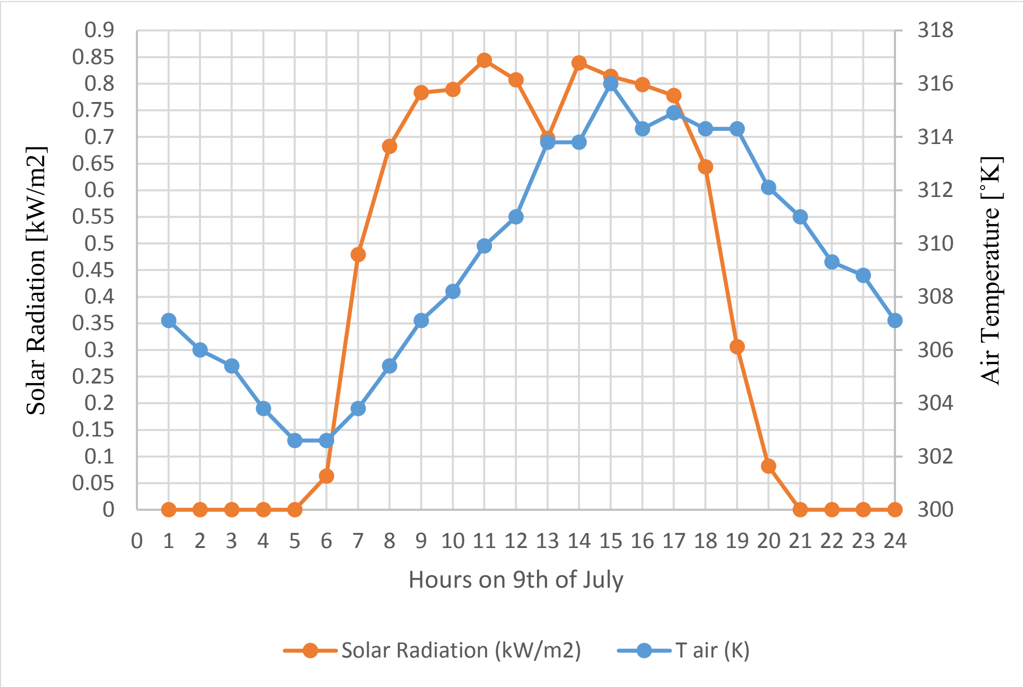

Figure 4.1. Temperature and the Solar Radiation variation of 9th of July

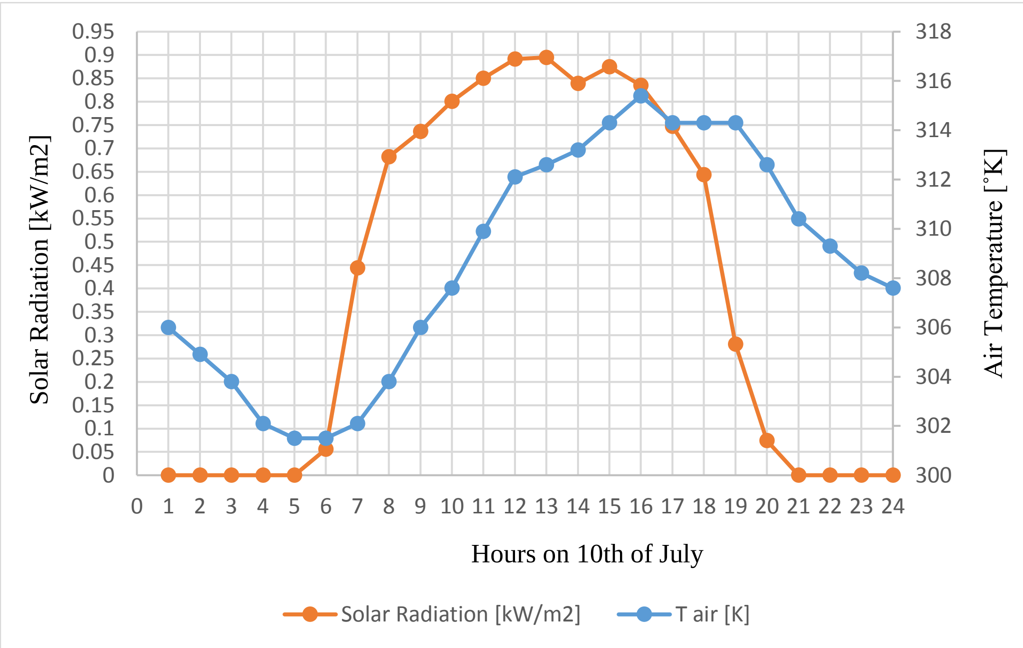

Figure 4.2. Temperature and the Solar Radiation variations of 10th of July

Table 1. Temperature and the Solar Radiation on 9th of July

hours in Day	Air Temperature $[K]$	Solar Radiation $[kW/m^2]$
6:00 9th of July	302.6	0.063
7:00 9th of July	303.8	0.479
8:00 9th of July	305.4	0.682
9:00 9th of July	307.1	0.783
10:00 9th of July	308.2	0.789
11:00 9th of July	309.9	0.844
12:00 9th of July	311.0	0.807
13:00 9th of July	313.8	0.697
14:00 9th of July	313.8	0.839
15:00 9th of July	316.0	0.814
16:00 9th of July	314.3	0.798
17:00 9th of July	314.9	0.778
18:00 9th of July	314.3	0.644
19:00 9th of July	314.3	0.306
20:00 9th of July	312.1	0.082

Table 2. Temperature and the Solar Radiation on 10th of July

hours in Day	Air Temperature $[K]$	Solar Radiation $[kW/m^2]$
6:00 10th of July	301.5	0.056
7:00 10th of July	302.1	0.444
8:00 10th of July	303.8	0.682
9:00 10th of July	306.0	0.736
10:00 10th of July	307.6	0.801
11:00 10th of July	309.9	0.85
12:00 10th of July	312.1	0.891
13:00 10th of July	312.6	0.895
14:00 10th of July	313.2	0.839
15:00 10th of July	314.3	0.875
16:00 10th of July	315.4	0.835
17:00 10th of July	314.3	0.747
18:00 10th of July	314.3	0.644
19:00 10th of July	314.3	0.281
20:00 10th of July	312.6	0.074

It should be noted that (figures 4.1 and 4.2) that within time range of 1:00 o'clock and 2:00 o'clock, there was a drop in solar radiation; meanwhile, the temperature remained constant, this due to the fact that solar radiation is dependent on the sun radiation; in other words, if at a certain time the sky is cloudy or there are shadow or shade casts on the panel, it can decrease the solar radiation. However, this might not be the case when talking about temperature as the temperature at any point depends on the previous point in time.

4.4 Pump

Figure 3.4 is a D5SOLAR-38/700B pump which was used in our system. The DC pump is approved for use with PV operated solar thermal systems of up to $8m^2$. This pump can be operated in combination with the UVR61-PV and a PV panel (>25W) even directly without a battery.

Figure 4.3. D5SOLAR-38/700B pump

4.4.1 Pump Specifications

Delivery: up to 1.5 m3/h

Head: up to 3 m

Supply voltage: 8-24 V

Minimum startup power: > 1W

Maximum power consumption: approx., 22W

Current consumption: 0.25 – 1.46 A

Power supply: single-phase 50 Hz

Maximum operating pressure: 10 bar

Temperature of pumped liquid:

Brass: -10°C to +95°C

Insulation class: F

Protection: IP42

4.4.2. Pump Materials

Pump body: Brass

Impeller: Nory l- Stainless steel

Lower sleeve: Stainless steel/Nory l

Wear Ring: Ceramic

Bearings: Carbon - Ceramic

Elastomers: EPDM

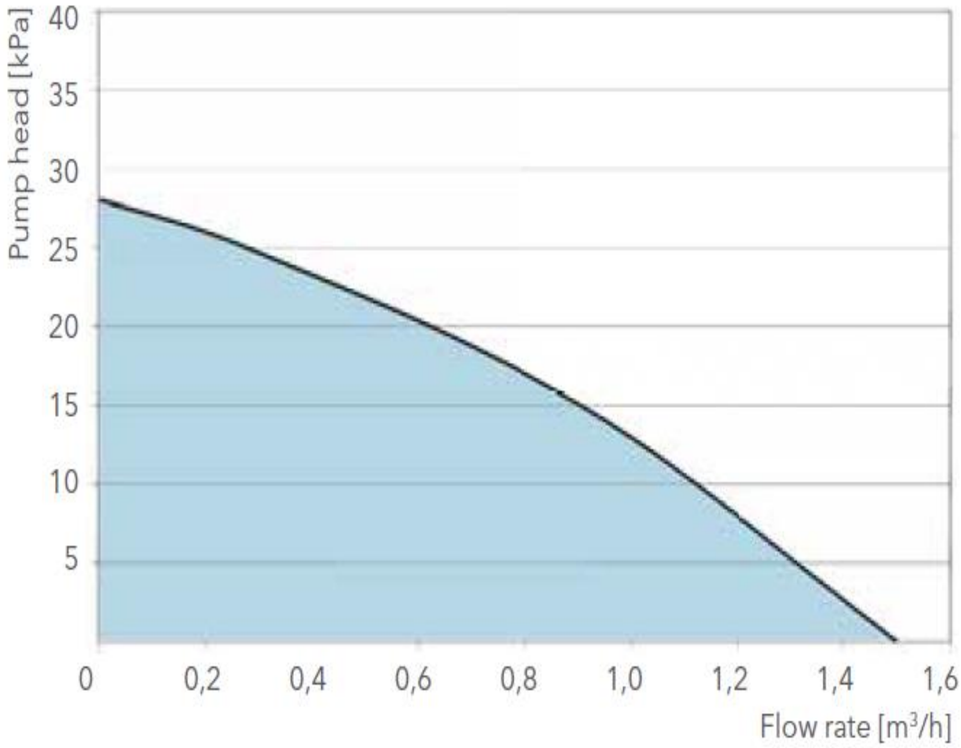

Figure 4.4. Variation of Pump head with flow rate

4.4.3 Pump Applications

For domestic hot water, heating water, water/glycol mixtures as well as other media on request.

4.5 Average Efficiency Calculation

To calculate average efficiency, formula (3.24) was used. The value of $T_{surface}$ was obtained from equation (3.21). And by pressing the calculate button on the software, the value of the average efficiency was obtained. Figure 4.5 is an example of the written codes for average efficiency.

```
def calcAvgEfficiency(self):
    # .365 - (Tsurface - 293.15)*.06
    Tsurface = Constants['Average Channel Surface Temperature']

    ans = 0.365 - (Tsurface - 293.15)*0.0006
    Constants['Average Efficiency'] = ans
    self.Ncell.set(ans)
```

Figure 4.5. The Average Efficiency Code of Python

4.6 The method of thermal energy calculation

To calculate the thermal energy, equation (3.30) was used. Figure 4.6 is an example of thermal energy code. Table 3 also shows thermal energy for different parameters.

```
def calcEuse(self):
    # Vloss * Ptanki-1 * htanki-1
    Vloss = Constants['Volume Of Fluid That Leaves System To Use']
    PTnk = Constants['Density Of Fluid In Tank From Previous hr Iteration']
    Htnk = Constants['Enthalpy Of Fluid In Storage Tank At Channel Segment']

    ans = Vloss * PTnk * Htnk
    Constants['Thermal Energy Leaving Storage Tank For Use'] = ans
    self.Euse.set(ans)
```

Figure 4.6. The Thermal Energy Code on Python

4.7 The method of electricity Calculation

To calculate the generated electricity, equation (3.25) was utilized. Average efficiency obtained from equation (3.24) was substituted into the equation and since all the parameters were known in the equation, the value of electricity was easily obtained. Figure 4.7 is an example of electricity code. Table 4 also shows the value of electricity for various parameters.

```
def calcElectricity(self):
    # CelllEff * (rows*L*w_concentrator)*(Radiation*80*.85)
    sr = Constants['Solar Radiation']
    w_c = Constants['Width Of Concentration Area']
    rows = Constants['Number Of Module Row In The LCPV Area']
    l = Constants['Module Length']
    cellef = Constants['Average Efficiency']
    ans = cellef * rows * w_c * sr * 80 * .85
    self.elect.set(ans)
```

Figure 4.7. The Electricity Code on Python

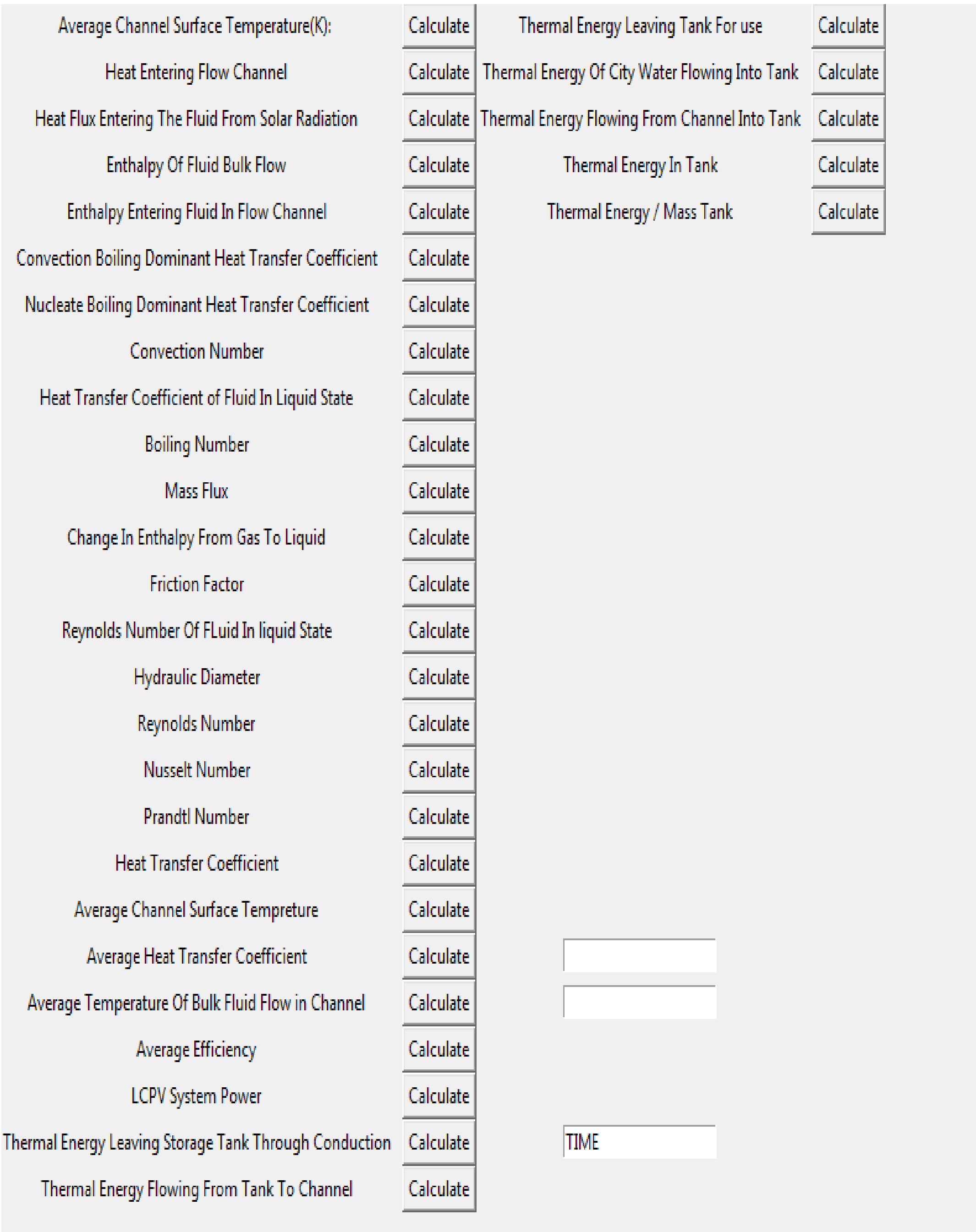

Figure 4.8. The flow chart of algorithm programming results

Chapter 5

RESULTS AND DISCUSSION

5.1 Introduction

The numerical study as well as an in depth comparisons of the outcomes from the numerical researches are presented in this section. In order to check the eligibility of different numerical methods to apply in different research areas, the results of this study had to be compared with the results obtained from the Kerzmann [2] study. Efforts have been made over the time to improve the existing methods or to create newer methodologies.

5.2 Electricity and Thermal Energy

An essential part of the LCPVs assessment is its temperature profile because this is where a greater part of the system is affected; this includes the fluid enthalpy and the efficiency of the cell. The interdependency between the flow rate and the generation of electricity is incredibly essential on ground and this is responsible for measuring the pumps' loads; this plays an important part in the optimization of the linear concentrating photovoltaic system. As stated earlier, an increase in the rate of flow produces a better efficiency and as such, this increases the electrical output of the system. Moreover, as the electrical energy drawn from pumps increases, the flow rate also increases; and so, the total electrical output of the system is decreased.

5.3 Comparison of Results

To ensure the accuracy of our results while using the Python programing platform, the results of this study were compared with the results of the study done with Engineering Equation Solver (EES) by Kerzmann et al. [2], (table 3). Interestingly, similar results were reported in both studies; however, there was a minimal error of about 0.002 (kWh).

Table 3. The Comparison of Energy Simulation Results

Average Efficiency [2] = 34.115% Average Efficiency (Python) = 34.113%	Energy (kWh)
Average Thermal Energy [2]	7.118 (kWh)
Average Thermal Energy (Python)	7.116 (kWh)
Average Electricity [2]	3.675 (kWh)
Average Electricity (Python)	3.673 (kWh)

The higher efficiency and energy savings is another advantage of the linear concentrating photovoltaic system. Another great advantage is that there is little pollution dissipated by using this form of renewable energy.

5.4 The Comparison of the New Design with the Old Design

In this section, a new design for LCPVs is shown in Figure 5.1. As in the figure 5.1, well water with brown color is shown. In the past, the design was in such a way that well water was directly in the tank i.e. before entering into the well water tank. In the new design, the well water is circulated around a glass wall so as to clean water tank and then into the well water tank. When we have solar radiation, the temperature built up at the glass wall causes the well water to evaporate. This increase in temperature reduces the rate of distillation. The new design makes it possible to maintain the temperature in the glass wall as well as the well water temperature, before entering the main tank which brings about an increase in the tank’s volume.

This design increases the evaporation rate of well water and distils it into clean water, resulting in an increase in the system’s efficiency.

As we mentioned earlier, in the new design, well water circulates in the glass wall before entering the main reservoir while its temperature also increases. In the new design, the temperature of well water increases from 293 ºK to 303 ºK. This increase in temperature causes an increase in the efficiency of the thermal energy. It is also worth noting that this was compared with an old design during the 9^{th} and 10^{th} of July. Table 4 shows the results of this comparison with increased system efficiency of 0.598 %.

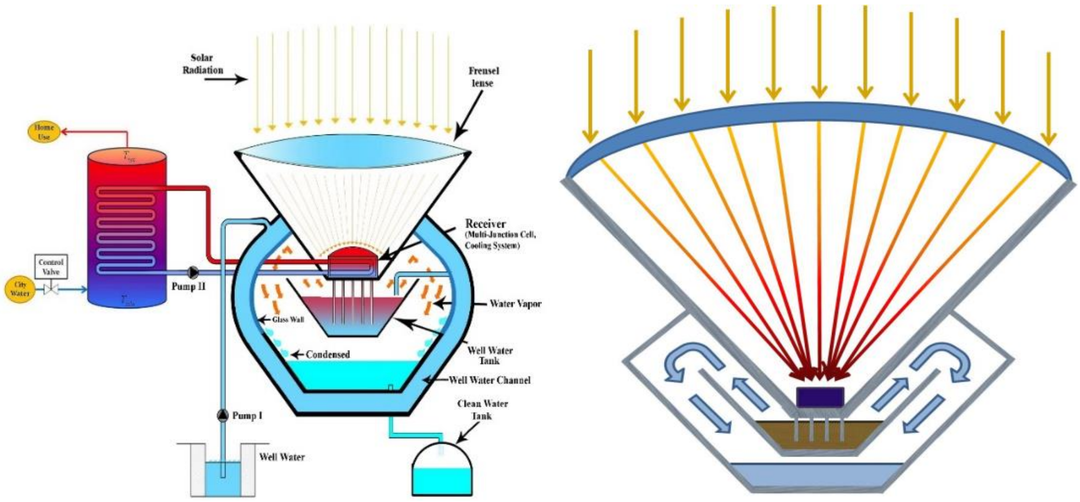

Figure 5.1 Proposed Design (left) and the design studied in Ref [2] (right)

Table 4. The Comparison of Energy Simulation Results for Old Design and New Design

Average Efficiency [2] = 34.115% Average Efficiency (proposed design) = 34.713%	Energy
Average Thermal Energy [2]	7.118 (kWh)
Average Thermal Energy (proposed design)	7.259 (kWh)
Average Electricity [2]	3.675 (kWh)
Average Electricity (proposed design)	3.737 (kWh)

Figure 5.2 and 5.3 compares the average efficiency for 9th and 10th of July i.e. between the old and new design. It is clear that the average efficiency of the new design is higher than that of the old design between 6:00 to 20:00 hours of the day (Appendix 3. B). It is also worth mentioning that when there is no sunshine the value of the average efficiency is equivalent to zero.

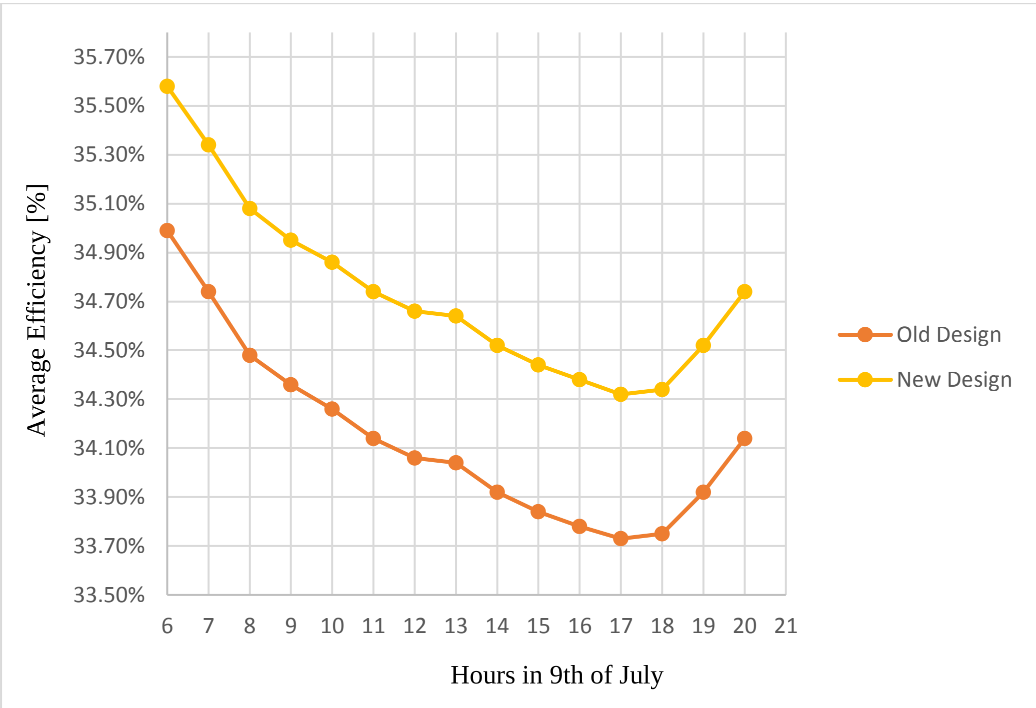

Figure 5.2. Average Efficiency in hours on 9th of July

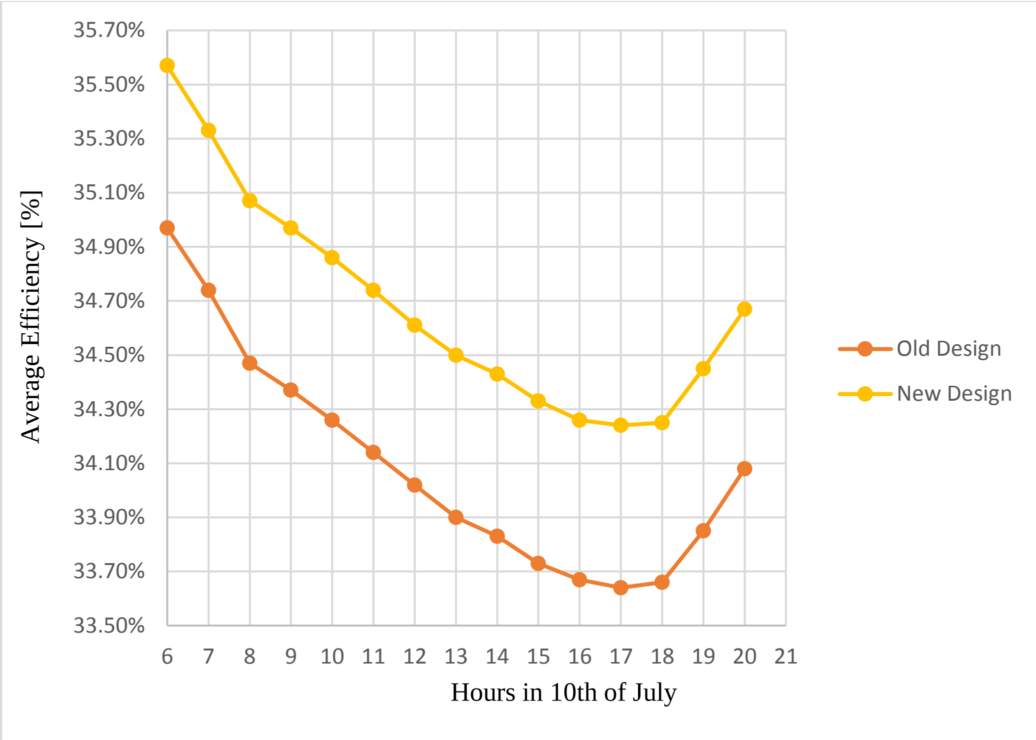

Figure 5.3. Average Efficiency in hours on 10th of July

Figures 5.4, 5.5 and 5.6, 5.7 compare the thermal energy and electric energy during 9th and 10th of July between the old and the new design. As shown in the figures 5.4, 5.5 and 5.6, 5.7 (the new design), the value of thermal and electric energy is respectively 0.141 and 0.062, which is more than that of the old design. Although this difference is not that significant, it should be noted that this comparison is only for a day. Figures 5.5 and 5.7, shows that at 1:00 o'clock in the afternoon, which is usually the hottest hour during the day, the electric and thermal energy has their highest values. In the new design, since the well water is heated a bit by circulating it around the glass wall, the value of electric and thermal energy is more than that of the old design. As stated earlier, when there is no sunshine the value of the electric and thermal energy will be zero.

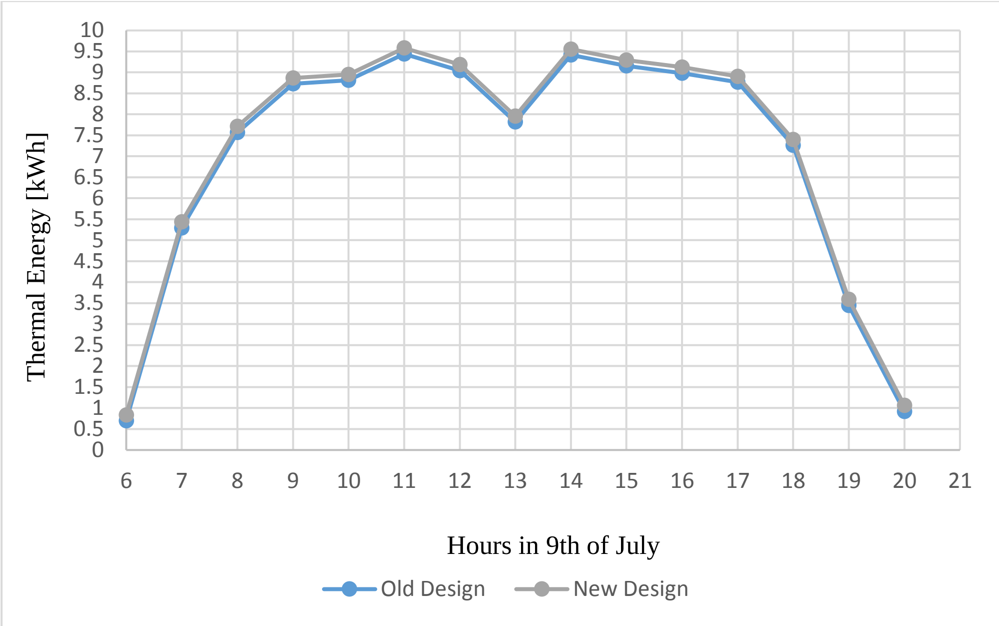

Figure 5.4. Thermal Energy in hours on 9th of July

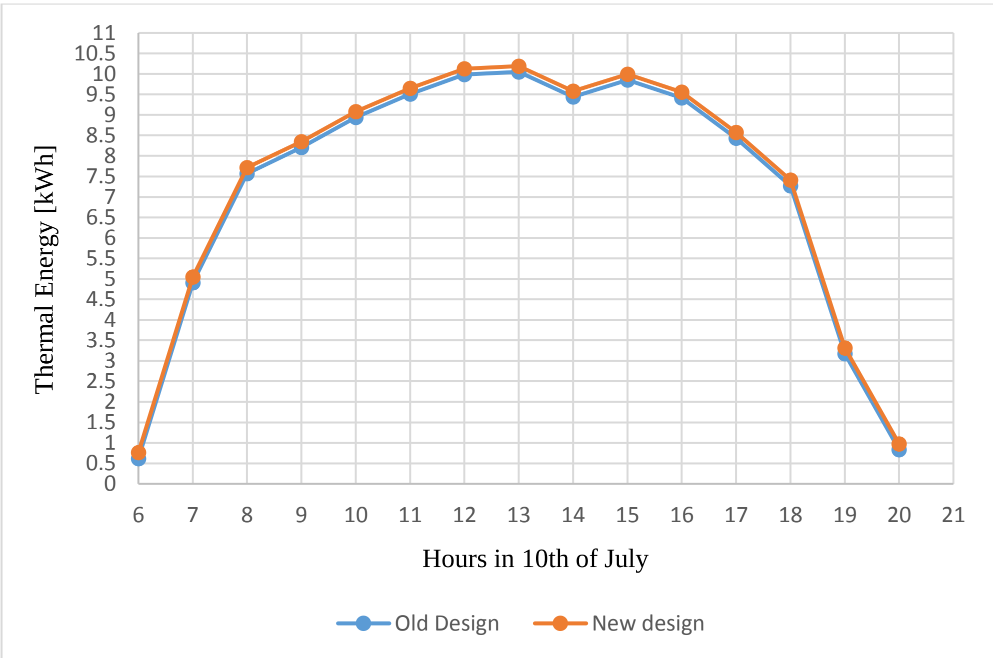

Figure 5.5. Thermal Energy in hours on 10th of July

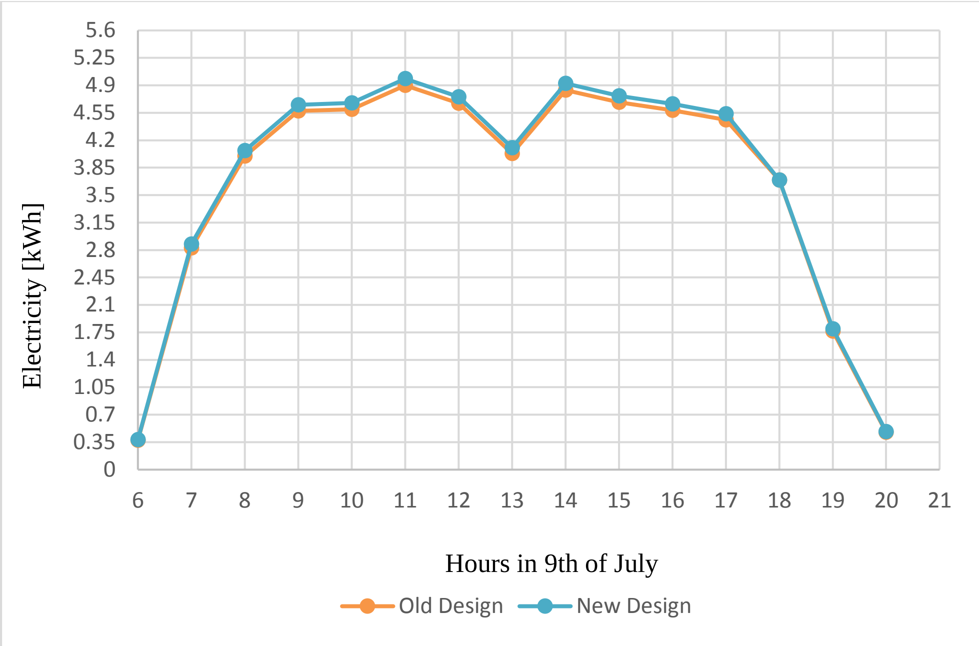

Figure 5.6. Electricity in hours on 9th of July

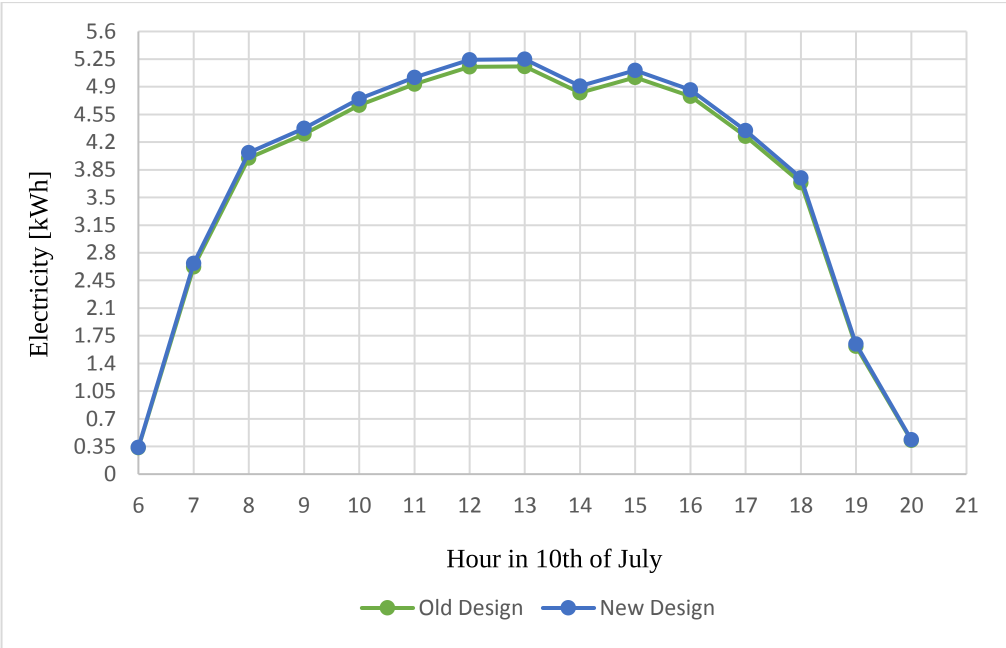

Figure 5.7. Electricity in hours on 10th of July

As figures 5.4, 5.5, 5.6 and 5.7 indicate, a drop, seen in the shape of a 'V' in each graph, shows the change in solar radiation within a specific time. And so, this illustrates the effect of solar radiation on an increasing thermal and electric energy.

Chapter 6

CONCLUSION

6.1 Conclusions

The LCPVs simulation assisted tremendously in enhancing the knowledge that surrounds the energy and environmental area of simulating concentrating photovoltaic system. There was a simulation that was carried out for the linear concentrating photovoltaic system which made use of a very vital fluid cooling channel system. For the computation of the output variables of the LCPVs under any given climatic and solar conditions, heat transfer, electrical circuits and fluid flow equations, as well as functions used in thermodynamics, where the basis on which the simulation ran. Numerous imputable parameters could be adjusted to a particular system within this simulation and consequently, the simulation of the LCPVs was an exceptional adaptable replica.

From these simulations, numerous valuable conclusions were obtained. An observation was made that with an increase in the flow rate of the cooling fluid, the effectiveness of the multi-junction cell also increased; this led to an equivalent increase in the output of electrical energy. In any case, where there was an increase in the flow rate, there was also an equivalent increase in the pumps load as well. Based on previous research works, the LCPVs from an energy and environmental point of view has been explicitly analyzed; this has given more details on the LCPVs, and it has notably aided research in the field of concentrating photovoltaic.

Now since the use of fossil fuels has led to the creation of energy that are not monetarily successful and has also resulted in some sort of endangerment to nature; consequently, there has been an expansion on the importance of exploration in the field of renewable energies; for example, sun oriented, wind and water resources. This study aimed at finding the optimum value of LCPVs that can be acquired from the flexibility of the simulations in the LCPVs; this was done by modifying the simulation as well as exciting changes in the data input. Also, the adjustments in effectiveness, the volume of the water created and the generated electricity were also investigated and compared. One of the objectives of this study was to focus on the changes that can be made within the scope of LCPVs which will result in the increase of total efficiency of the system.

6.2 Future Work

Since an adaptable simulation of the LCPVs has been created, numerous studies can therefore be conducted at a later period of time. Although, changes can be made to this simulation to dissect the diverse solar based photovoltaic and heat systems, a comparison between each simulation and the LCPVs can also made. This would provide a comprehensive system comparison under numerous conceivable input conditions. This research concentrated solely on the application of LCPVs within a residential-sized capacity, although simply alterations in the storage simulation and the systems size, multi-family residential buildings, commercial and industrial sectors that also consume electricity could also be carefully investigated. Likewise, the likelihood of supplanting the PV cell technologies with the multi-junction and scrutinizing the system from energy, economic and environmental perspective exists. Likewise, in light of the fact that the 3D concentrating Fresnel focal lenses would diminish the expense of the system, it would bear some significance to substitute the linear concentrating system and compare the two systems. Another investigation of great importance would be to improve the LCPVs of a given size and change its variables like the tank size, coefficient of heat transfer for the flow channel and its size will then be noteworthy.

Another vital aspect which can be included is the presentation of a financial report in the simulation. The monetary report can incorporate manufacturing, installation, repairs and maintenance costs. It could ascertain or help compute investment return for the model and it could be useful for decision making during the process of production.

REFERENCES

[1] Greene, L. E., & Sherif, R. A. (2009). *High and Low Concentrator Systems for Solar Electric Applications V*. publisher not identified.

[2] Kerzmann, T., & Schaefer, L. (2012). System simulation of a linear concentrating photovoltaic system with an active cooling system. Renewable Energy, 41, 254-261.

[3] Wang, Y. F., Li, M., Lin, W. X., Wang, L., Wei, S., Xu, Y., & Xiang, M. (2009). Performance analysis on solar cell modules of flat-plate and trough concentrating photovoltaic system. *Acta Optica Sinica*, *29*(8), 2287-2292.

[4] Othman, M. Y., Yatim, B., Sopian, K., & Bakar, M. N. A. (2007). Performance studies on a finned double- pass photovoltaic- thermal (PV/T) solar collector. *Desalination*, *209*(1), 43-49.

[5] Green, M. A., Emery, K., Hishikawa, Y., & Warta, W. (2010). Solar cell Efficiency tables (version 36). *Progress in Photovoltaics: Research and Applications* , *18*(5), 346-352.

[6] Borisenko, V. E. , & Ossicini, S. From Macroscopic Long-range Quantum Interference to Multiquantum Well. *What is in the Nanoworld: A Handbook on Nanoscience and Nanotechnology*, 175-195.

[7] Sonneveld, P. J., Swinkels, G. L. A. M., Kempkes, F., Campen, J. B., & Bot, G. P. A. (2006, April). Greenhouse with an integrated NIR filter and a solar cooling system. In *International Symposium on Greenhouse Cooling 719* (pp. 123-130).

[8] Robles-Ocampo, B., Ruiz-Vasquez, E., Canseco-Sanchez, H., Cornejo-Meza, R. C., Trápaga-Martínez, G., García-Rodriguez, F. J., & Vorobiev, Y. V. (2007). Photovoltaic / thermal solar hybrid system with bifacial PV module and transparent plane collector. *Solar Energy Materials and Solar Cells*, *91*(20), 1966-1971.

[9] Handbook, A. S. H. R. A. E. (2001). Fundamentals. *American Society of Heating, Refrigerating and Air Conditioning Engineers, Atlanta*, *111*.

[10] Gajbert, H. (2008). Solar thermal energy systems for building integration. *Licentiate Dissertation, University of Lund, Sweden*.

[11] Bett, A. W., Siefer, G., Baur, C., van Riesen, S., Peharz, G., Lerchenmüller, H., & Dimroth, F. (2005, June). FLATCON concentrator PV-technology ready for the market. In *Proceedings of the 20th European Photovoltaic Solar Energy Conference, Barcelona, Spain* (pp. 114-117).

[12] Cheknane, A., Benyoucef, B., Charles, J. P., Zerdoum, R., & Trari, M. (2005). Minimization of the effect of the collecting grid in a solar cell based silicon. *Solar energy materials and solar cells*, *87*(1), 557-565.

[13] Kandlikar, S. G., & Balasubramanian, P. (2004). An extension of the flow boiling correlation to transition, laminar, and deep laminar flows in minichannels and microchannels. *Heat Transfer Engineering*, *25*(3), 86-93.

[14] Natarajan, S. K., Mallick, T. K., Katz, M., & Weingaertner, S. (2011). Numerical investigations of solar cell temperature for photovoltaic concentrator system with and without passive cooling arrangements.*International journal of thermal sciences*, *50*(12), 2514-2521.

[15] Zhu, L., Boehm, R. F., Wang, Y., Halford, C., & Sun, Y. (2011). Water immersion cooling of PV cells in a high concentration system. *Solar Energy Materials and Solar Cells*, *95*(2), 538-545.

[16] Tonui, J. K., & Tripanagnostopoulos, Y. (2007). Air-cooled PV/T solar collectors with low cost performance improvements. *Solar energy*, *81*(4), 498-511.

[17] Bhattarai, S., Kafle, G. K., Euh, S. H., Oh, J. H., & Kim, D. H. (2013). Comparative study of photovoltaic and thermal solar systems with different storage capacities: performance evaluation and economic analysis. *Energy*,*61*, 272-282.

[18] Verlinden, P. J., Lewandowski, A., Bingham, C., Kinsey, G. S., Sherif, R. A., & Lasich, J. B. (2006, May). Performance and reliability of multi-junction III-V modules for concentrator dish and central receiver applications. In *2006 IEEE 4th World Conference on Photovoltaic Energy Conference* (Vol. 1, pp. 592-597).

[19] Wilcox, S. (2007). *National Solar Radiation Database 1991-2005 Update: User's Manual* (No. NREL/TP-581-41364). National Renewable Energy Laboratory (NREL), Golden, CO.

[20] Mokri, A., & Emziane, M. (2011, November). Performance-Based Analysis of a Double-Receiver Photovoltaic System. In *World Renewable Energy Congress-Sweden; 8-13 May; 2011; Linköping; Sweden* (No. 57, pp. 2946-2952). Linköping University Electronic Press.

[21] Chen, H., Ji, J., Wang, Y., Sun, W., Pei, G., & Yu, Z. (2014). Thermal analysis of a high concentration photovoltaic / thermal system. *Solar Energy*,*107*, 372-379.

[22] Sun, Y., Wang, Y., Zhu, L., Yin, B., Xiang, H., & Huang, Q. (2014). Direct liquid-immersion cooling of concentrator silicon solar cells in a linear concentrating photovoltaic receiver. *Energy*, *65*, 264-271.

[23] Rosell, J. I., Vallverdu, X., Lechon, M. A., & Ibanez, M. (2005). Design and simulation of a low concentrating photovoltaic / thermal system. *Energy Conversion and Management*, *46*(18), 3034-3046.

[24] Chen, K., & Gwilliam, S. B. (1996). An analysis of the heat transfer rate and efficiency of TE (thermoelectric) cooling systems. *International Journal of Energy Research*, *20*(5), 399-417.

[25] Simfukwe, J., Hatwaambo, S., & Kabumbwe, H. (2011, November). Using structured aluminum reflectors in flux scattering on module performance. In *World Renewable Energy Congress-Sweden; 8-13 May; 2011; Linköping; Sweden* (No. 057, pp. 2924-2929). Linköping University Electronic Press.

[26] Mittelman, G., Kribus, A., & Dayan, A. (2007). Solar cooling with concentrating photovoltaic / thermal (CPVT) systems. *Energy Conversion and Management*, *48*(9), 2481-2490.

APPENDICES

A.1 PAYTON Program – Definition

Definition

Constants = {'Efficiency':,

'Average Efficiency':,

'Thermal Conductivity':,

'Thermal Conductivity Of Fluid In Liquid State':,

'Dynamic Viscosity':,

'Kinematic Viscosity':,

'Density':,

'Density Of City Water':,

'Density Of Fluid In Gas State':,

'Density Of Fluid In Liquid State':,

'Density Of Fluid In Tank From Previous hr Iteration':,

'Cross Sectional Area Of Flow Channel':,

'Outside Surface Area Of Flow Channel':,

'Boiling Number':,

'Convection Number':,

'Specific Heat':,

'Hydraulic Diameter':,

'Thermal Energy Of City Water Flowing Into Storage Tank':, 'Thermal Energy Flowing From Channel Into Storage Tank':,

'Thermal Energy Leaving Storage Tank Through Conduction':,

'Thermal Energy Flowing From Storage Into Channel':,

'Thermal Energy In The Storage Tank':,

'Thermal Energy Leaving Storage Tank For Use':,

'Friction Factor':,

'Froude Number Of Fluid In Liquid State':,

'Mass Flux':,

'Enthalpy Of Fluid Bulk Flow':,

'Enthalpy Of Fluid Bulk Flow From Previous Chanel Segment':,

'Enthalpy Of The City Water':,

'Change In Enthalpy From Gas To Liquid State Fluid':,

'Enthalpy Of Fluid In Gas State':,

'Enthalpy Entering Fluid In Flow Channel':,

'Enthalpy Of Fluid In Liquid State':,

'Heat Transfer Coefficient':,

'Avg Heat Transfer Coefficient':,

'Heat Transfer Coefficient At Channel Segment':,

'Convective Boiling Dominant Heat Transfer Coefficient':,

'Heat Transfer Coefficient Of Fluid In Liquid State':, 'Convective Boiling Dominant Heat Transfer Coefficient':,

'Heat Transfer Coefficient Of Fluid In Liquid State':,

'Nucleate Boiling Dominant Heat Transfer Coefficient':,

'Enthalpy Of Fluid In Storage Tank':,

'Enthalpy Of Fluid In Storage Tank At Channel Segment':,

'Module Length':,

'Mass Flow Rate':,

'Mass Of Fluid In Storage Tank':,

'Nusselt Number':,

'Perimeter of Flow Channel Cross Section':,

'LCPV System Power':,

'Prandtl Number':,

'Heat Flux Entering Fluid From Solar Radiation':,

'Prandtl Number Of Fluid In Liquid State':,

'Solar Radiation':,

'Heat Entering Flow Channel':,

'Thermal Resistance Of Flow Channel In Solution':,

'Reynolds Number':,

'Reynolds Number Of Fluid In Liquid State':,

'Number Of Module Row In The LCPV Area':, 'Surface Area Of The Storage Tank':,

'Outdoor Temperature':,

'Temperature Of Bulk Fluid Flow In Channel':,

'Average Temperature Of Bulk Fluid Flow In Channel':,

'Temperature Of Bulk Fluid Flow In Channel Segment':,

'Indoor Air Temperature':,

'Average Channel Surface Temperature':,

'Temperature Of Fluid In Storage Tank':,

'Velocity Of Fluid In Liquid State':,

'Average Velocity Of Fluid In Channel':,

'Volume Of Fluid That Leaves System To Use':,

'Width Of Concentration Area':,

'Tank Energy From The Previous Hour Iteration':,

'Thermal Energy Mass ratio':}

A.2 Payton PROGRAM – Main Code

Main Code:

```
import Tkinter as Tk

from Tkinter import Label, Entry, Button, StringVar

from definitions import Constants

from math import sqrt, log

class App():

def __init__(self):

self.root = Tk.Tk()

self.root.geometry('800x800')

# Constants

# GUI variables

self.avgChnlTmp = StringVar()

self.heatTotal = StringVar()

self.heatFlux = StringVar()

self.hBulk = StringVar()

self.hHeat = StringVar()

self.htCBD = StringVar()

self.htNBD = StringVar()

self.convNum = StringVar()

self.htLIQUID = StringVar()

self.Bo = StringVar()

self.G = StringVar()

self.HFG = StringVar()
```

```
self.F = StringVar()
self.ReLiquid = StringVar()
self.Dh = StringVar()
self.Re = StringVar()
self.Nu = StringVar()
self.Pr = StringVar()
self.Ht = StringVar()
self.Tsurface = StringVar()
self.HtAvg = StringVar()
self.HtEnt = StringVar()
self.TbulkAvg = StringVar()
self.TbulkEnt = StringVar()
self.Ncell = StringVar()
self.Pcell = StringVar()
self.TimeEt = StringVar()
self.Eloss = StringVar()
self.Eout = StringVar()
self.Euse = StringVar()
self.Ecitywater = StringVar()
self.Ein = StringVar()
self.Etank = StringVar()
self.Utank = StringVar()
# GUI Initializations
self.TimeEt.set('TIME')
# labels
```

```
label01_a = Label(self.root, text="Average Channel Surface Temperature(K): ")
label01_b = Label(self.root, textvariable=self.avgChnlTmp)
label02_a = Label(self.root, text='Heat Entering Flow Channel')
label02_b = Label(self.root, textvariable=self.heatTotal)
label03_a = Label(self.root, text='Heat Flux Entering The Fluid From Solar
Radiation')
label03_b = Label(self.root, textvariable=self.heatFlux)
label04_a = Label(self.root, text='Enthalpy Of Fluid Bulk Flow')
label04_b = Label(self.root, textvariable=self.hBulk)
label05_a = Label(self.root, text='Enthalpy Entering Fluid In Flow Channel')
label05_b = Label(self.root, textvariable=self.hHeat)
label06_a = Label(self.root, text='Convection Boiling Dominant Heat Transfer
Coefficient')
label06_b = Label(self.root, textvariable=self.htCBD)
label07_a = Label(self.root, text='Nucleate Boiling Dominant Heat Transfer
Coefficient')
label07_b = Label(self.root, textvariable=self.htNBD)
label08_a = Label(self.root, text='Convection Number')
label08_b = Label(self.root, textvariable=self.convNum)
label09_a = Label(self.root, text='Heat Transfer Coefficient of Fluid In Liquid state')
label09_b = Label(self.root, textvariable=self.htLIQUID)
label010_a = Label(self.root, text='Boiling Number')
label010_b = Label(self.root, textvariable=self.Bo)
label011_a = Label(self.root, text='Mass Flux')
label011_b = Label(self.root, textvariable=self.G)
```

```
label012_a = Label(self.root, text='Change In Enthalpy From Gas To Liquid')
label012_b = Label(self.root, textvariable=self.HFG)

label013_a = Label(self.root, text='Friction Factor')
label013_b = Label(self.root, textvariable=self.F)
label014_a = Label(self.root, text='Reynolds Number Of FLuid In liquid State')
label014_b = Label(self.root, textvariable=self.ReLiquid)
label015_a = Label(self.root, text='Hydraulic Diameter')
label015_b = Label(self.root, textvariable=self.Dh)
label016_a = Label(self.root, text='Reynolds Number')
label016_b = Label(self.root, textvariable=self.Re)
label017_a = Label(self.root, text='Nusselt Number')
label017_b = Label(self.root, textvariable=self.Nu)
label018_a = Label(self.root, text='Prandtl Number')
label018_b = Label(self.root, textvariable=self.Pr)
label019_a = Label(self.root, text='Heat Transfer Coefficient')
label019_b = Label(self.root, textvariable=self.Ht)
label020_a = Label(self.root, text='Average Channel Surface Tempreture')
label020_b = Label(self.root, textvariable=self.Tsurface)

label013_a = Label(self.root, text='Friction Factor')
label021_a = Label(self.root, text='Average Heat Transfer Coefficient')
label021_b = Label(self.root, textvariable=self.HtAvg)
entry = Entry(self.root, textvariable=self.HtEnt)
label022_a = Label(self.root, text='Average Temperature Of Bulk Fluid Flow in
Channel')
label022_b = Label(self.root, textvariable=self.TbulkAvg)
```

```
entry1 = Entry(self.root, textvariable=self.TbulkEnt)
label023_a = Label(self.root, text='Average Efficiency')
label023_b = Label(self.root, textvariable=self.Ncell)
label024_a = Label(self.root, text='LCPV System Power')
label024_b = Label(self.root, textvariable=self.Pcell)
label025_a = Label(self.root, text='Thermal Energy Leaving Storage Tank Through
conduction')
label025_b = Label(self.root, textvariable=self.Eloss)
entry2 = Entry(self.root, textvariable=self.TimeEt)
label026_a = Label(self.root, text='Thermal Energy Flowing From Tank To
Channel')
label026_b = Label(self.root, textvariable=self.Eout)
label027_a = Label(self.root, text='Thermal Energy Leaving Tank For use')
label027_b = Label(self.root, textvariable=self.Euse)
label028_a = Label(self.root, text='Thermal Energy Of City Water Flowing Into
Tank')
label028_b = Label(self.root, textvariable=self.Ecitywater)
label029_a = Label(self.root, text='Thermal Energy Flowing From Channel Into
Tank')
label029_b = Label(self.root, textvariable=self.Ein)
label030_a = Label(self.root, text='Thermal Energy In Tank')
label030_b = Label(self.root, textvariable=self.Etank)
label031_a = Label(self.root, text='Thermal Energy / Mass Tank')
label031_b = Label(self.root, textvariable=self.Utank)
# buttons
```

```
button01 = Button(self.root, text="Calculate", command=self.calcTsurface)

button02 = Button(self.root, text='Calculate', command=self.calcHeatTotal)

button03 = Button(self.root, text='Calculate', command=self.calcHeatFluxEntering)

button04 = Button(self.root, text='Calculate', command=self.calcEnthalpyOfBulk)

button05 = Button(self.root, text='Calculate',

command=self.calcEnthalpyEnteringFlowChannel)

button06 = Button(self.root, text='Calculate', command=self.calcConvectiveBoiling)

button07 = Button(self.root, text='Calculate', command=self.calcNucleatBoiling)

button08 = Button(self.root, text='Calculate', command=self.calcConvectionNum)

button09 = Button(self.root, text='Calculate', command=self.calcHT_LIQUID)

button010 = Button(self.root, text='Calculate', command=self.calcBo)

button011 = Button(self.root, text='Calculate', command=self.calcMassFlux)

button012 = Button(self.root, text='Calculate', command=self.calcHFG)

button013 = Button(self.root, text='Calculate', command=self.calcFrictionFactor)

button014 = Button(self.root, text='Calculate', command=self.calcReLiquid)

button015 = Button(self.root, text='Calculate', command=self.calcDH)

button016 = Button(self.root, text='Calculate', command=self.calcRe)

button017 = Button(self.root, text='Calculate', command=self.calcNu)

button018 = Button(self.root, text='Calculate', command=self.calcPr)

button019 = Button(self.root, text='Calculate', command=self.calcHt)

button020 = Button(self.root, text='Calculate', command=self.calcTSurface_)

button021 = Button(self.root, text='Calculate', command=self.calcHtAvg)

button022 = Button(self.root, text='Calculate', command=self.calcTBulkAvg)

button023 = Button(self.root, text='Calculate', command=self.calcAvgEfficiency)

button024 = Button(self.root, text='Calculate', command=self.calcLCPV)
```

```
button025 = Button(self.root, text='Calculate', command=self.calcEloss)
button026 = Button(self.root, text='Calculate', command=self.calcEout)
button027 = Button(self.root, text='Calculate', command=self.calcEuse)
button028 = Button(self.root, text='Calculate', command=self.calcEcityWater)
button029 = Button(self.root, text='Calculate', command=self.calcEin)
button030 = Button(self.root, text='Calculate', command=self.calcEtank)
button031 = Button(self.root, text='Calculate', command=self.calcUtank)
#entries
label01_a.grid(row=1, column=1)
label01_b.grid(row=1, column=2)
button01.grid(row=1, column=3)
label02_a.grid(row=2, column=1)
label02_b.grid(row=2, column=2)
button02.grid(row=2, column=3)
label03_a.grid(row=3, column=1)
label03_b.grid(row=3, column=2)
button03.grid(row=3, column=3)
label04_a.grid(row=4, column=1)
label04_b.grid(row=4, column=2)
button04.grid(row=4, column=3)
label05_a.grid(row=5, column=1)
label05_b.grid(row=5, column=2)
button05.grid(row=5, column=3)
label06_a.grid(row=6, column=1)
label06_b.grid(row=6, column=2)
```

```
button06.grid(row=6, column=3)
label07_a.grid(row=7, column=1)
label07_b.grid(row=7, column=2)
button07.grid(row=7, column=3)
label08_a.grid(row=8, column=1)
label08_b.grid(row=8, column=2)
button08.grid(row=8, column=3)
label09_a.grid(row=9, column=1)
label09_b.grid(row=9, column=2)
button09.grid(row=9, column=3)
label010_a.grid(row=10, column=1)
label010_b.grid(row=10, column=2)
button010.grid(row=10, column=3)
label011_a.grid(row=11, column=1)
label011_b.grid(row=11, column=2)
button011.grid(row=11, column=3)
label012_a.grid(row=12, column=1)
label012_b.grid(row=12, column=2)
button012.grid(row=12, column=3)
label013_a.grid(row=13, column=1)
label013_b.grid(row=13, column=2)
button013.grid(row=13, column=3)
label014_a.grid(row=14, column=1)
label014_b.grid(row=14, column=2)
button014.grid(row=14, column=3)
```

```
label015_a.grid(row=15, column=1)
label015_b.grid(row=15, column=2)
button015.grid(row=15, column=3)
label016_a.grid(row=16, column=1)
label016_b.grid(row=16, column=2)
button016.grid(row=16, column=3)
label017_a.grid(row=17, column=1)
label017_b.grid(row=17, column=2)
button017.grid(row=17, column=3)
label018_a.grid(row=18, column=1)
label018_b.grid(row=18, column=2)
button018.grid(row=18, column=3)
label019_a.grid(row=19, column=1)
label019_b.grid(row=19, column=2)
button019.grid(row=19, column=3)
label020_a.grid(row=20, column=1)
label020_b.grid(row=20, column=2)
button020.grid(row=20, column=3)
label021_a.grid(row=21, column=1)
label021_b.grid(row=21, column=2)
button021.grid(row=21, column=3)
entry.grid(row=21, column=4)
label022_a.grid(row=22, column=1)
label022_b.grid(row=22, column=2)
button022.grid(row=22, column=3)
```

```
entry1.grid(row=22, column=4)

label023_a.grid(row=23, column=1)

label023_b.grid(row=23, column=2)

button023.grid(row=23, column=3)

label024_a.grid(row=24, column=1)

label024_b.grid(row=24, column=2)

button024.grid(row=24, column=3)

label025_a.grid(row=25, column=1)

label025_b.grid(row=25, column=2)

button025.grid(row=25, column=3)

entry2.grid(row=25, column=4)

label026_a.grid(row=26, column=1)

label026_b.grid(row=26, column=2)

button026.grid(row=26, column=3)

label027_a.grid(row=1, column=4)

label027_b.grid(row=1, column=5)

button027.grid(row=1, column=6)

label028_a.grid(row=2, column=4)

label028_b.grid(row=2, column=5)

button028.grid(row=2, column=6)

label029_a.grid(row=3, column=4)

label029_b.grid(row=3, column=5)

button029.grid(row=3, column=6)

label030_a.grid(row=4, column=4)

label030_b.grid(row=4, column=5)
```

```
button030.grid(row=4, column=6)
label031_a.grid(row=5, column=4)
label031_b.grid(row=5, column=5)
button031.grid(row=5, column=6)
self.root.mainloop()
def calcTsurface(self):
# Tsurface = (.00365+293.15*0.0006-Un)/0.0006
ans = 0.00365 + 293.15*0.0006-Constants['Average Efficiency']
ans /= 0.0006
self.avgChnlTmp.set(str(ans))
def calcHeatTotal(self):
# Heat Entering Flow Channel * Aconcentration - Rchannel*Asurface(Tbulk-Tair)
Heat_Entering_Flow_Channel = Constants['Heat Flux Entering Fluid From Solar
Radiation']
Rchannel = Constants['Thermal Resistance Of Flow Channel In Solution']
Asurface = Constants['Outside Surface Area Of Flow Channel']
Tbulk = Constants['Average Temperature Of Bulk Fluid Flow In Channel']
Tair = Constants['Outdoor Temperature']
Aconcentration = 0.0 #SimpleDialog(self.root, 'A Concentrator:')
ans = Heat_Entering_Flow_Channel*Aconcentration-Rchannel*Asurface*(Tbulk-
Tair)
Constants['Heat Entering Flow Channel'] = ans
self.heatTotal.set(str(ans))
def calcHeatFluxEntering(self):
# Solar Radiation x (1  - Efficiency)x 0.085 x 80
```

```
SolarRadiation = Constants['Solar Radiation']

Efficiency = Constants['Efficiency']

ans = SolarRadiation*(1 - Efficiency) * 0.085 * 80

Constants['Heat Flux Entering Fluid From Solar Radiation'] = ans

self.heatFlux.set(str(ans))

def calcEnthalpyOfBulk(self):

# Hbulki-1 + Hheat

Hbulki_1 = Constants['Enthalpy Of Fluid Bulk Flow From Previous Chanel

Segment']

Hheat = Constants['Enthalpy Entering Fluid In Flow Channel']

ans = Hbulki_1 + Hheat

Constants['Enthalpy Of Fluid Bulk Flow'] = ans

self.hBulk.set(str(ans))

def calcEnthalpyEnteringFlowChannel(self):

# qTotal / m'

qTotal = Constants['Heat Entering Flow Channel']

m = Constants['Mass Flow Rate']

Constants['Enthalpy Entering Fluid In Flow Channel'] = qTotal/m

self.hHeat.set(qTotal/m)

def calcConvectiveBoiling(self):

# 1.136 x co^-0.9 x (1-x)^0.8 x F x Fliquid x hliquid + 667.2*Bo^0.7 x (1-x)^0.8 x

hliquid

# ans = 1.136*pow(Co, -0.9)*pow(1-x, 0.8) * f * hLiquid + 667.2*pow(Bo, 0.7) *

pow(1-x, 0.8)*hLiquid

Constants['Convective Boiling Dominant Heat Transfer Coefficient'] =
```

```
self.calcHt_(1.136, 667.2, 0.9)
self.htCBD.set(str(Constants['Convective Boiling Dominant Heat Transfer
Coefficient']))
def calcNucleatBoiling(self):
Constants['Nucleate Boiling Dominant Heat Transfer Coefficient'] =
self.calcHt_(0.6683, 1058, 0.2)
self.htNBD.set(Constants['Nucleate Boiling Dominant Heat Transfer Coefficient'])
def calcHt_(self, x1, x2, p):
hLiquid = Constants['Heat Transfer Coefficient Of Fluid In Liquid State']
frLiquid = Constants['Froude Number Of Fluid In Liquid State']
x = 1 # Equality
f = Constants['Friction Factor']
Bo = Constants['Boiling Number']
Co = Constants['Convection Number']
return x1*pow(Co, -p)*pow(1-x, 0.8) * f * hLiquid + x2*pow(Bo, 0.7) * pow(1-x,
0.8)*hLiquid
def calcConvectionNum(self):
# ((Pgas/Pliquid)^0.5) x ((1-x/x)^0.8)
Pgas = Constants['Density Of Fluid In Gas State']
Pliquid = Constants['Density Of Fluid In Liquid State']
x = 1
ans = pow(Pgas/Pliquid, 0.5) * pow((1-x)/x, 0.8)
Constants['Convection Number'] = ans
self.convNum.set(ans)
def calcHT_LIQUID(self):
```

```
# (Reliquid - 1000)xPrliquidx(f/2)x(Kliquid/Dh)   /  1+12.7((Prliquid^(2/3)-
1)x(f/2)^0.5
Reliquid = Constants['Reynolds Number Of Fluid In Liquid State']
Prliquid = Constants['Prandtl Number Of Fluid In Liquid State']
Kliquid = Constants['Thermal Conductivity Of Fluid In Liquid State']
f = Constants['Friction Factor']
Dh = Constants['Hydraulic Diameter']
ans = (Reliquid-1000) * Prliquid * (f/2) * (Kliquid/Dh)
ans /= (1 + 12.7 *((pow(Prliquid, 2/3) - 1) * pow(f/2, 0.5)))
Constants['Heat Transfer Coefficient Of Fluid In Liquid State'] = ans
self.htLIQUID.set(ans)
def calcBo(self):
# Qtotal/ ( GxHfg)
Qtotal = Constants['Heat Entering Flow Channel']
G = Constants['Mass Flux']
Hfg = Constants['Change In Enthalpy From Gas To Liquid State Fluid']
ans = Qtotal/(G*Hfg)
Constants['Boiling Number'] = ans
self.Bo.set(ans)
def calcMassFlux(self):
# p x Uliquid
p = Constants['Density']
Uliquid = Constants['Velocity Of Fluid In Liquid State']
ans = p * Uliquid
```

```
Constants['Mass Flux'] = ans
self.G.set(ans)
def calcHFG(self):
# Hgas - Hliquid
Hgas = Constants['Enthalpy Of Fluid In Gas State']
Hliquid = Constants['Enthalpy Of Fluid In Liquid State']
ans = Hgas - Hliquid
Constants['Change In Enthalpy From Gas To Liquid State Fluid'] = ans
self.HFG.set(ans)
def calcFrictionFactor(self):
# sqrt(1.58*ln(Reliquid)-3.28
Reliquid = Constants['Reynolds Number Of Fluid In Liquid State']
ans = sqrt(1.58 * log(Reliquid) - 3.28)
Constants['Friction Factor'] = ans
self.F.set(ans)
def calcReLiquid(self):
# Uliquid*Dh / V
Uliquid = Constants['Velocity Of Fluid In Liquid State']
Dh = Constants['Hydraulic Diameter']
v = Constants['Kinematic Viscosity']
ans = Uliquid*Dh/v
Constants['Reynolds Number Of Fluid In Liquid State'] = ans
self.ReLiquid.set(ans)
def calcDH(self):
# 4 x AcrossSection  / p
```

```
Across = Constants['Cross Sectional Area Of Flow Channel']

p = Constants['Perimeter of Flow Channel Cross Section']

ans = 4*Across/p

Constants['Hydraulic Diameter'] = ans

self.Dh.set(ans)

def calcRe(self):

# Um x Dh / v

Um = Constants['Average Velocity Of Fluid In Channel']

Dh = Constants['Hydraulic Diameter']

v = Constants['Kinematic Viscosity']

ans = Um*Dh/v

Constants['Reynolds Number'] = ans

self.Re.set(ans)

def calcNu(self):

# 0.023 * Re^0.8 * Pr^0.4

Re = Constants['Reynolds Number']

Pr = Constants['Prandtl Number']

ans = 0.023 * pow(Re, 0.8) * pow(Pr, 0.4)

Constants['Nusselt Number'] = ans

self.Nu.set(ans)

def calcPr(self):

# Cp x M / K

Cp = Constants['Specific Heat']

M = Constants['Dynamic Viscosity']

K = Constants['Thermal Conductivity']
```

```
ans = Cp*M/K

Constants['Prandtl Number'] = ans

self.Pr.set(ans)

def calcHt(self):

# Nu x k / Dh

Nu = Constants['Nusselt Number']

k = Constants['Thermal Conductivity']

Dh = Constants['Hydraulic Diameter']

ans = Nu*k/Dh

Constants['Heat Transfer Coefficient'] = ans

self.Ht.set(ans)

def calcTSurface_(self):

# Tbulk + Qtotal/Ht

Tbulk = Constants['Average Temperature Of Bulk Fluid Flow In Channel']

Qtotal = Constants['Heat Entering Flow Channel']

Ht = Constants['Avg Heat Transfer Coefficient']

ans = Tbulk+Qtotal/Ht

Constants['Average Channel Surface Temperature'] = ans

self.Tsurface.set(ans)

def calcHtAvg(self):

ans = self.calcAvg(self.HtEnt.get())

Constants['Avg Heat Transfer Coefficient'] = ans

self.HtAvg.set(str(ans))

def calcTBulkAvg(self):

ans = self.calcAvg(self.TbulkEnt.get())
```

```
Constants['Average Temperature Of Bulk Fluid Flow In Channel'] = ans
self.TbulkAvg.set(str(ans))
def calcAvg(self, lst):
lst = lst.split(',')
lst = map(float, lst)
ans = sum(lst)/len(lst)
return ans
def calcAvgEfficiency(self):
# .365 - (Tsurface - 293.15)*.06
Tsurface = Constants['Average Channel Surface Temperature']
ans = 0.365 - (Tsurface - 293.15)*0.0006
Constants['Average Efficiency'] = ans
self.Ncell.set(ans)
def calcLCPV(self):
# Ncell*Rows*Length*width concentration * qrod* 80 * 0.85
Ncell = Constants['Average Efficiency']
Rows = Constants['Number Of Module Row In The LCPV Area']
widthC = Constants['Width Of Concentration Area']
length = Constants['Module Length']
qrod = Constants['Solar Radiation']
ans = Ncell*Rows*length*widthC*qrod*80*0.85
Constants['LCPV System Power'] = ans
self.Pcell.set(ans)

def calcEloss(self):
# surfaceAreaTank * Rtank * (Ttank - Troom) * Time
```

```
surfaceAreaTnk = Constants['Surface Area Of The Storage Tank']
Rtank = Constants['Thermal Resistance Of Flow Channel In Solution']
Ttank = Constants['Temperature Of Fluid In Storage Tank']
Troom = Constants['Indoor Air Temperature']
Time = int(self.TimeEt.get())
ans = surfaceAreaTnk * Rtank * (Ttank - Troom) * Time
Constants['Thermal Energy Leaving Storage Tank Through Conduction'] = ans
self.Eloss.set(ans)
def calcEout(self):
# hTanki-1 * m' * Time
htanki = Constants['Enthalpy Of Fluid In Storage Tank At Channel Segment']
m = Constants['Mass Flow Rate']
time = int(self.TimeEt.get())
ans = htanki * m * time
Constants['Thermal Energy Flowing From Storage Into Channel'] = ans
self.Eout.set(ans)
def calcEuse(self):
# Vuse * Ptanki-1 * htanki-1
Vuse = Constants['Volume Of Fluid That Leaves System To Use']
PTnk = Constants['Density Of Fluid In Tank From Previous hr Iteration']
Htnk = Constants['Enthalpy Of Fluid In Storage Tank At Channel Segment']
ans = Vuse * PTnk * Htnk
Constants['Thermal Energy Leaving Storage Tank For Use'] = ans
self.Euse.set(ans)
def calcEcityWater(self):
```

```
# Vuse * Pcitywater * hcityWater
Vuse = Constants['Volume Of Fluid That Leaves System To Use']
Pcity = Constants['Density Of City Water']
Hcity = Constants['Enthalpy Of The City Water']
ans = Vuse * Pcity * Hcity
Constants['Thermal Energy Of City Water Flowing Into Storage Tank'] = ans
self.Ecitywater.set(ans)
def calcEin(self):
# Hbulk * m' * Time
Hbulk = Constants['Enthalpy Of Fluid Bulk Flow From Previous Chanel Segment']
m = Constants['Mass Flow Rate']
Time = int(self.TimeEt.get())
ans = Hbulk * m * Time
Constants['Thermal Energy Flowing From Channel Into Storage Tank'] = ans
self.Ein.set(ans)
def calcEtank(self):
# Etanki-1 + Ein + EcityWater - Euse - Eout - Eloss
Etnkprev = Constants['Tank Energy From The Previous Hour Iteration']
Ein = Constants['Thermal Energy Flowing From Channel Into Storage Tank']
Ecitywater = Constants['Thermal Energy Of City Water Flowing Into Storage Tank']
Euse = Constants['Thermal Energy Leaving Storage Tank Through Conduction']
Eout = Constants['Thermal Energy Flowing From Storage Into Channel']
Eloss = Constants['Thermal Energy Leaving Storage Tank Through Conduction']
ans = Etnkprev + Ein + Ecitywater - Euse - Eout - Eloss
Constants['Thermal Energy In The Storage Tank'] = ans
```

```
self.Etank.set(ans)

def calcUtank(self):

# Etank / Masstank

Etank = Constants['Thermal Energy In The Storage Tank']

Masstnk = Constants['Mass Of Fluid In Storage Tank']

ans = Etank/Masstnk

Constants['Thermal Energy Mass ratio'] = ans

self.Utank.set(ans)

if __name__ == '__main__':

App()
```

APPENDIX.3 LCPV SIMULATION OUTCOMES

Table 5. APPENDIX 3.1. A. LCPV Simulation Old Design in 9^{th} of July [13]

Hour	Efficiency %	Electricity (kWh)	E save (kj)	E tank (kj)	Thermal Energy (kWh)	Radiation (kW/m^2)	T air (K)	T bulk (K)	T bulk average (K)	T surface (K)	T tank (K)	V Loss (m^2)
6:00 9^{th} of July	34.99	0.3747	0	71045	0.6955	0.063	302.6	318.4	318.1	318.4	318.0	0.02227
7:00 9^{th} of July	34.74	2.829	0	74758	5.294	0.479	303.8	323.1	320.6	322.5	320.4	0.02227
8:00 9^{th} of July	34.48	3.998	3621	76481	7.57	0.682	305.4	327.6	324.0	326.8	321.5	0.02227
9:00 9^{th} of July	34.36	4.573	3722	78931	8.727	0.783	307.1	329.8	325.7	328.9	323.0	0.02227
10:00 9^{th} of July	34.26	4.595	3775	81292	8.811	0.789	308.2	331.4	327.3	330.5	324.5	0.02227
11:00 9^{th} of July	34.14	4.899	3775	83960	9.44	0.844	309.9	333.5	329.1	332.5	326.2	0.02227
12:00 9^{th} of July	34.06	4.673	3775	86176	9.043	0.807	311.0	334.8	330.6	333.8	327.6	0.02227
13:00 9^{th} of July	34.04	4.033	3775	87382	7.82	0.697	313.8	335.0	331.5	334.2	328.4	0.02227
14:00 9^{th} of July	33.92	4.838	3775	89659	9.415	0.839	313.8	337.3	333.0	336.2	329.8	0.02227
15:00 9^{th} of July	33.84	4.683	3775	91608	9.153	0.814	316.0	338.5	334.3	337.4	331.1	0.02227
16:00 9^{th} of July	33.78	4.583	3775	93315	8.983	0.798	314.3	339.6	335.5	338.5	332.1	0.02227
17:00 9^{th} of July	33.73	4.461	3775	94761	8.766	0.778	314.9	340.4	336.4	339.4	333.0	0.02227
18:00 9^{th} of July	33.75	3.694	3775	95036	7.262	0.644	314.3	339.9	336.6	339.1	333.2	0.02227
19:00 9^{th} of July	33.92	1.765	3775	92548	3.449	0.306	314.3	336.5	335.0	336.2	331.6	0.02227
20:00 9^{th} of July	34.14	0.4759	3775	88393	0.9209	0.082	312.1	332.5	332.5	332.5	329.0	0.02227

Table 6. APPENDIX 3.2. A. LCPV Simulation Old Design in 10th of July [13]

Hour	Efficiency %	Electricity (kWh)	E save (kj)	E tank (kj)	Thermal Energy (kWh)	Radiation (kW/m^2)	T air (K)	T bulk (K)	T bulk average (K)	T surface (K)	T tank (K)	V Loss (m^2)
6:00 10th of July	34.97	0.3329	0	71597	0.6184	0.056	301.5	318.8	318.5	318.7	318.4	0.02227
7:00 10th of July	34.74	2.622	0	75031	4.908	0.444	302.1	323.0	320.7	322.6	320.6	0.02227
8:00 10th of July	34.47	3.997	3637	76737	7.57	0.682	303.8	327.7	324.2	327.0	321.6	0.02227
9:00 10th of July	34.37	4.301	3737	78795	8.204	0.736	306.0	329.4	325.7	328.6	322.9	0.02227
10:00 10th of July	34.26	4.665	3775	81259	8.943	0.801	307.6	331.4	327.3	330.5	324.5	0.02227
11:00 10th of July	34.14	4.933	3775	83973	9.507	0.85	309.9	333.5	329.1	332.5	326.2	0.02227
12:00 10th of July	34.02	5.152	3775	86874	9.985	0.891	312.1	335.7	331.1	334.6	328.1	0.02227
13:00 10th of July	33.90	5.158	3775	89639	10.05	0.895	312.6	337.6	333.0	336.4	329.8	0.02227
14:00 10th of July	33.83	4.825	3775	91793	9.437	0.839	313.2	338.8	334.4	337.6	331.2	0.02227
15:00 10th of July	33.73	5.017	3775	94115	9.852	0.875	314.3	340.5	336.0	339.3	332.6	0.02227
16:00 10th of July	33.67	4.779	3775	95980	9.417	0.835	315.4	341.6	337.3	340.4	333.8	0.02227
17:00 10th of July	33.64	4.272	3775	97023	8.432	0.747	314.3	341.8	337.0	340.8	334.5	0.02227
18:00 10th of July	33.66	3.685	3775	97165	7.271	0.644	314.3	341.4	338.1	340.5	334.6	0.02227
19:00 10th of July	33.85	1.617	3775	94347	3.171	0.281	314.3	337.6	336.2	337.3	332.8	0.02227
20:00 10th of July	34.08	0.4287	3775	90017	0.8318	0.074	312.6	333.6	333.3	333.6	330.0	0.02227

Table 7. APPENDIX 3. 1. B. LCPV Simulation New Design in 9th of July

Hour	Efficiency %	Electricity (kWh)	Hot water (Lit/h)	Clean water (Lit/h)	Thermal Energy (kWh)	Radiation (kW/m^2) [13]	Pump1 Flow rate (Lit/h)	Pump2 Flow rate (Lit/h)	T air (K) [13]	T surface (K) [13]	T hot water tank (K)
6:00 9th of July	35.58	0.3810	30.07	3.12	0.8363	0.063	30	3	302.6	318.4	323.0
7:00 9th of July	35.34	2.877	200.3	23.74	5.435	0.479	200	23	303.8	329.5	328.4
8:00 9th of July	35.08	4.067	285.2	33.81	7.711	0.682	280	33	305.4	338.8	328.5
9:00 9th of July	34.95	4.652	327.4	38.81	8.868	0.783	330	39	307.1	345.9	328.0
10:00 9th of July	34.86	4.676	329.95	39.11	8.952	0.789	330	39	308.2	346.5	328.5
11:00 9th of July	34.74	4.984	352.95	41.84	9.581	0.844	350	42	309.9	363.5	328.2
12:00 9th of July	34.66	4.755	295.25	40.0	9.184	0.807	300	40	311.0	357.8	333.6
13:00 9th of July	34.64	4.104	255.0	34.55	7.961	0.697	250	34	313.8	339.2	333.4
14:00 9th of July	34.52	4.923	307	41.59	9.556	0.839	300	41	313.8	360.2	333.8
15:00 9th of July	34.44	4.765	297.85	40.35	9.294	0.814	290	40	316.0	359.4	333.1
16:00 9th of July	34.38	4.664	292	39.55	9.124	0.798	290	39	314.3	348.5	333.1
17:00 9th of July	34.32	4.539	284.65	38.56	8.907	0.778	280	38	314.9	343.4	333.0
18:00 9th of July	34.34	3.694	235.6	31.92	7.403	0.644	230	31	314.3	335.1	333.2
19:00 9th of July	34.52	1.795	116.45	15.17	3.59	0.306	120	15	314.3	326.2	333.6
20:00 9th of July	34.74	0.4843	30.0	4.06	1.062	0.082	30	41	312.1	321.5	333.0

Table 8. APPENDIX 3. 2. B. LCPV Simulation New Design in 10th of July

Hour	Efficiency %	Electricity (kWh)	Hot water (Lit/h)	Clean water (Lit/h)	Thermal Energy (kWh)	Radiation (kW/m^2) [13]	Pump1 Flow rate (Lit/h)	Pump2 Flow rate (Lit/h)	T air (K) [13]	T surface (K) [13]	T Hot water tank (K)
6:00 10th of July	35.57	0.3386	27.3	2.77	0.7594	0.056	27	3	301.5	318.7	323.4
7:00 10th of July	35.33	2.666	185.65	22.01	5.049	0.444	180	22	302.1	327.6	328.6
8:00 10th of July	35.07	4.066	285.2	33.80	7.717	0.682	280	34	303.8	338.0	328.6
9:00 10th of July	34.97	4.375	307.75	36.48	8.345	0.736	310	36	306.0	328.6	328.9
10:00 10th of July	34.86	4.747	334.95	39.70	9.084	0.801	340	39	307.6	330.5	328.5
11:00 10th of July	34.74	5.019	355.45	42.13	9.648	0.85	350	42	309.9	332.5	328.2
12:00 10th of July	34.61	5.242	326	44.17	10.126	0.891	330	44	312.1	334.6	333.1
13:00 10th of July	34.50	5.249	327.45	44.36	10.191	0.895	330	44	312.6	336.4	333.8
14:00 10th of July	34.43	4.911	307	41.59	9.578	0.839	300	42	313.2	337.6	333.2
15:00 10th of July	34.33	5.106	320.15	43.37	9.993	0.875	320	43	314.3	339.3	333.6
16:00 10th of July	34.26	4.863	305.5	41.39	9.558	0.835	300	41	315.4	340.4	333.8
17:00 10th of July	34.24	4.348	273.3	37.03	8.573	0.747	270	37	314.3	340.8	333.5
18:00 10th of July	34.25	3.749	235.6	31.92	7.412	0.644	230	30	314.3	340.5	333.6
19:00 10th of July	34.45	1.645	102.8	13.93	3.312	0.281	100	13	314.3	337.3	333.8
20:00 10th of July	34.67	0.4361	27.05	3.66	0.9728	0.074	270	3	312.6	333.6	333.0