NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

THE EFFECT OF TEMPERATURE ON THE
OPTIMIZATION OF PHOTOVOLTAIC CELLS USING
SILVACO ATLAS MODELING

by

Brian P. Sullivan

September 2010

Thesis Advisor: Sherif Michael
Second Reader: Todd R. Weatherford

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE September 2010	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE The Effect of Temperature on the Optimization of Photovoltaic Cells Using Silvaco ATLAS Modeling		**5. FUNDING NUMBERS**	
6. AUTHOR(S) Brian P. Sullivan			

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
   Naval Postgraduate School
   Monterey, CA 93943-5000
		8. PERFORMING ORGANIZATION REPORT NUMBER	

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A		**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**	

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: _____N/A_____.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)

In this thesis, the author analyzes the effect of temperature on the performance of photovoltaic cells using a virtual wafer fabrication tool called Silvaco ATLAS. Work from a previous Naval Postgraduate School thesis provided the initial InGaP/GaAs/Ge triple-junction solar cell design used in this analysis. As is often done, the previous work on this solar cell involved optimizing it at 300 K, even though operating temperatures are typically higher. Therefore, for this thesis, the author models each subcell of the multijunction solar cell at 275 to 400 K in 25-degree increments while varying their thicknesses and doping levels, as well as varying the molecular composition of InGaP. The author chose to vary these design parameters to observe their effect on performance and suggest a better design for operating at higher temperatures. The percent improvement increases to more than 11.9% as the temperature increases to 400 K when the author increases the mole fraction of InGaP and the doping in the GaAs emitter and the base of each subcell.

14. SUBJECT TERMS Silvaco ATLAS, Modeling, Simulation, Photovoltaic Cells, Multijunction Solar Cells, Temperature, Optimization, Indium Gallium Phosphide, Gallium Arsenide, Germanium	15. NUMBER OF PAGES 171
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

THE EFFECT OF TEMPERATURE ON THE OPTIMIZATION OF PHOTOVOLTAIC CELLS USING SILVACO ATLAS MODELING

Brian P. Sullivan
Captain, United States Marine Corps
B.S., Mississippi State University, 2002

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 2010

Author: Brian P. Sullivan

Approved by: Sherif Michael
                 Thesis Advisor

                 Todd R. Weatherford
                 Second Reader

                 R. Clark Robertson
                 Chairman, Department of Electrical and
                 Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

In this thesis, the author analyzes the effect of temperature on the performance of photovoltaic cells using a virtual wafer fabrication tool called Silvaco ATLAS. Work from a previous Naval Postgraduate School thesis provided the initial InGaP/GaAs/Ge triple-junction solar cell design used in this analysis. As is often done, the previous work on this solar cell involved optimizing it at 300 K, even though operating temperatures are typically higher. Therefore, for this thesis, the author models each subcell of the multijunction solar cell at 275 to 400 K in 25-degree increments while varying their thicknesses and doping levels, as well as varying the molecular composition of InGaP. The author chose to vary these design parameters to observe their effect on performance and suggest a better design for operating at higher temperatures. The percent improvement increases to more than 11.9% as the temperature increases to 400 K when the author increases the mole fraction of InGaP and the doping in the GaAs emitter and the base of each subcell.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Technology surrounding the development of solar cells
continues to expand as the demand for solar cells grows.
However, the solar cell industry often advertises the
performance of its solar cells based on operating near
300 K, even though operating temperatures as high as 400 K
are common. Designing cells at unrealistic operating
temperatures hinders the development of solar cells because
it hides the actual performance of cells at realistic
operating temperatures. The performance of cells at the
higher, more realistic temperatures is important, since
efficiency decreases as temperature increases.

In order to develop solar cells that operate better at
realistic operating temperatures, it is beneficial to
understand the effect temperature has on the performance of
solar cells. An InGaP/GaAs/Ge triple-junction cell serves
as the model for this analysis. The performance of this
cell is simulated using Silvaco's virtual wafer fabrication
software called ATLAS. Matlab is used to vary the cell's
operating temperature and its design specifications for
each ATLAS simulation. The design parameters that are
varied for these tests are the InGaP mole fraction and
emitter and base thicknesses and doping levels of each
subcell. These design parameters are varied and each new
design is tested at 275, 300, 325, 350, 375, and 400 K to
observe the effect altering the design has on solar cell
performance in this temperature range.

The ideal result of these design variations is to find
at least one design parameter that produces a cell that

does not drop in efficiency as the temperature increases. However, for all the design parameters that are varied, the output of the cell decreases as temperature increases over the range of temperature and design specification tested. Nevertheless, it is important to know the expected operating temperature of a cell when designing it to see whether some design specification can be altered to give a better performing cell at this temperature. Furthermore, even though efficiency drops as temperature increases, improvements in efficiency at 300 to 400 K are obtained when the InGaP mole fraction, GaAs emitter doping, and InGaP, GaAs, and Ge base doping are increased. Increasing the mole fraction and doping in these layers causes an increase in the triple-junction cell's output as the temperature increases from 300 to 400 K, giving an 11.92% increase in efficiency at the highest temperature tested.

ACKNOWLEDGMENTS

When considering who has contributed in the progress of this thesis, several people come to mind. Specifically, I would like to extend a sincere thank you to Dr. Sherif Michael, my thesis advisor, for his invaluable guidance and assistance that made the completion of this thesis a reality. I would also like to express my appreciation to Dr. Todd Weatherford, my second reader. His immeasurable knowledge, in particular of semiconductor physics, provided the foundation I needed to conduct the research for this thesis.

My family has also been a source of strength while completing this thesis. I would like to thank my parents, whose voice of encouragement has meant a great deal to me during not only my endeavors with this thesis, but also throughout my life. I also want to thank Chandelle, my wife who continually supports me in all that I set out to accomplish, and my children, who have kept me smiling among the many long hours of work.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

Alternative energy has become more popular. One form of alternative energy that has proven useful for space and terrestrial applications is photovoltaic cells. As the demand for photovoltaic cells increases, its ability to fulfill a wide range of needs must improve, as well as its efficiency in meeting those needs. In the beginning, photovoltaic cells were single-junction, but with time, the use of multijunction solar cells proved to be a more efficient means of using solar energy. Since it costs more to manufacture these multijunction solar cells, improving their efficiency is even more of a concern. A major factor that determines how efficient photovoltaic cells are in producing power is the temperature at which they operate.

B. OBJECTIVE

Temperature is a major factor determining the operating performance of photovoltaic cells. Since those who use photovoltaic cells often require them to operate in high temperature conditions, such as when powering satellites in space, the objective of this thesis is to examine the effect temperature has on the performance of photovoltaic cells, and improve the efficiency of multijunction solar cells based on this analysis. The author conducts all tests using Silicon Valley Company (Silvaco) Data System Inc.'s ATLAS software, which has proven to model the performance of solar cells accurately.

C.   RELATED WORK

Many other researchers at the Naval Postgraduate School have modeled solar cells using Silvaco ATLAS. Michalopoulos [1] and Green [2] demonstrated that the software could successfully model solar cells by building solar cells with published data and obtaining similar results. Lavery [3] improved the general software model of solar cells by figuring out how to better simulate the behavior of the tunnel junctions in multijunction solar cells. Bates [4], Utsler [5], and Tsutagawa [6] used Silvaco ATLAS to optimize the performance of solar cells. Canfield's [7] and Crespin's [8] simulations of solar cells focused on the effect of temperature and radiation, respectively. Canfield's work was on thermophotovoltaic cells, which is a specific type of solar cell not covered in this thesis. The focus in this thesis is to analyze the effect of temperature on basic photovoltaics using a multijunction solar cell optimized by Tsutagawa to be 36.28% efficient.

II. SEMICONDUCTOR FUNDAMENTALS

This thesis examines the effect of temperature on multijunction solar cells and methods used to optimize the performance of solar cells. To provide a basis for this work, this chapter covers some fundamental principles of semiconductors. Readers who are familiar with the fundamentals of semiconductors and photovoltaic cells may want to skip all or some of this chapter and the next chapter and go directly to Chapter IV.

A. MATERIAL PROPERTIES

Like any material, the material used to make photovoltaic cells demonstrates specific properties. These material properties determine the characteristics and performance of the solar cells. A key electrical property of materials is resistivity, ρ, or its reciprocal conductivity, σ. A material is either an insulator, a semiconductor, or a conductor based on its conductivity. Photovoltaic cells are made of semiconductors. The semiconductor materials of photovoltaic cells can be either elemental or compound. Silicon (Si) and Germanium (Ge) are two popular materials in solar cells. These two elemental materials are found in group IV of the periodic table. A portion of the periodic table is shown in Figure 1.

| | III | IV | V | VI |

Figure 1.   Portion of Periodic Table.

Besides those elements in group IV, compounds can be semiconductors if made of elements from groups III and V. For instance, Gallium Arsenide (GaAs) and Indium Phosphate (InP) are compounds that are semiconductors. It is also possible to take more than one element from either group III or V, combine it with an element from the other group, and form a semiconductor compound, such as Indium Gallium Phosphate ($In_{1-x}Ga_xP$) and Indium Gallium Nitrogen Arsenide ($In_{1-x}Ga_xN_{1-y}As_y$). The x and y in these compounds specify the mole fraction of the compounds. Semiconductor fabricators also combine elements from groups II and VI to form semiconductor compounds, such as Zinc Oxide (ZnO). However, the solar cells analyzed in this thesis are made of

4

semiconductors from elements in group IV and compounds formed from elements in groups III and V.

B. ATOMIC STRUCTURES

The number of protons, neutrons, and electrons determine the specific element in the periodic table. Since the elements of the periodic table are arranged according to their number of protons, neutrons, and electrons, it is significant that the elements in group IV, or combinations of elements equal-distant from group IV, are semiconductors. This is important because of the number of electrons in the outer shell of the atoms of the element or compounds. Based on Niels Bohr's theory, the electrons of an atom reside in orbits or shells around the nucleus. The closest shell has a maximum of two electrons in it, while all other shells have a maximum of eight electrons in them. The shells closest to the nucleus fill up first. The elements in group IV, or compounds formed from elements equal-distant from group IV, have four electrons in the outer, or valence, shell of their atoms. Therefore, each atom of these elements or compounds is able to form a bond with four other atoms by sharing one of its electrons in the outer shell with each of the other four atoms. In Figure 2, atoms of a semiconductor are pictured sharing their four valence shell electrons with nearby atoms in an effort to fill up this outer shell with the maximum number of electrons allowed.
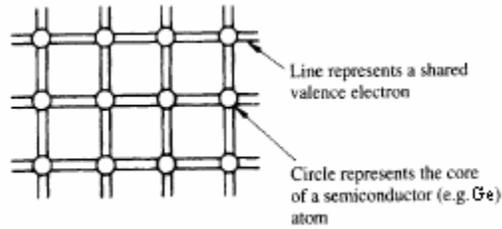
Figure 2.   Bonding Model (From [9]).

Each pair of shared electrons forms a covalent bond. When each atom is able to share its outer shell electrons with four other atoms, it is equivalent to having atoms with their valence shell full of electrons. This is ideal, since atoms want to reach their lowest possible energy state by having their valence shell full. If the outer shell is not full, the atom will attempt to give up or take additional electrons to fill its outer shell and achieve atomic stability. This is why elements in group III, which have three electrons in the outer shell, combine with elements in group V, which have five electrons in the outer shell, to form semiconductor compounds with full valence shells via covalent bonds.

C.   DOPING

The absence of an electron is a hole. The concept of the absence of holes and free electrons and the presence of a free electron and the presence of a hole is illustrated in Figure 3.

6
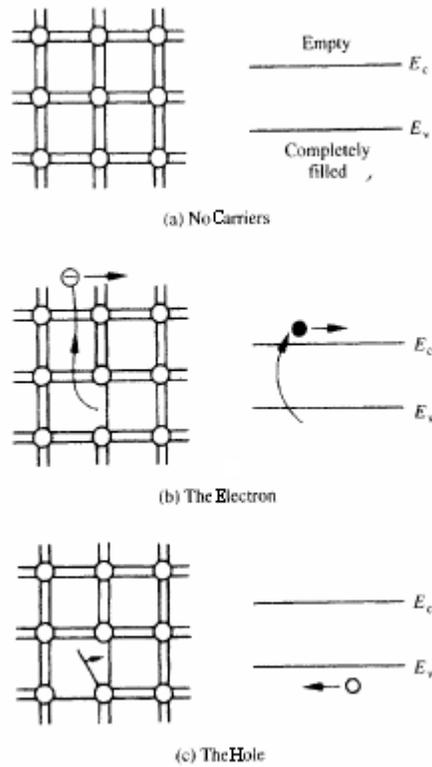
Figure 3. Carriers (From [9]).

In a pure, or intrinsic, semiconductor, as shown in Figure 3 (a), energy is required in order to break the covalent bond, release the electron, and create a hole. However, if the semiconductor is doped with another material, there may exist free electrons or holes naturally. In an extrinsic semiconductor, an impurity has been added that either has additional electrons that cannot be bonded with nearby atoms to fill the valence bands or holes because there are not enough electrons to fill the valence bands. An example of both these cases is demonstrated in Figure 4.
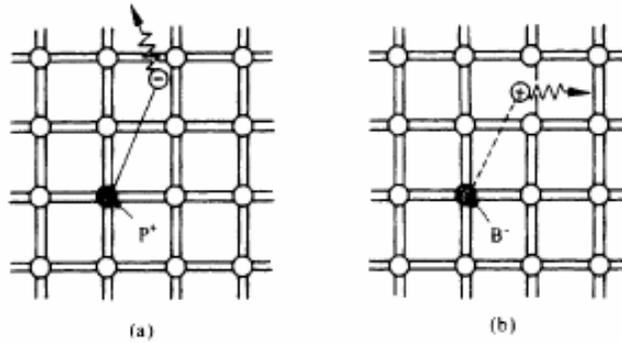
Figure 4.   (a) Donor and (b) Acceptor (From [9]).

The picture in (a) is of a semiconductor that has been doped with Phosphorous (P), which is from group V of the periodic table and a donor because it donates an extra electron. Semiconductors doped with donors are called n-type semiconductors because they have a majority of mobile negative charge carriers, or electrons, and a minority of positive charge carriers, or holes [9]. The picture in (b) is of a semiconductor doped with Boron (B), which is from group III of the periodic table, and an acceptor because it can accept an extra electron, creating a hole. Semiconductors doped with acceptors are called p-type semiconductors because they have a majority of mobile positive charge carriers, or holes, and a minority of negative charge carriers, or electrons [9]. Therefore, doping, whether n-type or p-type, increases the number of carriers in the semiconductor material. The presence of carriers enables the flow of current. Therefore, if a semiconductor material's valence bands are full due to covalent bonds, no carriers exist and no current flows.

D.   BAND GAP

Current flows in a semiconductor by creating carriers by doping the material and breaking the covalent bonds. The energy required to break the covalent bonds is called the band gap energy, E_g. The band gap energy is dependent on the material and temperature. In Figure 5, the pictures that are given show how band gaps differ, depending on whether the material is an insulator, semiconductor, or conductor (assuming that they are at the same temperature).
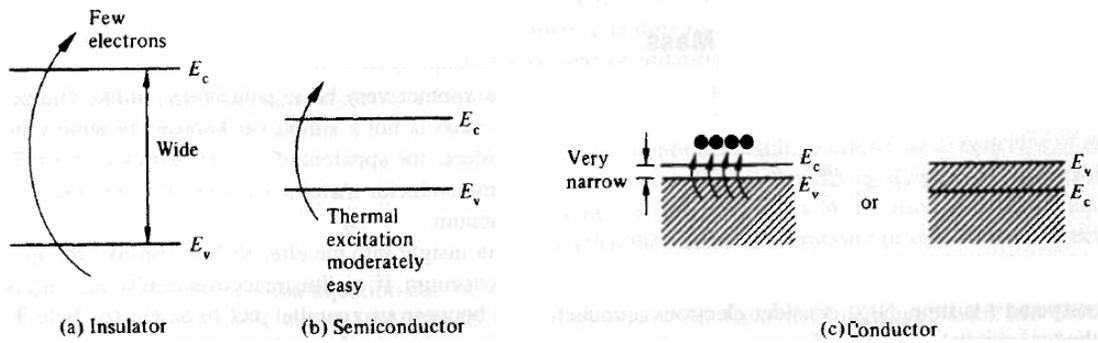
Figure 5.   Band Gap (From [9]).

As the pictures show, the band gap is the difference between the energy level of the conduction band, E_c, and that of the valence band, E_v, with the conduction band having a higher energy level. Insulators have the largest band gaps, being greater than 5 eV, and conductors have the smallest band gaps near 0 eV. Semiconductors have band gaps that range between the other two material types and can behave more like an insulator or conductor if the application calls for it.

A minimum energy must be maintained in order to keep the electrons in the conduction band allowing current to flow. Electrons cannot stay somewhere between the valence

and conduction bands; they must reside in either one or the other. In equilibrium, semiconductors have a certain number of free electrons, or electrons in the conduction band, at a specific temperature. The thermal energy, which is based on the operating temperature, is needed to excite the electrons to move them from the valence band to the conduction band [10].

E.   FERMI LEVEL

The Fermi level indicates the probability that an electron will occupy an energy state E at equilibrium. The Fermi function is a probability distribution function where the probability of the electron energy, E, is a function of the electron energy, the Fermi level, E_F, Boltzmann's constant, k, and the temperature, T, in Kelvin.

$$f(E) = \frac{1}{1 + e^{(E-E_F)/kT}}$$ (2.1)

As dopants are added to an intrinsic semiconductor, the Fermi level changes based on the new carrier concentrations. The various possible locations of the Fermi level, which are dependent on the amount of dopant in the semiconductor, are shown in Figure 6.
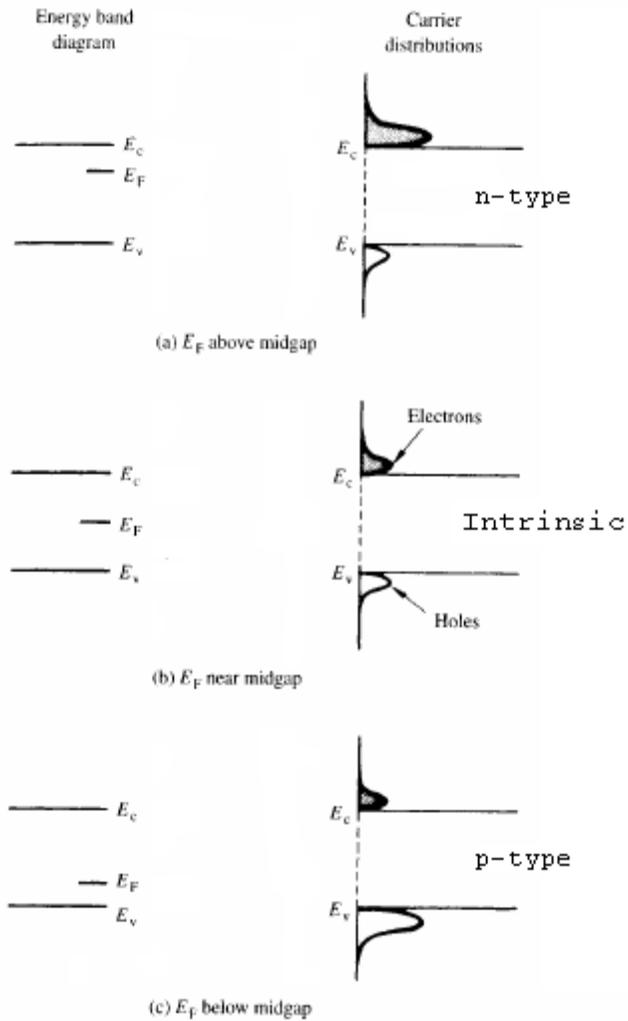
Figure 6.   Carrier Distributions (After [9]).

The Fermi level of an intrinsic semiconductor lies right between the valence and conduction bands, since the electron and hole concentration levels are the same at equilibrium. The Fermi level of an n-doped semiconductor is closer to the conduction band, since the material has a greater electron concentration. The Fermi level of a p-doped semiconductor is closer to the valence band, since the material has a greater hole concentration.

F.   RECOMBINATION AND GENERATION

Recombination and generation are a result of the movement of carriers. Carriers move, causing the flow of current either by drift or by diffusion. If one applies an external force, like an electric or magnetic field, to the material, carriers drift. Figure 7 is a picture of how the carriers drift due to this type of external force.
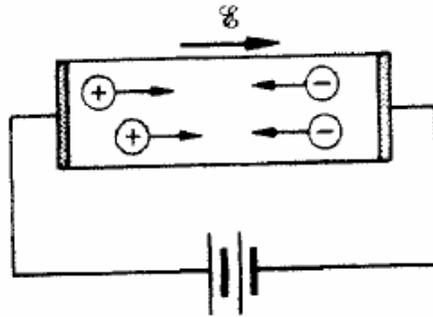
Figure 7.   Carrier Drift (From [9]).

Drift current, I_S, moves in the direction of the external force and the holes, which are a circled plus sign in Figure 7. The electrons, which are circled minus signs in the figure, move in the opposite direction. The minority carrier concentration and temperature has a great influence on drift current.

However, the majority carrier concentration and doping have more of an effect on diffusion current, I_D. Carriers diffuse when there is a higher concentration of them in one portion of the semiconductor material, which is often the result of doping. They move toward areas of the material where their concentration is smaller in an effort to

12

distribute the concentration. In Figure 8, the drawing illustrates how the carriers diffuse through the semiconductor.
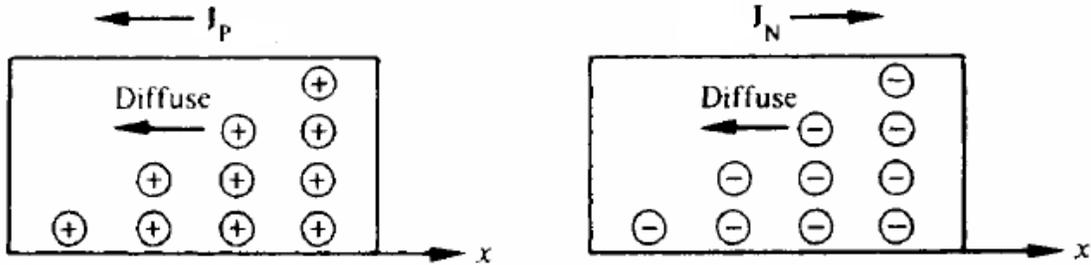
Figure 8.   Carrier Diffusion (From [9]).

The direction of current flow is dependent on which carrier is the majority carrier. Diffusion current flows in the same direction as holes diffuse but in the opposite direction that electrons diffuse. The magnitude and direction of the resulting diffusion current is the vector sum of the two carrier diffusion currents.

Since a hole is actually the absence of an electron, it is possible that the drift or diffusion of the carriers can cause electrons and holes to recombine. These electron-hole pairs disappear through recombination, thus reducing the number of carriers and, consequently, the flow of current. Therefore, a slow recombination rate is desirable when using photovoltaic cells. The recombination of an electron-hole pair releases energy in one of three ways. The energy is released by emitting a photon (light), emitting a phonon (heat), or transferring energy to another carrier through a process called Auger recombination.

Recombination occurs in either one of three ways: direct recombination, indirect recombination, or Auger recombination. An illustration of each of these methods of recombination is shown in Figure 9.

(a) Band-to-band recombination

(b) R–G center recombination

(c) Auger recombination
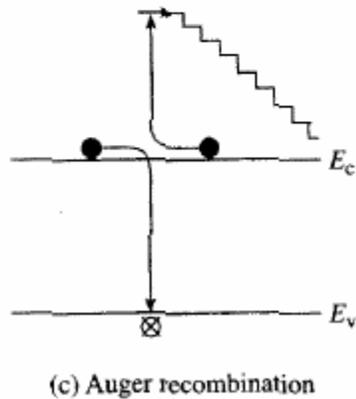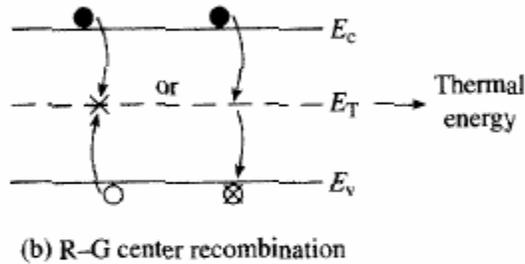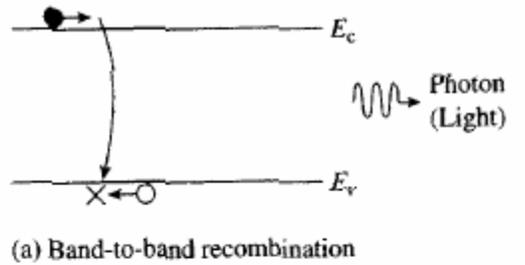
Figure 9. Recombination Methods (From [9]).

Typically, a photon is released when the recombination is band-to-band, or direct, recombination and involves the combination of a conduction band electron and a valence band hole. Indirect recombination occurs at special

locations called R-G centers. The R-G centers are defects in the material or impurity locations where one carrier type is trapped and recombines with the other carrier type. Indirect recombination typically releases thermal energy, or heat [9]. Auger recombination is a special case of band-to-band recombination wherein recombination occurs at the same time that two like carriers collide. The like carrier that does not recombine receives the transfer of energy and gradually loses this energy in a stepwise fashion, as illustrated in the bottom of Figure 9. On the other hand, generation occurs when an electron-hole pair is created. To generate these carriers, the different types of recombination described above are reversed [9].

G.   P-N JUNCTIONS

A p-n junction, or diode, is used in many semiconductor device applications. The formation of a p-n junction helps to control the electrical characteristics of the material. A p-n junction forms when a material is doped with impurities. One side is doped with acceptors (p), and the other side is doped with donors (n).

When the junction initially forms, there is a very large majority carrier density gradient at the junction. As the material strives for equilibrium, holes, as the majority carrier, will diffuse from the p-region into the n-region, and electrons, as the majority carrier in the n-region, will diffuse into the p-region, as illustrated in Figure 10.
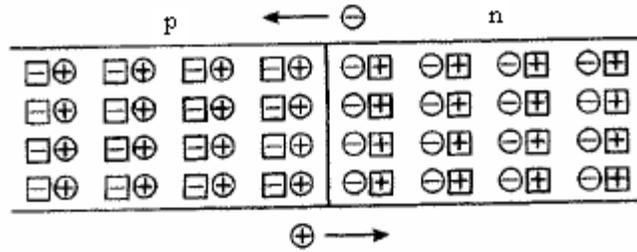
Figure 10.   Diffusion at P-N Junction (After [9]).

    As the majority carriers diffuse, they will recombine with the minority carriers, creating a depletion region, as long as no external electric field is applied. The depletion, or space charge, region, which is shown in Figure 11, is the area at the junction where the majority carriers have diffused and left behind oppositely charged atoms.
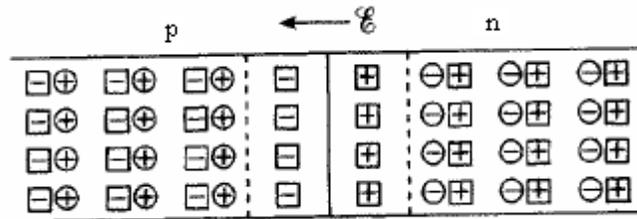
Figure 11.   Depletion Region of P-N Junction (After [9]).

    A small drift current develops, due to the minority carriers in the depletion region being repelled to the opposite ends of the region. The depletion region will continue to grow until eventually, the current generated from the diffusion of the majority carriers gradually decreases and is equal to the current due to the drift of the minority carriers in the depletion region. Without any additional energy applied to the material, the Fermi levels of the p- and n-regions are the same. A built-in voltage,

16

V_{bi}, exists that maintains equilibrium between the majority carrier of one region and the minority carrier of the other region. The formation of a V_{bi} is demonstrated in Figure 12.
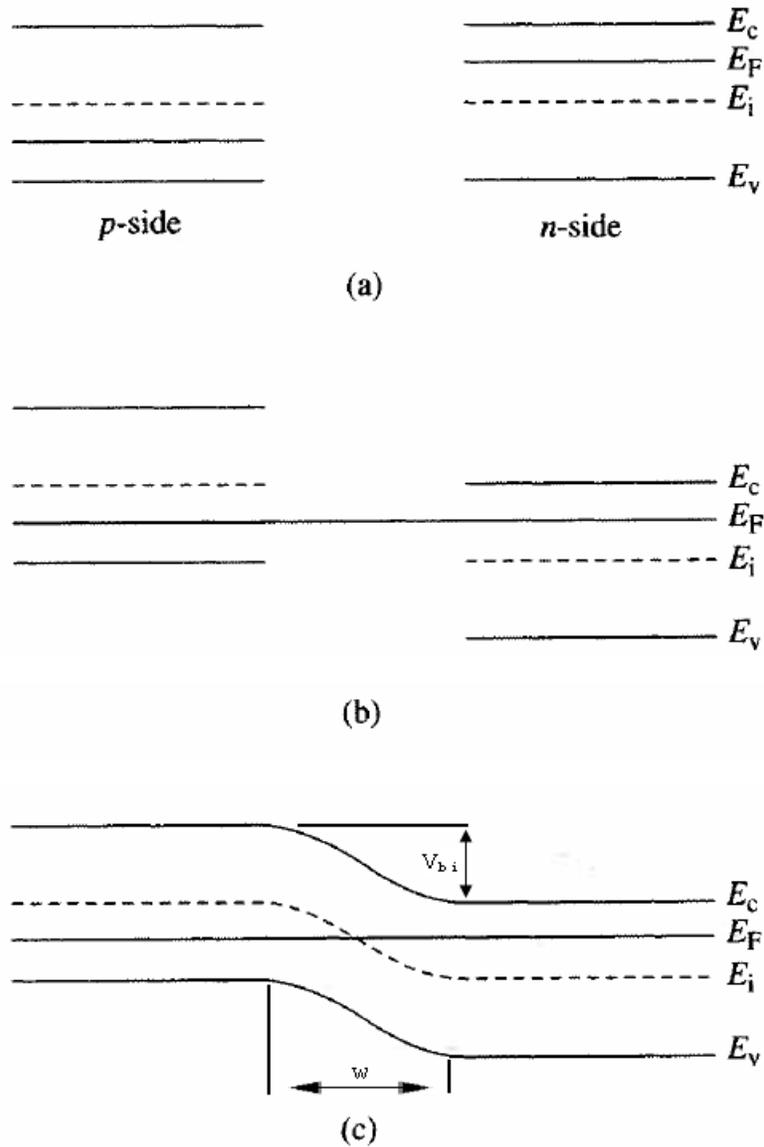
Figure 12.   Depletion Region in Equilibrium (After [9]).

The magnitudes of this built-in voltage, V_{bi}, and the depletion region width, w, are dependent on the doping levels in each region.

Applying a bias across the p- and n-regions causes the depletion region to leave the state of equilibrium. Under an applied voltage, the Fermi levels in each region will not be the same. A positive voltage applied to the p-region is a forward bias. This applied voltage, V_A, increases the Fermi level of the p-region while decreasing the Fermi level of the n-region, as illustrated in Figure 13.

(a) Equilibrium $(V_A = 0)$
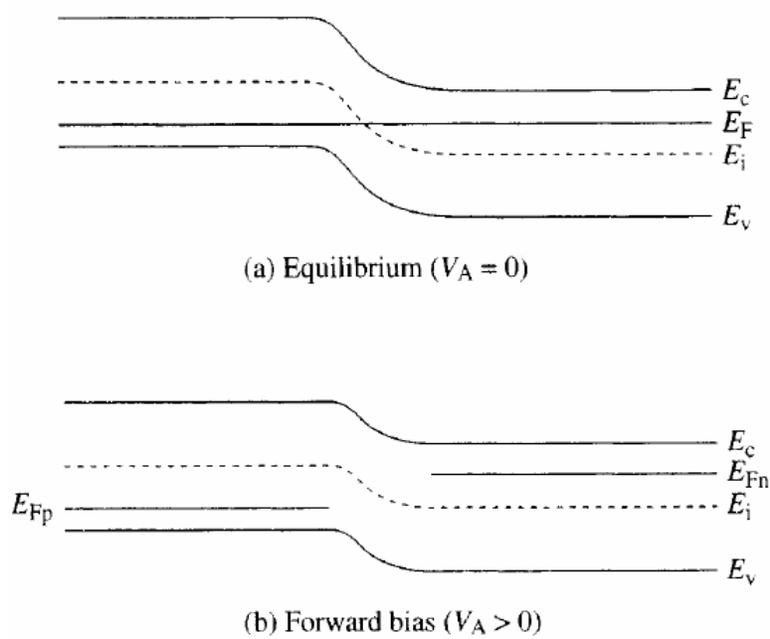
(b) Forward bias $(V_A > 0)$

Figure 13.   Forward Bias Applied (From [9]).

It acts against the built-in voltage and decreases the width of the depletion region. No current exists until a threshold voltage is reached that is greater than the built-in voltage.

A negative voltage applied to the p-region with respect to the n-region is a reverse bias. A reverse bias increases the Fermi level of the n-region while decreasing the Fermi level of the p-region, as illustrated in Figure 14.

18

(a) Equilibrium $(V_A = 0)$
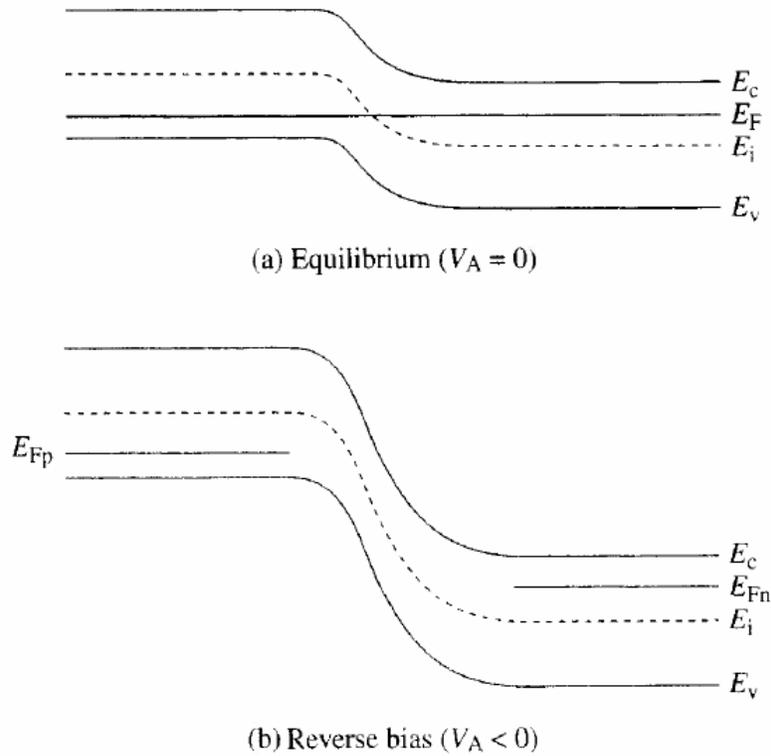
(b) Reverse bias $(V_A < 0)$

Figure 14.   Reverse Bias Applied (From [9]).

A reverse bias increases the width of the depletion region and adds to the built-in voltage to increase the total potential between the two regions. This increased potential allows only a small diffusion current to exist. Therefore, the total current is approximately equal to the drift current. However, if a large enough voltage under reverse bias is applied, the p-n junction will essentially breakdown and allow a large current to flow.

THIS PAGE INTENTIONALLY LEFT BLANK

III. SOLAR CELL FUNDAMENTALS

With the basics of semiconductors covered, it is now appropriate to discuss the fundamentals of solar cells.

A. BASIC OPERATION

First, it is important to understand that a solar cell is simply a p-n junction, or diode. For this thesis, solar-cell construction consists of n-type on top of p-type semiconductor material. The n-type material is the emitter, and the p-type is the base, or substrate [10].

With light shining on a solar cell, it produces power through a process called photogeneration. Photons from the source of light enter the semiconductor material when they contain energy equal to or greater than the band gap of the material and transfer enough energy to electrons in the valence band to cause them to move to the conduction band. This transfer of energy, therefore, creates electron-hole pairs. Photons with less energy than the band gap energy of the material do not have enough energy to cause an electron to move from the valence band to the conduction band, and, therefore, the solar cell does not absorb them. Photons with an energy level close to the band gap energy level are ideal, since a greater photon energy level can cause lattice vibrations in the material that produce heat, which negatively affects the performance of solar cells. However, as electron-hole pairs appear, the electric field of the depletion region causes the carriers to separate to opposite ends of the material. A demonstration of this process is shown in Figure 15.
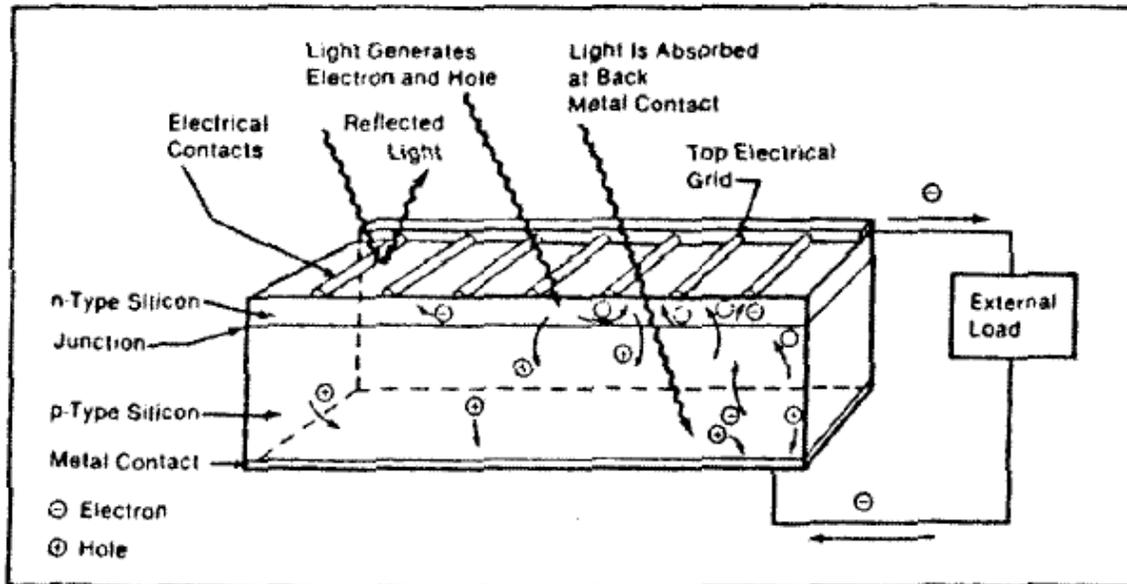
Figure 15. Solar Cell Operation (From [10]).

As illustrated in Figure 15, the electrons move to the n-region, giving an excess negative charge at the top of the solar cell while the holes move to the p-region giving an excess positive charge at the bottom of the solar cell [10].

The flow of carriers to opposite ends of the device creates a current in the material and a voltage differential on the two ends. Metal contacts placed on the top and bottom of the solar cell allow it to power an attached circuit, as illustrated in Figure 15. The metal contacts on top of the solar cell are extremely thin and evenly distributed to allow light to shine on the solar cell. The metal contact on the bottom of the solar cell covers the whole surface. This bottom contact is often polished to assist in the creation of electron-hole pairs by reflected light that has passed through the solar cell without creating an electron-hole pair back through the

solar cell. If any electrons or holes recombine at any time during this process, the number of free carriers is reduced as well as the resulting current flow, and, therefore, the productive output of the solar cell is reduced [10].

B.   PERFORMANCE CHARACTERISTICS

Several performance characteristics specify how well a solar cell operates. These performance characteristics are open-circuit voltage (V_{OC}), short-circuit current (I_{SC}), maximum voltage (V_{MAX}), maximum current (I_{MAX}), maximum power (P_{MAX}), fill factor (FF), and efficiency (η). One can visualize most of these characteristics on the current-voltage (I-V) curve, which is shown in Figure 16 with negative current oriented up.
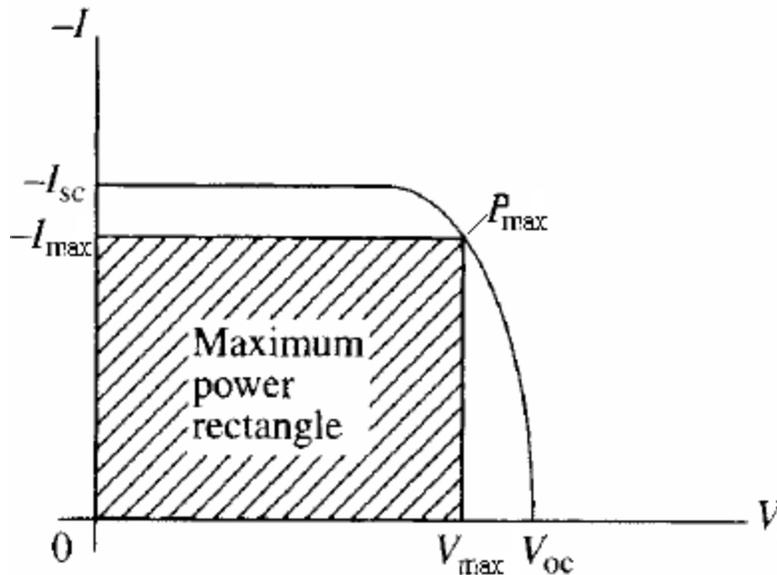
Figure 16.   I-V Curve (After [9]).

The solar cell generates power if its operating point lies in the fourth quadrant where the current is negative and the voltage is positive, which is achieved by placing a

23

load in series with the illuminated solar cell. The open-circuit voltage, V_{OC}, is the maximum voltage supplied by the cell at equilibrium for a specific illumination achieved when no load is connected to the solar cell. Furthermore, the short-circuit current, I_{SC}, is the maximum current generated by the solar cell which would be achieved by simply connecting the top and bottom contacts of the solar cell allowing the current to flow without any measurable resistance or voltage differential present. When the solar cell is illuminated with a load connected, the operating point will lie on the I-V curve between the I_{SC} and the V_{OC}. The maximum voltage, V_{MAX}, and maximum current, I_{MAX}, define the operating point that produces the maximum power, P, and is, therefore, the desired operating point. The fill factor, FF, is the ratio of the maximum power rectangle to the V_{OC} and I_{SC} rectangle.

$$FF = \frac{P_{MAX}}{V_{OC} \cdot I_{SC}} = \frac{V_{MAX} \cdot I_{MAX}}{V_{OC} \cdot I_{SC}} \qquad (3.1)$$

The FF actually tells how "square" the I-V curve is. The closer this number is to 1 the more square the curve is. A squarer curve indicates a greater maximum power, since the current and voltage do not decrease as much at the maximum power point. The efficiency, η, of a solar cell is a ratio of the output power, P_{OUT} to the input power, P_{IN}, provided by the light source times 100 percent to give the efficiency as a percentage.

$$\eta = \frac{P_{OUT}}{P_{IN}} \cdot 100\% \qquad (3.2)$$

Since the desired operating point is at P_{MAX}, the author calculates the efficiency using P_{MAX} as P_{OUT} in this thesis [9].

$$\eta = \frac{P_{MAX}}{P_{IN}} \cdot 100\% = \frac{V_{MAX} \cdot I_{MAX}}{P_{IN}} \cdot 100\% = \frac{FF \cdot V_{OC} \cdot I_{SC}}{P_{IN}} \cdot 100\% \qquad (3.3)$$

Usually, the light source used in illuminating solar cells is the sun and it is the light source used in this thesis. The sun provides an average of 135 mW/cm^2 of input power for objects in orbit around the Earth and as much as 100 mW/cm^2 for objects on the surface of Earth.
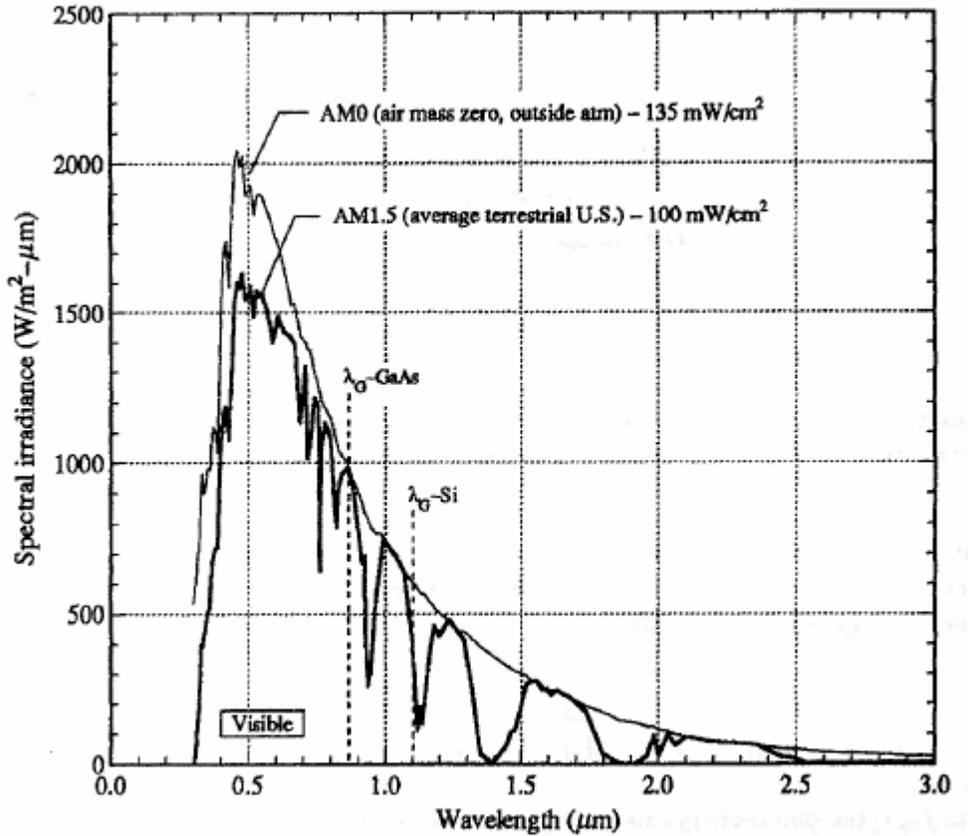
Figure 17.   Solar Spectrum (From [9]).

25

Figure 17 contains a plot of the solar spectrum for objects in Earth's orbit, called Air Mass Zero (AM0). Also shown in this figure is the spectrum for terrestrial applications: AM1.5. However, for the calculations in this thesis, the author uses AM0 exclusively. Just as the input power is per square centimeter, the output power is a power density with the same units. This is because the author measures the current per square centimeter for the solar cells in this thesis.

C.   PERFORMANCE FACTORS

Several factors exist that affect the performance of solar cells. The light source's spectral intensity affects the performance of a solar cell for good or bad, since the output power of a solar cell is dependent on the power made available to it [9].

1.   Negative Factors

However, several factors exist that negatively affect the performance of solar cells. Some of these negative factors are listed here:

- The natural reflection of light off the surface of solar cells can greatly reduce efficiency. The reflection of light can be as high as 36% for untreated surfaces [10].

- The reflection of light off the surface due to the electric conductors further decreases the efficiency by reducing the amount of light shining through the solar cell. The reduction in

light is equivalent to how much these contacts cover the surface, which is typically 8% [10].

- Photons entering the solar cell with a lower or higher energy level than the band gap energy level of the material can lower efficiency by creating heat and raising the temperature of the solar cell [10].

- The material's natural recombination rate of electron-hole pairs through direct and indirect recombination can reduce current flow and lower the efficiency of solar cells [10].

- However, the recombination rate is often greater due to material defects from impurities and imperfect crystal structures [10].

- Resistance present in the solar cell reducing charge and current flow reduces solar cell efficiency. This resistance exists in the bulk of the base, at the surface, and at the contact junction. Additionally, ohmic resistance in the metal contacts adds to the reduction of efficiency [10].

- Operating the solar cell at a higher or lower temperature than what is optimal for its particular design also reduces efficiency since the lattice vibrations that result reduce the flow of carriers [10]. The author discusses the effect of temperature in more detail later.

- Over time, the performance of solar cells in space applications are degraded [10].

2. Positive Factors

On the other hand, positive factors also exist to improve the performance of solar cells. Some of these factors are as follows:

- Making solar cells with shallow junctions that are less than 0.2 microns thick are called blue cells and can improve the efficiency of solar cells [10].

- Making the solar cell with a back surface field (BSF), which is a p+ layer on the bottom of the cell, can increase V_{OC} and I_{SC} resulting in a better FF and a higher efficient cell.

- The reflection of light off the surface of the solar cell can be reduced from as high as 36% to 5% if an antireflection coating, such as Silicon Oxide (SiO), and surface texturing are used [10].

- Surface texturing alone can increase solar cell efficiency by approximately 8% [10].

- The addition of a back surface reflector acting as a mirror to reflect the light back through the cell can increase efficiency by 3% [10].

- Producing multilayer solar cells can increase efficiency [10].

3. Multijunction Solar Cells

One understands the idea for creating multilayer, or multijunction, solar cells when considering the solar spectrum shown in Figure 17. As shown in this figure,

different materials accept different wavelengths of sunlight, which correspond to different levels of input power from the sun. When p-n junctions of different materials with varying band gaps are stacked on top of each other, the light passing thru the top layers is absorbed in the underlying layers. This increases the efficiency of the overall cell. The subcells are connected in series; therefore, the subcell with the lowest current defines the overall current for the cell, but the voltage of the cell is the sum of the voltages produced by each subcell. A solar cell consisting of multiple layers that accept more of the available light increases the useful power provided to the solar cell and produces a greater output power.

The way different materials accept different wavelengths may best be illustrated with specific examples. Some common materials used to form multijunction solar cells are InGaP, GaAs, InGaNAs, and Ge. These materials are lattice matched making it possible to stack them on top of one another to create a multijunction solar cell. The plot in Figure 18 shows the range of wavelengths absorbed by each of these semiconductors.
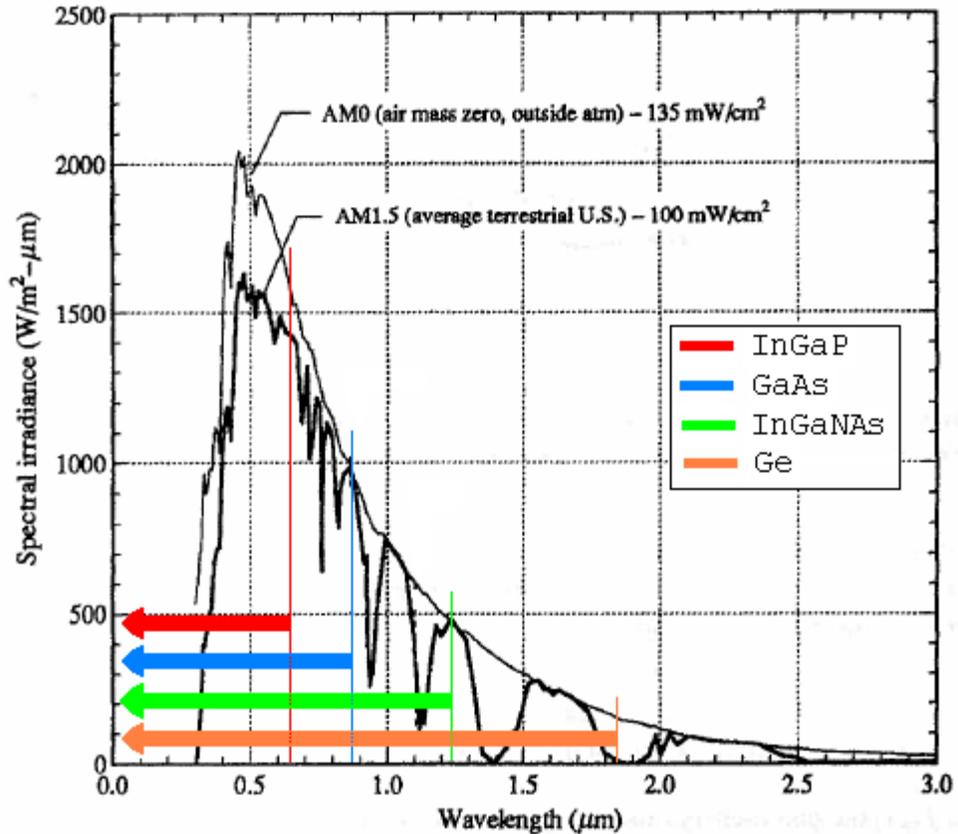
Figure 18. Wavelength Absorption (After [10]).

The semiconductors accepting the lowest range of wavelengths are placed on top of those accepting larger wavelengths. Therefore, out of the materials shown, InGaP is placed on top and Ge is placed on bottom. Photon energy, E, is inversely proportional to wavelength, λ, and a function of h, Planck's constant (4.136×10^{-15} eV-s), and c, the speed of light [10].

$$E = \frac{h \cdot c}{\lambda} \qquad (3.4)$$

Therefore, each material starting with InGaP will absorb the photons with energy levels equal to and above

its band gap energy level and pass photons with lower energy levels to the material below it until the bottom layer—Ge—is reached. Ge's low band gap energy level allows it to absorb almost the entire solar spectrum and, therefore, practically all the remaining photons that have been passed thru the layers above it.

The I-V characteristics of the materials of each layer of a multijunction solar cell also helps to explain why this type of solar cell is often more efficient than any single layer solar cell. Figure 19 contains an I-V curve of each of the materials previously shown on the solar spectrum plot.
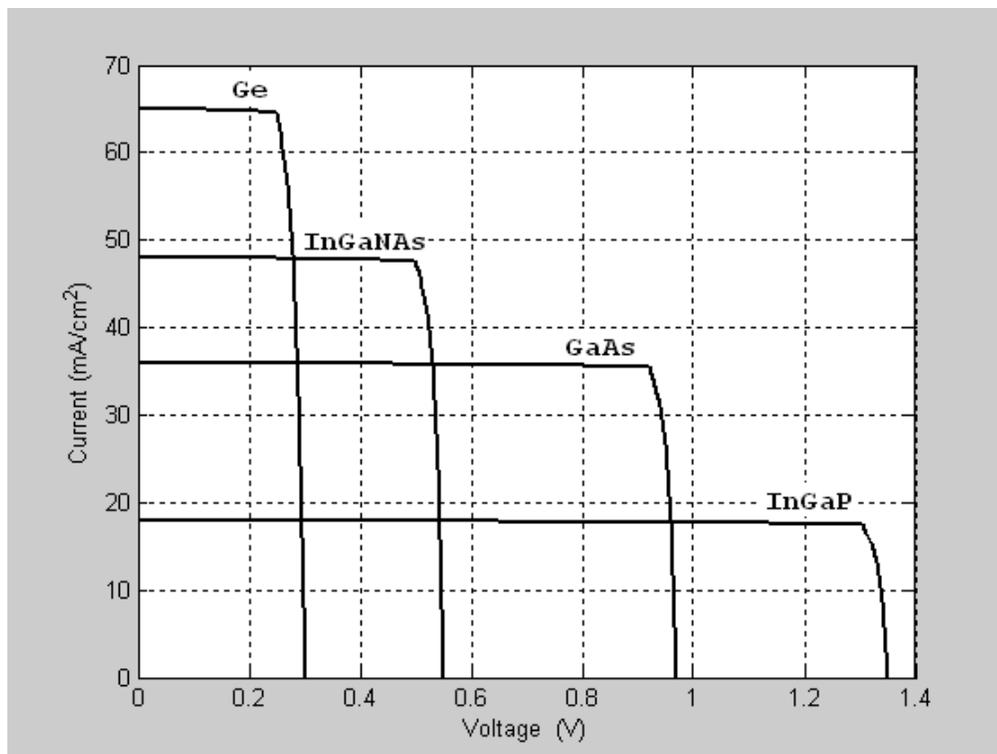
Figure 19.   I-V Curves (After [10]).

31

Although Ge has the highest current per square centimeter of all the materials, it has the lowest efficiency because its voltage is so low. On the other hand, InGaP has the highest efficiency of the single layer cells because its voltage is so high, even though its current density is the lowest. Assuming no subcell is thick enough to limit the current flow of the subcells below it, InGaP is the current-limiting layer of these materials. However, each voltage generated contributes to the total voltage produced by the multijunction solar cell, making it more efficient than any one of its single layers could be alone.

In order to connect the various layers of the p-n junctions in a multijunction solar cell a special kind of junction is inserted between each layer. Just connecting the single-layer cells on top of one another creates new junctions that are reverse-biased and oppose the flow of current due to the electric fields of the depletion regions forming at each junction. Instead, tunnel junctions are inserted that are p-n junctions but in the reverse order than the single-layers of the cell. For multijunction solar cells made of p-n junctions that are n-type on p-type material, the tunnel junctions are p-type on n-type material. In addition, the tunnel junction regions are doped with a greater concentration than the single-layer cells to which they connect. The greater doping concentration of the tunnel junctions allow for the free flow of carriers from the p to the p+ regions and from the n+ to n regions.

4.   Temperature

A major influence on the performance of solar cells and one particularly focused on in this thesis is the temperature at which they operate. Some increases in the operating temperature are beneficial because these increases can assist in the generation of electron-hole pairs through the transfer of heat energy and increase the rate of diffusion. However, as the temperature increases it begins to affect solar cell performance negatively. This is due to higher temperatures increasing the resistivity of the material's properties. Higher operating temperatures cause I_{SC} to increase slightly due to the rate of diffusion increasing, yet V_{OC} decreases at a significant rate. In fact, V_{OC} decreases at approximately 2 mV/$^{\circ}$C [10]. As an illustration of the general effect of temperature on I_{SC} and V_{OC}, consider the I-V curves in Figure 20.
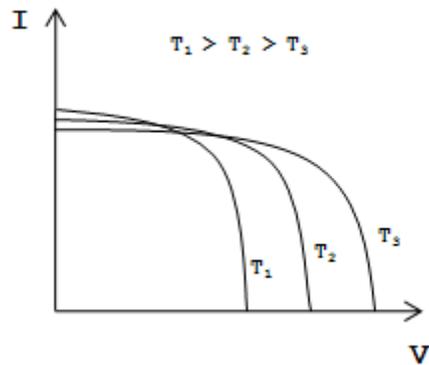
Figure 20.   Temperature Effect on I-V Curve (After [10]).

Also, as temperature is increased, the FF of the curves decreases. Because of the significant decrease in voltage and reduction in FF, higher operating temperatures greatly reduce the efficiency of solar cells. Even though

the current density increases a little, the decrease in the voltage reduces the maximum power of the solar cell.

In this thesis, the author makes changes to the design of the solar cell to see how performance is affected at high temperatures and to see if the changes result in a better performing solar cell. Some of the changes include varying doping levels and layer thicknesses as well as changing the mole fraction of semiconductor compounds. The mole fraction is of particular concern since it affects the band gap of the material. The effect of mole fraction on band gap is unique for each semiconductor. For example, the following equation is for determining the band gap of $In_{1-x}Ga_xP$ as a function of mole fraction at 300 K [11].

$$E_{g\_300K} = 1.35 + 0.73 \cdot x + 0.70 \cdot x^2 \qquad (3.5)$$

It is evident that the InGaP band gap increases as the mole fraction is increased. However, temperature also affects band gap. In fact, band gap narrows as temperature increases above 300 K. The Varshni Equation shows how band gap changes as a function of temperature.

$$E_g = E_{g\_300K} + \alpha \left[\frac{(300K)^2}{300K + \beta} - \frac{T^2}{T + \beta} \right] \qquad (3.6)$$

E_{g_300K} is the band gap of the material at 300 K and α and β are empirically determined values specific for each semiconductor. For example, the band gap for InGaP varies according to the following equation [12]:

$$E_g = E_{g\_300K} + 4.5x10^{-4} eV / K \left[\frac{(300K)^2}{300K + 400K} - \frac{T^2}{T + 400K} \right] \qquad (3.7)$$

Since mole fraction and temperature affect band gap in opposite ways, it could be possible to alter the band gap in the design process by changing the mole fraction so that the semiconductor demonstrates better results when subjected to high temperatures.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. SOLAR CELL MODELING AND SIMULATION SOFTWARE

The importance of modeling and simulating the performance of solar cells cannot be overemphasized. Building a solar cell and testing it to determine if it performs as desired is too expensive and time consuming, especially considering that this process may have to be repeated numerous times until a solar cell is built that produces the desired results. Silicon Valley Company (Silvaco) has created ATLAS, a technology computer aided design (TCAD) software program that effectively simulates the behavior of semiconductor devices. Silvaco's ATLAS program is the software used in this thesis to simulate the performance of solar cells.

A. SILVACO ATLAS

ATLAS is a physically based two- and three-dimensional device simulator of semiconductor devices. It is able to predict the electrical characteristics of specified solar cell structures [13]. The programs that Silvaco offers as tools to create specific semiconductor devices to simulate with ATLAS and the different type of results that ATLAS is capable of producing are shown in Figure 21.
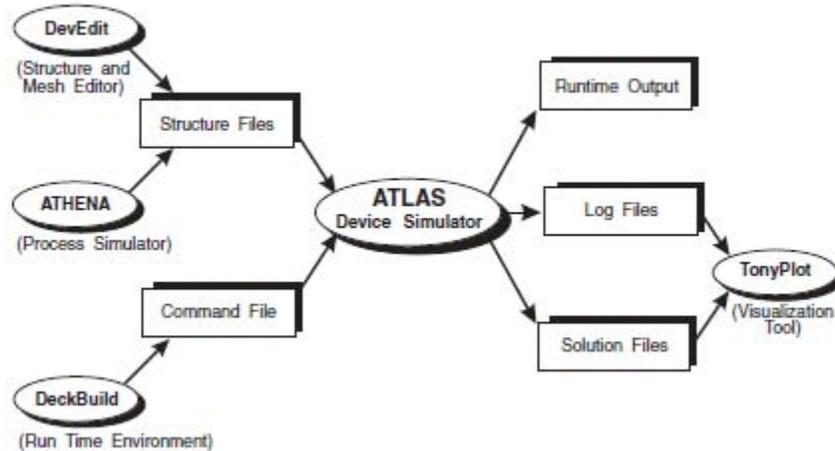
Figure 21.   ATLAS Inputs and Outputs (From [13]).

Since DeckBuild is capable of specifying the entire solar cell structure, it is the only input used in this thesis. The programmer builds the structure in the DeckBuild run-time environment where ATLAS is called to conduct the device simulation and gives the results as a run-time output or in log files. The programmer activates ATLAS in DeckBuild using the line:

```
go atlas
```

DeckBuild also contains a sequence of statements with set parameters used to build and simulate the semiconductor device. Each of these lines of code has the general form:

```
<STATEMENT>    <PARAMETER>=<VALUE>
```

In each statement, the parameters may have four different types of values: real, integer, character, or logical [13]. The following is a sample line of code for DeckBuild.

```
DOPING N.TYPE CONC=1.0e16 REGION-1 OUTFILE=my.dop
```

In this line of code, "DOPING" is the statement and the remaining are examples of logical, real, integer, and

character parameters, respectively [13]. Although the statement must come first in any line of code, the order of the parameters is arbitrary. The spelling of any parameters may be shortened as long as enough letters are present to distinguish it from any other possible parameter. Any logical parameter may be set to false by preceding it with the ^ symbol. A programmer inserts a comment within the DeckBuild lines of code by preceding it with the # symbol. The programmer must give the statements in DeckBuild in a specific order; otherwise, ATLAS generates errors or gives incorrect results. The different groups of statements and the order in which the programmer is to give them are shown in Figure 22.

Figure 22.   ATLAS Order of Statements (From [13]).

1.   Constants

To make it easier to change the design of the solar cell for numerous simulations, the programmer uses set commands to define common variables specifying the design of the solar cell. The following is an example of a set command in DeckBuild:

set x=0.51

This command defines the mole fraction of $In_{1-x}Ga_xP$ and allows it to be easily changed for different solar cell designs. The programmer can change the design of the solar cell more quickly this way since x and many other variables like it are used more than once in the DeckBuild lines of code. For example, x is used later where the material's composition is specified by x.comp=$x. The $ symbol means that a pre-defined variable is to be used as the x.comp value. Matlab is used in this thesis to run DeckBuild and iterate the specified variables, such as emitter thickness and emitter doping, given in the set commands to see how different solar cell designs perform. The programmer is able to run many simulations by changing two variables at a time.

2.   Structure

After specifying the set commands, the programmer defines the mesh, regions, electrodes, and doping of the structure.

a.   Mesh

The meshing is a set of horizontal and vertical lines that make a grid for building the structure. The programmer specifies these lines with commands in the following form:

X.MESH LOCATION=<VALUE> SPACING=<VALUE>

Y.MESH LOCATION=<VALUE> SPACING=<VALUE>

An example of a mesh created for ATLAS is given in Figure 23.


```
x.mesh loc=-250 spac=50
x.mesh loc=0     spac=10
x.mesh loc=250   spac=50

y.mesh loc=-0.87  spac=0.003
y.mesh loc=-0.86  spac=0.003
y.mesh loc=-0.69  spac=0.003
y.mesh loc=-0.34  spac=0.03
y.mesh loc=-0.06  spac=0.003
y.mesh loc=-0.05  spac=0.003
y.mesh loc=-0.02  spac=0.002

y.mesh loc=-0.005 spac=0.001
y.mesh loc=0.01   spac=0.001

y.mesh loc=0.02   spac=0.003
y.mesh loc=0.03   spac=0.003
y.mesh loc=2.815  spac=0.03
y.mesh loc=3.90   spac=0.003
y.mesh loc=3.91   spac=0.003
y.mesh loc=4.21   spac=0.002

y.mesh loc=4.225  spac=0.001
y.mesh loc=4.24   spac=0.001

y.mesh loc=4.29   spac=0.01
y.mesh loc=4.39   spac=0.01
y.mesh loc=304.39 spac=50
```

Figure 23.   Mesh Defined and Illustrated.

The solar cell's electrical characteristics are evaluated at each crossing of the horizontal and vertical lines defined in the mesh to determine the solar cell's overall performance [13].

Quantum tunnel mesh is a special type of grid specification used to simulate tunnel junctions in multijunction solar cells. Similar to the regular meshing, the programmer uses the following general format in quantum tunnel mesh statements:

```
QTX.MESH LOCATION=<VALUE> SPACING=<VALUE>

QTY.MESH LOCATION=<VALUE> SPACING=<VALUE>
```

This special mesh is required to measure the current flow through the tunnel junction and lies over the other mesh previously created. Since the flow of current is in the y direction through the tunnel junction, the spacing in this direction is very fine, whereas the spacing in the x direction is much larger. The following are examples of typical quantum tunnel mesh statements for one tunnel junction.

```
qtx.mesh loc=-250 spac=50

qtx.mesh loc=0 spac=10

qtx.mesh loc=250 spac=50

qty.mesh loc=4.225 spac=0.001

qty.mesh loc=4.2325 spac=0.00005

qty.mesh loc=4.24 spac=0.001
```

Although additional QTY.MESH statements are required for any remaining tunnel junctions in the solar cell, the QTX.MESH statements are sufficient for all tunnel junctions that exist in the solar cell.

b. Regions

The programmer further defines the solar cell by specifying the different materials in the regions of the structure. The following general code is used for this:

REGION number=<interger> <material_type> \

<position parameters>

In this example, the \ character means that the command continues on the next line. The following is an example of a regions statement used in DeckBuild:

region num=3 material=InGaP x.comp=0.51 x.min=-250 \ x.max=250 y.min=-0.86 y.max=-0.69

Up to 200 regions may be defined in ATLAS and a large number of materials are available for use. For materials with a specific mole fraction, the x and y composition fractions (X.COMP and Y.COMP, respectively) are to be specified in the region statements. The position parameters are specified using X.MIN, X.MAX, Y.MIN, and Y.MAX parameters. The programmer must specify a material for every portion of the structure defined by the mesh statements [13].

c. Electrodes

The next portion of the structure specification commands defines the electrodes. The general format for these commands is of the following form:

ELECTRODE NAME=<electrode name> <position_parameters>

For example, the following are statements for electrodes used in DeckBuild:

```
electrode name=cathode x.min=-5 x.max=5 y.min=-0.87 \
    y.max=-0.87

electrode    name=anode    x.min=-250    x.max=250    \
    y.min=304.39 y.max=304.39
```

Although up to 50 electrodes can be specified in the program, only the anode and cathode are needed for single-layer solar cells [13]. For multijunction solar cells, an electrode is added for every tunnel junction that exists. The anode is gold and only covers a small percentage of the top, usually eight percent or less. The cathode covers the entire bottom surface of the solar cell.

d. *Doping*

The last portion of the structure specification is the doping specified for each region of the solar cell. The general form of the doping statements is as follows:

```
DOPING    <distribution_type>    <dopant_type>    \
    <position_parameters>
```

The doping statements specify the type and amount of doping in each section of the solar cell. The two following doping statements illustrate how this is done.

```
doping uniform region=3 n.type conc=4.64e17

doping uniform region=4 p.type conc=1e17
```

The doping can be n- or p-type with either uniform, Gaussian, or complementary error function distribution specified [13].

3. Material Models

The material models specification consists of commands for specifying the materials, models, contacts, and interfaces of the solar cell.

a. Materials

The programmer defines the materials of the solar cell in DeckBuild using the following format:

MATERIAL <identification> <material_parameters>

The programmer can give the material identification in several ways. The material may be specified by a particular region or name that it has been given in previous lines of code, or the material statement might be given for all materials of its type. The following examples illustrate the different ways the material may be identified.

MATERIAL MATERIAL=InGaP EG300=1.9 MUN=1945

MATERIAL REGION=4 TAUN0=1e-7 TAUP0=1e-7

MATERIAL NAME=base NC300=1.3e20 NV300=1.28e19

Electron affinity (AFFINITY), room temperature band gap (EG300), electron (MUN) and hole (MUP) mobility, electron (TAUN0) and hole (TAUP0) recombination lifetimes, valence (NV300) and conduction (NC300) band density of states, and other material parameters are often defined for semiconductors. Otherwise, ATLAS uses a default value if a material parameter is not specified [13].

b.   *Models*

A programmer specifies the physical models using MODELS and IMPACT statements of the following form.

MODELS <physical_model> <model-dependent_parameters> \
    <general_parameters>

However, the physical models may be specified for specific materials by using the following format.

MODELS MATERIAL=<material> <physical_models> \

    <model-dependent_parameters> <general_parameters>

The physical models consist of five classes: mobility, recombination, carrier statistics, impact ionization, and tunneling [13]. The models that are used depend on the materials of the solar cell, and that is why they can be specified for only one material or the whole cell, if appropriate. A summary of the model parameters used in this thesis is presented in Table 1.

Table 1.   Basic Models (After [13]).

Parameter	Syntax	Explanation
Shockley-Read-Hall	SRH	Recombination model that uses fixed minority carrier lifetimes; should be used in most simulations
Bandgap Narrowing	BGN	Carrier statistics model that is important in heavily doped regions
Operating Temperature	TEMPERATURE	Parameter that specifies the operating temperature
Model and Parameter Verification	PRINT	Parameter that lists in the run-time output the models and parameters used during the simulation to allow for verification of correct modeling

A programmer could arguably use other models in the simulations conducted for this thesis; however, for simplicity and uniformity, these are the models used. This is especially crucial when considering that the basis of this thesis work stems from previous work conducted where these were the only models used. Therefore, in order to make equal comparisons between the simulations conducted in this thesis and previous work, it is important to use the same models in both sets of simulations.

To properly simulate solar cells with tunnel junctions, additional models are required. Models that deal directly with tunnel junctions are applied. A summary of some of the models to use with the tunnel junctions of multijunction solar cells is given in Table 2.

Table 2.   Tunneling Models (After [13]).

Parameter	Syntax	Explanation
Band-to-Band (standard)	BBT.STD	For direct transitions; required with very high fields
Klaassen Band-to-Band	BBT.KL	Includes direct and indirect transitions
Non-local Band-to-Band	BBT.NONLOCAL	For non-local band-to-band tunneling
	BBT.NLDERIVS	Includes non-local coupling

The BBT.STD and BBT.KL models can be used together with the BBT.NONLOCAL and BBT.NLDERIVS models or separate. Together, they often lead to results that are more accurate but require more time to calculate.

c.   Contacts

The CONTACT statement is used to describe the electrodes. These statements are of the following form:

CONTACT NAME=<name> <property_specification>

The CONTACT statements are not required if the properties of the electrodes need no special designation or modification [13].

d.   Interfaces

The INTERFACE statement is used to specify the interface charge density and surface recombination velocity at interfaces between semiconductors and insulators [13]. The INTERFACE statement is given in the following format.

INTERFACE <parameters> <interface_dimensions>

However, interface properties are not specified in the solar cells simulated in this thesis.

4.   Light Beams

For solar cell simulation, it is important to specify a light source. A programmer can accomplish this with one line of code. The following is an example of this.

```
BEAM NUMBER=1 X.ORIGIN=0 Y.ORIGIN=-5 ANGLE=90 \
BACK.REFL POWER.FILE=AM0nrel.spec \
WAVE1.START=0.12 WAVE1.END=2.4 WAVE1.NUM=1500
```

This statement completely describes the light source. The exact location and angle of the light relative to the solar cell is specified. The power file containing information about the sun is used. The range of wavelengths is also specified along with the number of beams to use in this range of wavelengths. BACK.REFL makes sure the reflection of sunlight off the surface of the solar cell is taken into account.

5.   Numerical Methods

The METHOD statement is the only statement in the numerical methods specification. A variety of numerical methods exists for making calculations to determine

semiconductor device behavior. Among these methods, three basic types of techniques for calculating the solutions exist: decoupled (GUMMEL), fully coupled (NEWTON), and decoupled and fully coupled (BLOCK). The METHOD statement is given in the following general format.

METHOD <methods>

The NEWTON method is the most preferred method used and is, therefore, the default method. However, as the following example illustrates, the programmer can specify more than one method to use on the models that are appropriate for them [13].

METHOD GUMMEL BLOCK NEWTON

6.   **Solution**

The solution specification consists of LOG, SOLVE, LOAD, and SAVE statements.

a.   *Log*

The LOG statement is used to capture all the terminal characteristics calculated by ATLAS in a specified file. The following general format is used for the LOG statement.

LOG OUTFILE=<filename>

The filename has the .log file extension. The creation of this log file makes it possible to save all the I-V characteristics of the solar cell.

b. Solve

The SOLVE statement is used to calculate specified bias points. This statement is of the following basic format.

SOLVE <bias_points>

The bias points can be single points are multiple points that are stepped from an initial point to a final point, as the following example illustrates.

SOLVE vanode=0.0 vstep=0.05 vfinal=2.4 name=anode

In order to obtain convergence for the equations used to calculate these bias points, it is important to give a good initial guess prior to this line of code.

c. Load

The LOAD statement loads previous solutions from files as initial guesses to other bias points [13]. This statement is of the following form.

LOAD INFILE=<filename>

If more than one file is to be loaded, the following format is used.

LOAD IN1FILE=<filename> IN2FILE=<filename>

d. Save

The SAVE statement saves all node point information into an output file [13]. The general format for this statement is as follows.

SAVE OUTFILE=<filename>

The mesh, electrodes, doping, and region boundaries are stored in the file along with all electrical data from the last solution if a SOLVE statement precedes the SAVE statement.

7. Results

The results analysis consists of EXTRACT and TONYPLOT statements that specify how to handle the solutions calculated in the ATLAS simulation.

a. *Extract*

The EXTRACT statement allows device parameters to be displayed in the run-time output. The EXTRACT statement operates on the previous solved curve or structure file and by default uses the currently open log file [13]. However, the name of a file to be used can be specified before the extraction routine by using the following form of code.

EXTRACT INITIAL INFILE="<filename>"

Then a series of EXTRACT statements can follow to display the desired device parameters. The following is an example of an EXTRACT statement used to extract the open-circuit voltage of the solar cell's I-V characteristics.

EXTRACT NAME="Voc" X.VAL from \

CURVE(v."anode," i."cathode") where Y.VAL=0.0

Not only will the results of the extraction commands be displayed in the run-time output but also will be by default stored in the file results.final. However, the results can be stored in a different file by specifying

it at the end of the EXTRACT statements. The following code is added on the end of each EXTRACT statement to save the extraction in a specific file.

    DATAFILE="<filename>"

The same file can be used to store the results of all EXTRACT statements.

b. *Tonyplot*

The TONYPLOT statement starts the graphical post-processor TonyPlot [13]. This statement gives the specific file to be graphically displayed in TonyPlot. For example, the following is a command used to plot the I-V curve.

    TONYPLOT IVcurve.dat

The dat file in this example was created from the EXTRACT commands as follows.

    EXTRACT NAME="iv" \

    CURVE(v."anode," i."cathode") OUTFILE="IVcurve.dat"

Besides plotting the results with TonyPlot, the log files are interpreted and plotted using Matlab script files.

8. Simulation Source Code

Appendix A contains the DeckBuild codes used for running ATLAS in this thesis. Although some parameters are changed for the single-layer and multijunction solar cells to produce different designs, the basic solar cells are included in the appendix. To maintain uniformity and promote simplicity and understanding, the codes are organized in the same general manner with the following sections:

- Constants

- Mesh

- Regions

- Electrodes

- Doping

- Material Properties

- Models

- Light Beam

- Solutions

- Outputs

B.   MATLAB

Matlab makes it possible to quickly test different solar cell models. Although the Silvaco software is capable of running independently, Matlab is used with it in this thesis. Matlab is capable of not only interpreting and using the results produced by ATLAS but also opening DeckBuild, changing lines of code, and running it after the changes have been made. Matlab scripts have been written by other researchers at the Naval Postgraduate School that open DeckBuild files, change settings in the solar cell design, and run multiple iterations to find the optimal design. Davenport wrote a Matlab script that iterates one design parameter for multiple runs of ATLAS [14]. Canfield modified Davenport's script to be able to iterate two parameters [7]. The modified script iterates one parameter for each iteration of a second parameter.

The author uses the atlasrun_param1_param2.m file that Canfield created with some modifications to fit this thesis. The Matlab file creates an Excel spreadsheet and saves the ATLAS results after the two parameters are finished iterating and running the corresponding simulations. This allows the programmer to examine the results in order to choose the parameter combination that produces the best solar cell model. Appendix B contains a copy of the modified atlasrun_param1_param2.m file for two variables iterated for this thesis. This appendix also contains the other Matlab scripts that are used to run the Silvaco software, interpret the results, and display them as desired.

THIS PAGE INTENTIONALLY LEFT BLANK

V. MODELING OF MULTIJUNCTION SOLAR CELL TEMPERATURE EFFECTS

When modeling a device's behavior using software simulation tools, the accuracy of the simulations is a concern. This has been the challenge for many researchers using Silvaco ATLAS to model solar cells. The validity of the research is based on having an accurate model, which can only really be verified with actual performance results.

A. VERIFICATION OF MULTIJUNCTION SOLAR CELL MODEL

Proprietors guard solar cell designs closely to prevent others from stealing their designs. Therefore, finding an exact design to model and verify the simulated results with the published ones is a difficult task.

1. Multijunction Solar Cell Model Verified

Tsutagawa was able to find an article about a dual-junction solar cell with specific design specifications and performance results. He used the data contained within this paper to validate Silvaco ATLAS as a modeling tool [6]. The published InGaP/GaAs solar cell design that Tsutagawa found and simulated is shown in Figure 24.
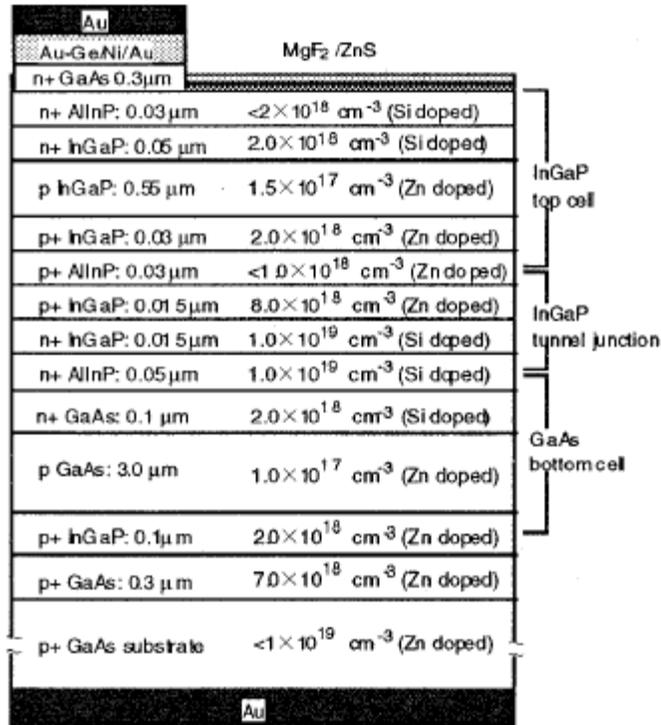
Figure 24.   Published Dual-Junction Design (From [15]).

Tsutagawa modeled the dual-junction solar cell with the specified materials and their exact thicknesses and doping levels at 25.3°C as specified in the article. His Silvaco ATLAS model of the published design is given in Figure 25.

Window	n+AlInP	0.03µm	$2e18\,cm^{-3}$
Emitter	n+InGaP	0.05µm	$2e18\,cm^{-3}$
Base	p+InGaP	0.55µm	$1.5e17\,cm^{-3}$
BSF	p+InGaP	0.03µm	$2e18\,cm^{-3}$
Buffer	p+AlInP	0.03µm	$1e18\,cm^{-3}$
Tunnel Emitter	p+InGaP	0.015µm	$8e18\,cm^{-3}$
Tunnel Base	n+InGaP	0.015µm	$1e19\,cm^{-3}$
Window	n+AlInP	0.05µm	$1e19\,cm^{-3}$
Emitter	n+GaAs	0.1µm	$2e18\,cm^{-3}$
Base	p+GaAs	3µm	$1e17\,cm^{-3}$
BSF	p+InGaP	0.1µm	$2e18\,cm^{-3}$
Buffer	p+GaAs	0.3µm	$7e18\,cm^{-3}$
Substrate	p+GaAs	300µm	$1e19\,cm^{-3}$

Top InGaP Layer — Tunnel Junction — Bottom GaAs Layer

Figure 25.   Model of Dual-Junction Design (From [6]).

After successfully modeling the dual-junction solar cell, Tsutagawa made a comparison of his simulated model and the results published in the article.

Table 3.   Published and Modeled Results (After [6]).

Parameter	Published	Modeled	Error
$V_{OC}(V)$	2.488	2.49172	0.15%
$J_{SC}(mA/cm^2)$	14.22	13.94890	1.91%
$V_{MAX}(V)$	N/A	2.30499	N/A
$J_{MAX}(mA/cm^2)$	N/A	13.68200	N/A
$P_{MAX}(mW/cm^2)$	N/A	31.53687	N/A
FF	85.6%	90.70%	5.96%
η	30.28%	31.54%	4.16%

As shown in Table 3, the Silvaco ATLAS model provided similar results to those published. Since the published and modeled results are based on terrestrial applications using AM1.5, Tsutagawa also simulated the performance of the solar cell for space applications using AM0.

Table 4.   Dual-Junction Model Results at AM0 (From [6]).

Parameter	Modeled
$V_{OC}(V)$	2.50490
$J_{SC}(mA/cm^2)$	19.12330
$V_{MAX}(V)$	2.29500
$J_{MAX}(mA/cm^2)$	18.72910
$P_{MAX}(mW/cm^2)$	42.98328
FF	89.70%
η	31.77%

The results he obtained are similar with the biggest difference being the increase in the current density. This increase is expected since the intensity of the light source is increased from AM1.5 to AM0. Tsutagawa's successful modeling of the dual-junction cell illustrates how well Silvaco ATLAS is at modeling multijunction solar cells.

2.   Multijunction Solar Cell Optimized

Once Tsutagawa had verified Silvaco ATLAS as an accurate means to model solar cells, he proceeded to develop a triple-juncition solar cell based on the dual-junction design he verified. Using the layers of the dual-junction as the top two layers, he added Ge to the bottom to complete his triple-junction cell. After running multiple simulations where he iterated the thicknesses and doping in each layer, he obtained an optimal design of the

triple-junction solar cell. He called his optimal design the champion cell [6]. The following diagram in Figure 26 contains the exact design of this multijunction cell.

Top InGaP Layer	Window	n+AlInP	0.01 μm	5e19 cm^{-3}
	Emitter	n+InGaP	0.17 μm	4.64e17 cm^{-3}
	Base	p+InGaP	0.63 μm	1e17 cm^{-3}
	BSF	p+InGaP	0.01 μm	5e19 cm^{-3}
	Buffer	p+AlInP	0.3 μm	1e18 cm^{-3}
Tunnel Junction	Tunnel Emitter	p+InGaP	0.015 μm	8e18 cm^{-3}
	Tunnel Base	n+InGaP	0.015 μm	1e19 cm^{-3}
Middle GaAs Layer	Window	n+AlInP	0.01 μm	4.64e17 cm^{-3}
	Emitter	n+GaAs	0.01 μm	4.64e15 cm^{-3}
	Base	p+GaAs	3.87 μm	1e17 cm^{-3}
	BSF	p+InGaP	0.01 μm	5e19 cm^{-3}
	Buffer	p+GaAs	0.3 μm	7e18 cm^{-3}
Tunnel Junction	Tunnel Emitter	p+GaAs	0.015 μm	8e18 cm^{-3}
	Tunnel Base	n+GaAs	0.015 μm	1e19 cm^{-3}
Bottom Ge Layer	Window	n+GaAs	0.05 m	7e18 cm^{-3}
	Emitter	n+Ge	0.1 μm	3e18 cm^{-3}
	Substrate	p+Ge	300 μm	3e18 cm^{-3}

Figure 26.   Champion Cell Design (From [6]).

All the electrical characteristics that ATLAS produced for this solar cell design are given in Table 5.

Table 5.   Champion Cell Performance (After [6]).

Parameter	Result
$V_{OC}(V)$	2.76425
$J_{SC}(mA/cm^2)$	19.85310
$V_{MAX}(V)$	2.52500
$J_{MAX}(mA/cm^2)$	19.44190
$P_{MAX}(mW/cm^2)$	49.09080
FF	89.45%
η	36.28%

After optimizing the cell, Tsutagawa was able to report unprecedented results at that time. Most significant was designing a triple-junction solar cell with an efficiency of 36.28% at 300 K for space applications.

B.   EFFECT OF HIGHER TEMPERATURE OPERATION

The focus of this thesis was to determine the effect of temperature on the performance of a solar cell. Although Tsutagawa simulated the "champion cell" at several different temperatures, he conducted these simulations only after optimizing the cell for operation at 300 K [6]. Often solar cells are required to operate at much higher temperatures than room temperature; therefore, it is beneficial to consider the operation of a solar cell at elevated temperatures in the design process.

1.   Temperature Effect on Triple-junction Cell

First, the author simulated Tsutagawa's champion cell at 275, 300, 325, 350, 375, and 400 K. The results of these simulations are given in Table 6.

Table 6.   Champion Cell Temperature Performance.

Parameter	275 K	300 K	325 K	350 K	375 K	400 K
V_{OC}(V)	2.88	2.76	2.64	2.52	2.39	2.27
J_{SC}(mA/cm^2)	19.85	19.85	19.85	19.85	19.85	19.85
V_{MAX}(V)	2.66	2.53	2.4	2.25	2.10	2.00
J_{MAX}(mA/cm^2)	19.49	19.44	19.30	19.36	19.39	19.01
P_{MAX}(mW/cm^2)	51.74	49.09	46.32	43.56	40.73	38.02
FF	90.35%	89.45%	88.31%	87.10%	85.74%	84.53%
η	38.24%	36.28%	34.24%	32.19%	30.10%	28.10%

As the results show in Table 6, the efficiency goes down about 2% for every 25-K increase in the temperature. The plot of the efficiency versus temperature in Figure 27 is a visual illustration of this.
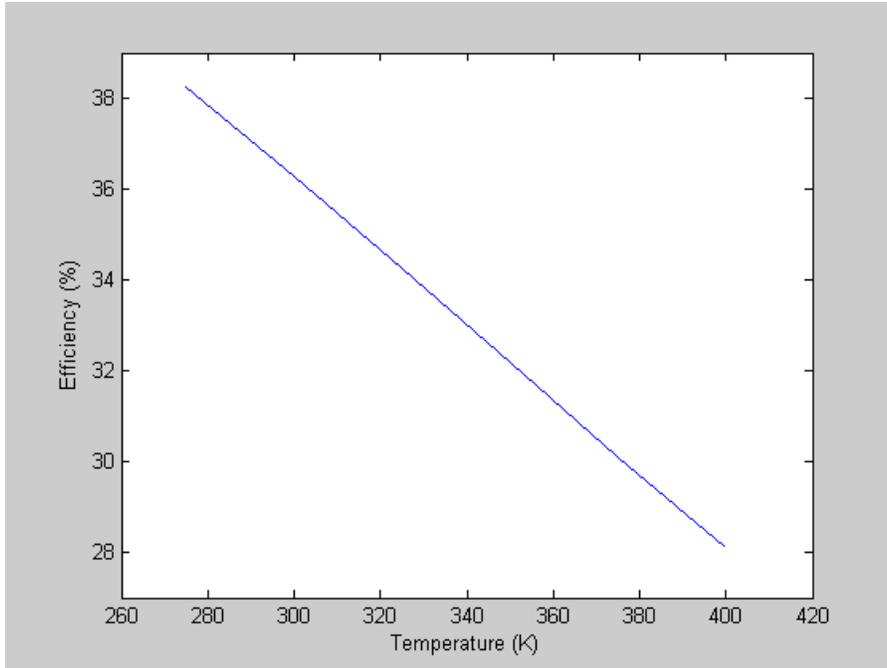
Figure 27.   Efficiency Versus Temperature.

As a comparison, consider a solar cell developed by Emcore. The following graph shows the rate at which the efficiency of one of its triple-junction InGaP/InGaAs/Ge

63

cells, which is up to 37% efficient under concentrated illumination, drops as the operating temperature increases from about 300 to 350 K [16].
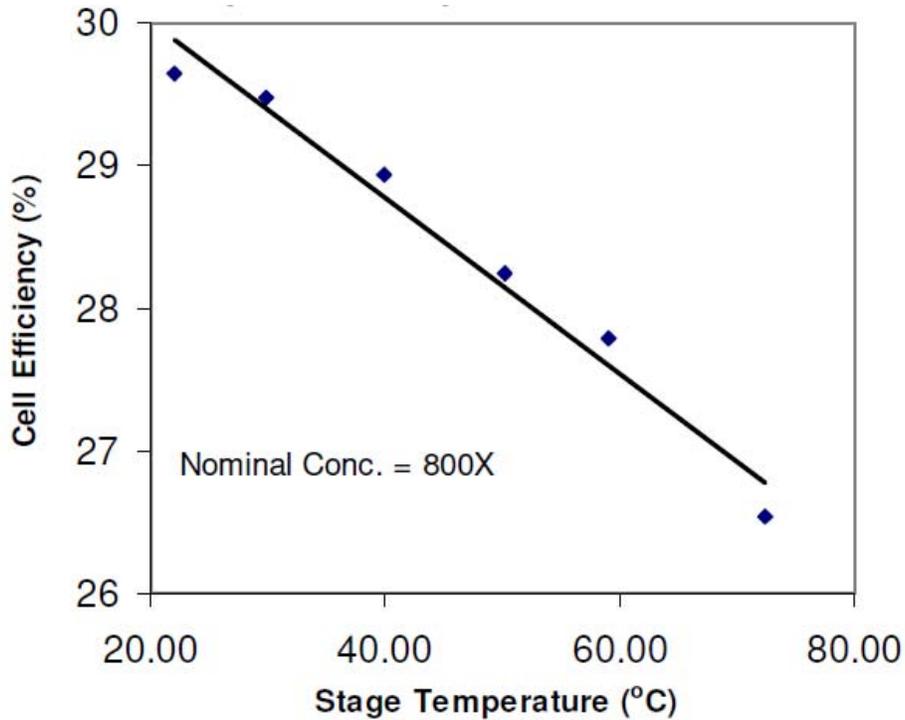
Figure 28.   Emcore Photovoltaics Cell Efficiency Versus Temperature (From [16]).

The results presented in the two preceding graphs make it clear that the effect of temperature is very similar on these two cells since the slopes of the two lines—0.0818 %/°C in Figure 27 and 0.0625 %/°C in Figure 28—are close over a similar temperature range. The drop in efficiency for Tsutagawa's champion cell equates to about a 0.11 $mW/cm^2/K$ decrease in maximum power, which is shown in the plot presented in Figure 29.
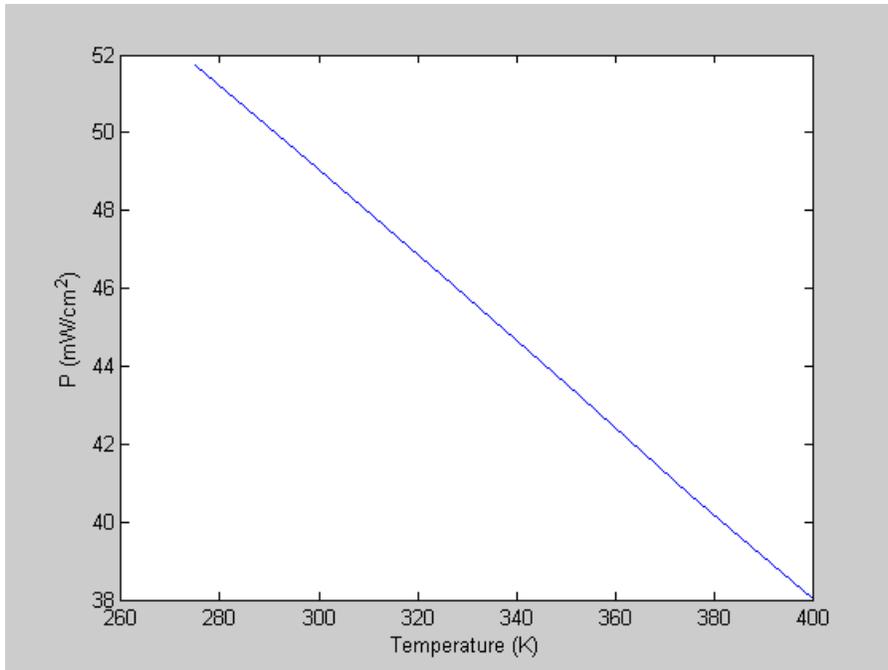
Figure 29.   P$_{MAX}$ Versus Temperature.

The calculation for the change in maximum power versus temperature agrees with the derivation given by the Photovoltaic Systems Department of Sandia National Laboratories, a Lockheed Martin Company [17].

$$\frac{dP_{mp}}{dT} = V_{mp} \cdot \frac{dI_{mp}}{dT} + I_{mp} \cdot \frac{dV_{mp}}{dT} \qquad (3.1)$$

The reason for the decrease in solar cell efficiency and maximum power is due mainly to the drop in the voltage, which is on average 4.8 mV/K for the open-circuit voltage and 5.28 mV/K for the voltage at maximum power. Plots of these two parameters versus temperature are given in Figure 30.

65
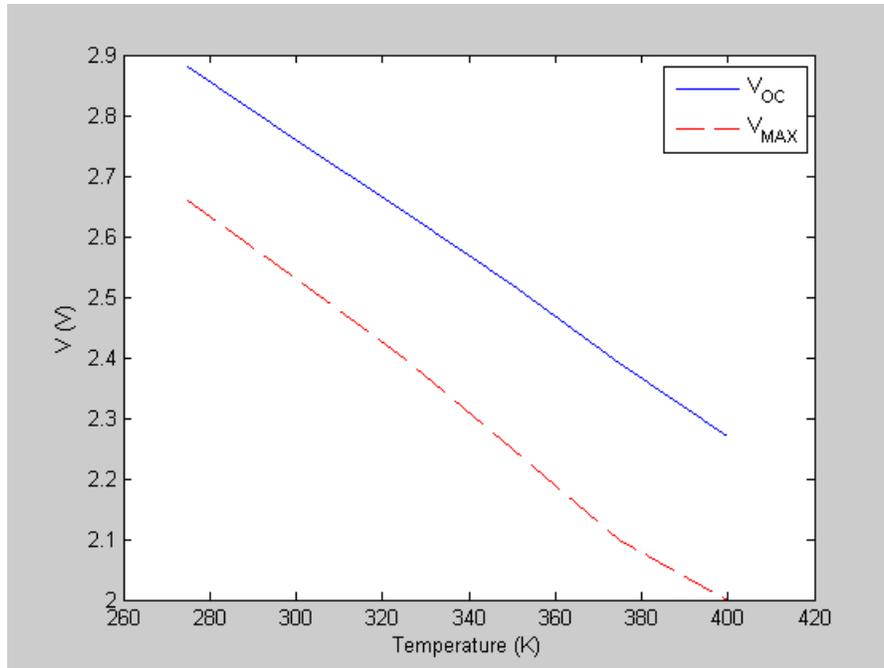
Figure 30. V_{OC} and V_{MAX} Versus Temperature.

This drop in voltage is not a surprise, since it is expected to be around 2 mV/K for the typical performance of a single-layer cell [10]. On the other hand, the short-circuit current remains constant and the current at maximum power only reduces slightly, as expected, and is illustrated in the plots of these two parameters shown in Figure 31.
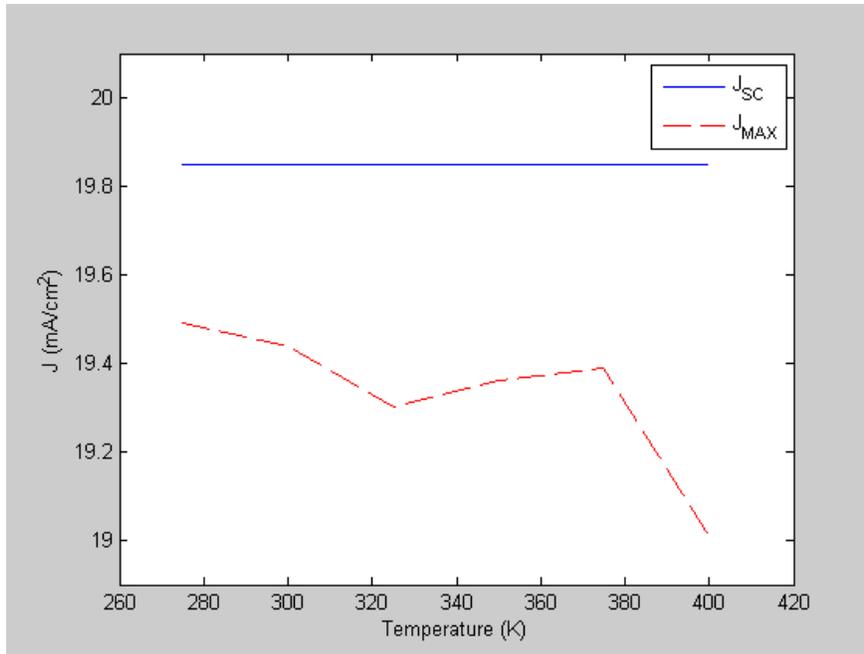
Figure 31. J_{SC} and J_{MAX} Versus Temperature.

In addition, the FF drops on average 1.164% for every 25-K increase in the temperature following a similar decrease rate as the voltage at maximum power. A plot of the FF versus temperature is presented in Figure 32.

67
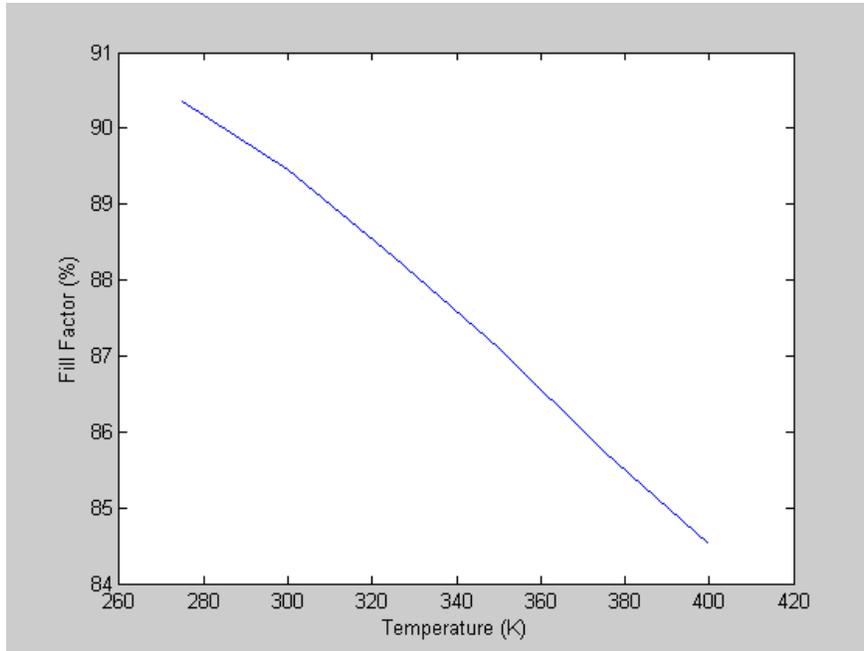
Figure 32. Fill Factor Versus Temperature.

2.   Realistic Operating Temperatures

The results of the temperature simulations are significant since solar cells are required to operate at elevated temperatures and not near room temperature, or 300 K. First, the operating temperature of solar cells is higher than the ambient temperature. For example, consider the results of one study conducted on Si-based solar cells with different surface materials. Plots of the results for this study are given in Figure 33.

68
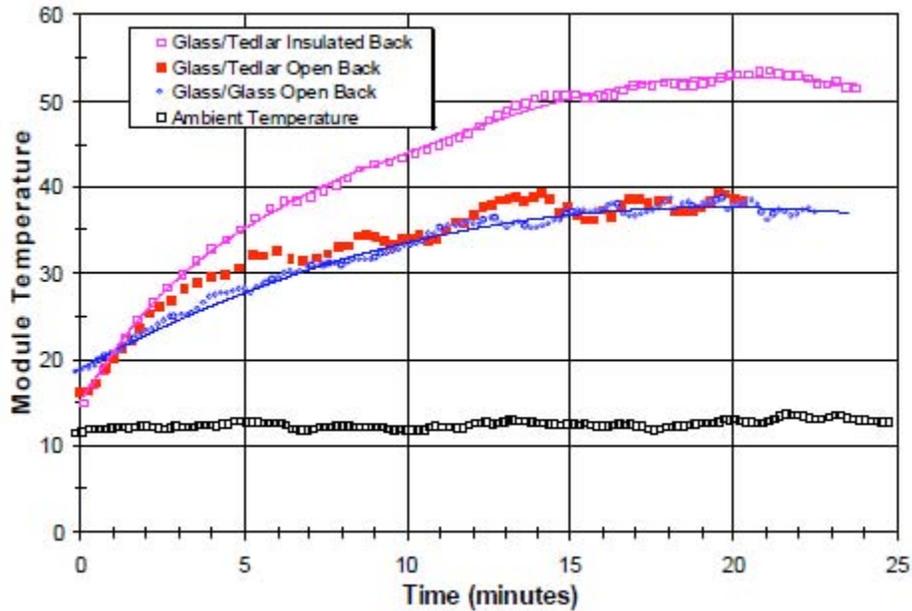
Figure 33.   Operating Temperature Versus Time (From [17]).

The graph shows how the operating temperature of these cells started at a rather low ambient temperature and rose to a higher quasi-steady operating temperature. As indicated by these results, the actual operating temperature of solar cells can be as much as 40°C above the ambient temperature. Even solar cells used in terrestrial applications at room temperature will have an operating temperature much higher. However, many terrestrial solar cells are used in very hot climates where the ambient temperature is higher, resulting in even higher operating temperatures.

Solar cells in space applications are often exposed to even higher temperatures than those in terrestrial applications and, therefore, have higher operating temperatures. The graphs in Figure 34 show the typical temperatures of solar arrays in Low Earth Orbit (LEO) and Geosynchronous Earth Orbit (GEO).
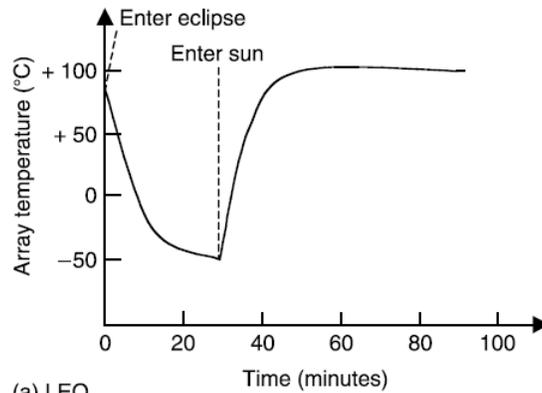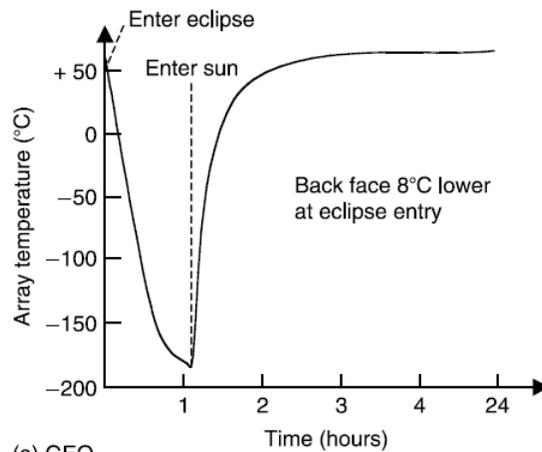
Figure 34. Solar Array Temperature at LEO and GEO (From [18]).

As illustrated in the above graphs, the ambient temperature of solar cells in LEO and GEO are around 100°C and 60°C, respectively, when not in a solar eclipse. One should expect the actual operating temperature of solar cells used in these orbits to be even higher if directed at the sun constantly as they are when used on three-axis stabilized satellites.

Some have developed techniques to counter the effect of the higher operating temperatures. Although techniques and devices exists to actively cool solar cells while in

use, such as solid-state refrigerators, these, in general, require more power to operate than the resultant gain in efficiency [19]. Solar cells used in space for applications closer to the sun, such as probes sent to Mercury, can expect to operate in temperatures that exceed 200°C [20]. For these space missions closer to the sun, techniques that serve to shield the cells from the high temperatures by reflecting the incident solar energy away not only reduces the operating temperature but also reduces the performance of the cells [19]. Since the temperature at which solar cells usually operate is much higher than room temperature, the best solution is to design them to give their best performance at these temperatures.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. TEMPERATURE AND OPTIMIZATION OF SOLAR CELLS

Since solar cells frequently operate at higher temperatures than the temperature for which they are optimally designed, it is beneficial to examine the effect temperature has on the design of the solar cells. Observing the effect temperature has on the design can lead to a better design for solar cells operating at higher temperatures.

A. THE EFFECT OF TEMPERATURE ON SOLAR CELL DESIGN

First, the author examines the effect of temperature on the original design of the InGaP/GaAs/Ge triple-junction subcells. Second, the author varies the design of the subcells and tests them at various temperatures to see the effect this has on the performance of the cells. The results are analyzed to see if the cells can be redesigned to produce better results for cells that are expected to operate at elevated temperatures.

1. Temperature Effect on Triple-junction Subcells

After finding the effect of temperature on the triple-junction cell as a whole, the author separates the cell into its individual layers—InGaP, GaAs, and Ge—and tests each layer at these same temperature settings. Tables 7-9 contain the simulation results, rounded to the nearest thousandth, for the three layers.

73

Table 7. InGaP Temperature Performance.

Parameter	275	300	325	350	375	400
$V_{OC}(V)$	1.47	1.42	1.37	1.33	1.28	1.23
$J_{SC}(mA/cm^2)$	19.85	19.85	19.85	19.85	19.85	19.85
$V_{MAX}(V)$	1.35	1.30	1.25	1.20	1.15	1.10
$J_{MAX}(mA/cm^2)$	19.35	19.29	19.18	19.05	18.91	18.75
$P_{MAX}(mW/cm^2)$	26.12	25.07	23.97	22.86	21.74	20.62
FF	89.73%	88.96%	87.98%	86.90%	85.77%	84.57%
η	19.30%	18.53%	17.72%	16.90%	16.07%	15.24%

Table 8. GaAs Temperature Performance.

Parameter	275	300	325	350	375	400
$V_{OC}(V)$	1.11	1.08	1.05	1.01	0.97	0.94
$J_{SC}(mA/cm^2)$	39.65	39.32	38.73	37.86	36.75	35.46
$V_{MAX}(V)$	0.97	0.92	0.87	0.85	0.80	0.75
$J_{MAX}(mA/cm^2)$	37.85	37.18	36.21	34.12	33.18	32.16
$P_{MAX}(mW/cm^2)$	36.53	34.21	31.68	29.00	26.54	24.12
FF	82.65%	80.54%	78.26%	75.86%	74.19%	72.59%
η	27.00%	25.28%	23.42%	21.44%	19.62%	17.83%

Table 9. Ge Temperature Performance.

Parameter	275	300	325	350	375	400
$V_{OC}(V)$	0.34	0.30	0.27	0.23	0.19	0.15
$J_{SC}(mA/cm^2)$	71.85	71.78	71.71	71.65	71.59	71.52
$V_{MAX}(V)$	0.28	0.24	0.21	0.18	0.15	0.10
$J_{MAX}(mA/cm^2)$	65.79	65.35	63.52	57.42	51.77	57.10
$P_{MAX}(mW/cm^2)$	18.42	15.69	13.02	10.34	7.77	5.71
FF	75.69%	72.24%	68.28%	63.03%	58.50%	52.56%
η	13.61%	11.59%	9.62%	7.64%	5.74%	4.22%

Each layer of the triple-junction cell performs in a similar fashion as the temperature increases. The graphs in Figures 35-41 are visual representations of the data given in Tables 7-9.
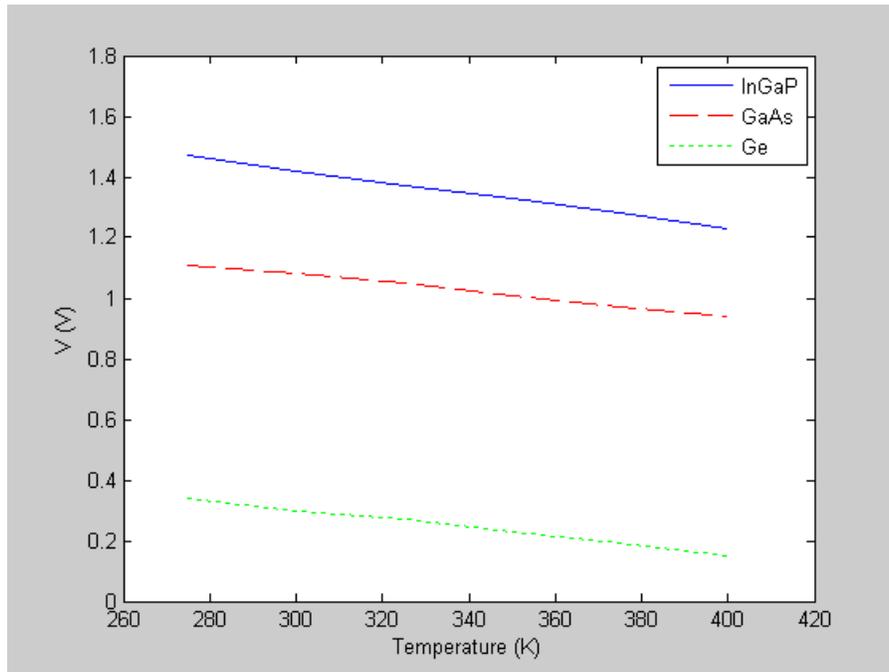
Figure 35. Champion Subcells' V_{OC} Versus Temperature.
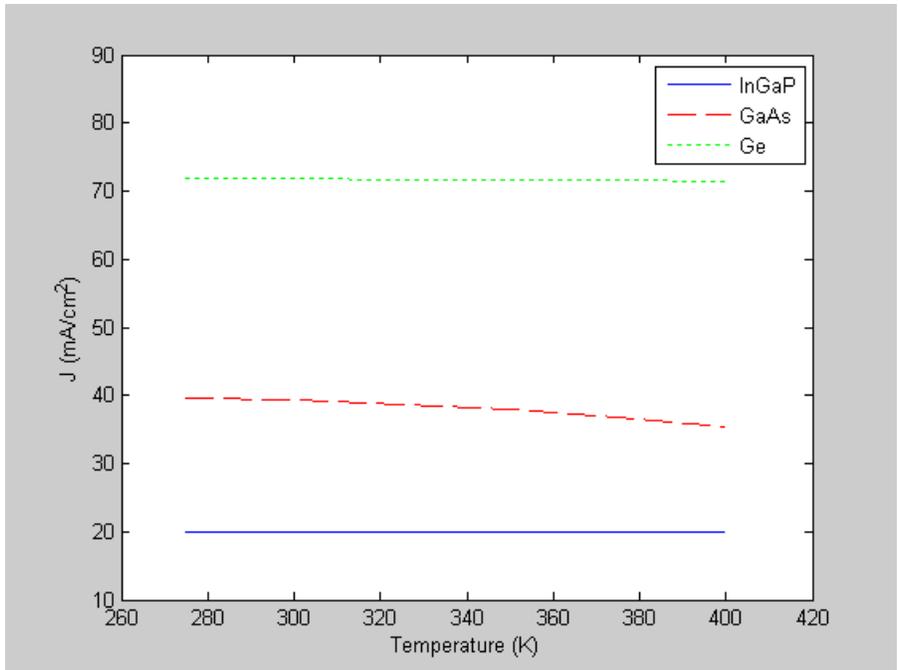
Figure 36. Champion Subcells' J_{SC} Versus Temperature.
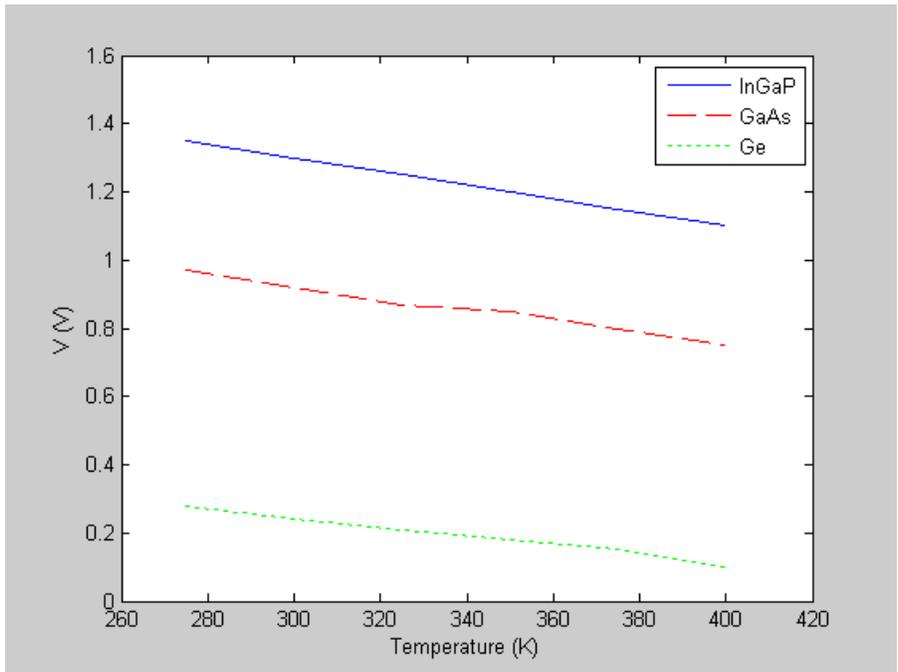
Figure 37. Champion Subcells' V_{MAX} Versus Temperature.
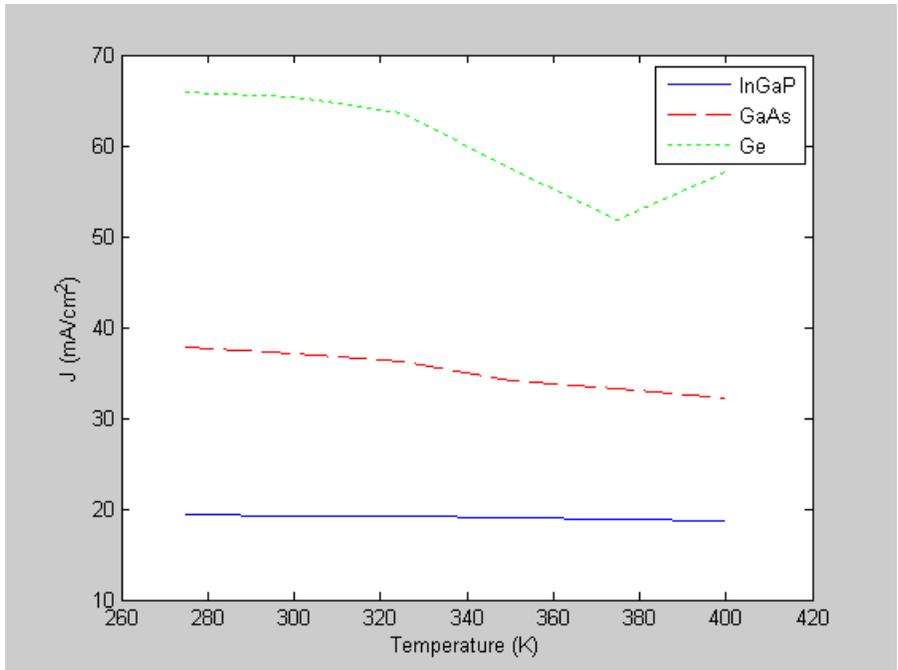
Figure 38.   Champion Subcells' J_{MAX} Versus Temperature.
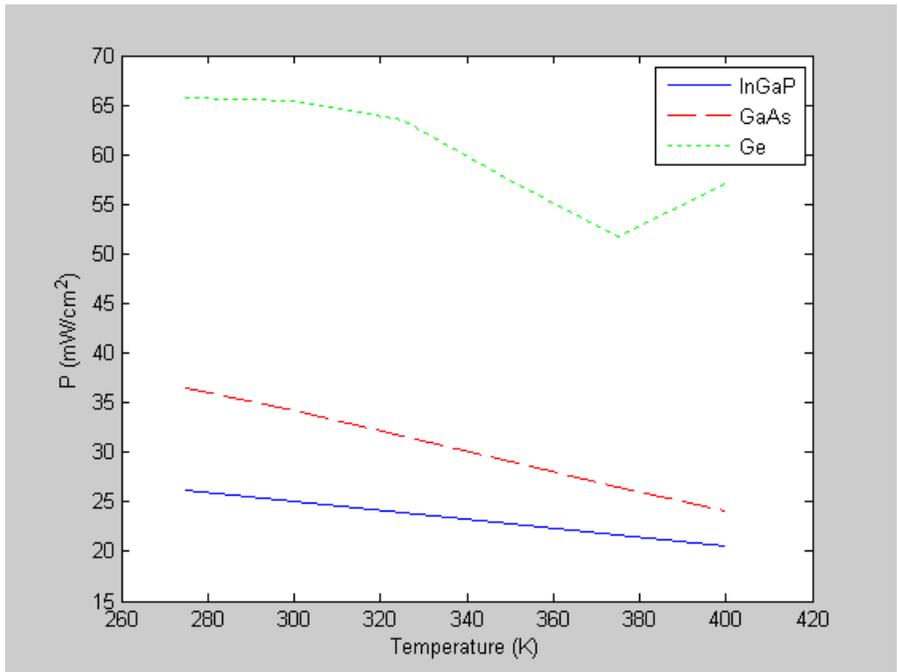
Figure 39.   Champion Subcells' P_{MAX} Versus Temperature.

77
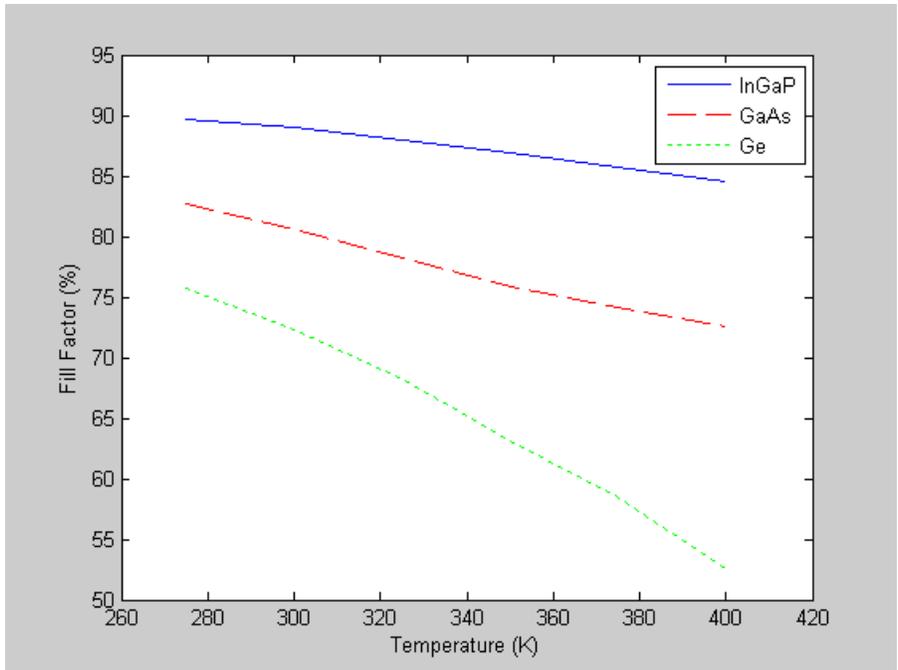
Figure 40.   Champion Subcells' FF Versus Temperature.
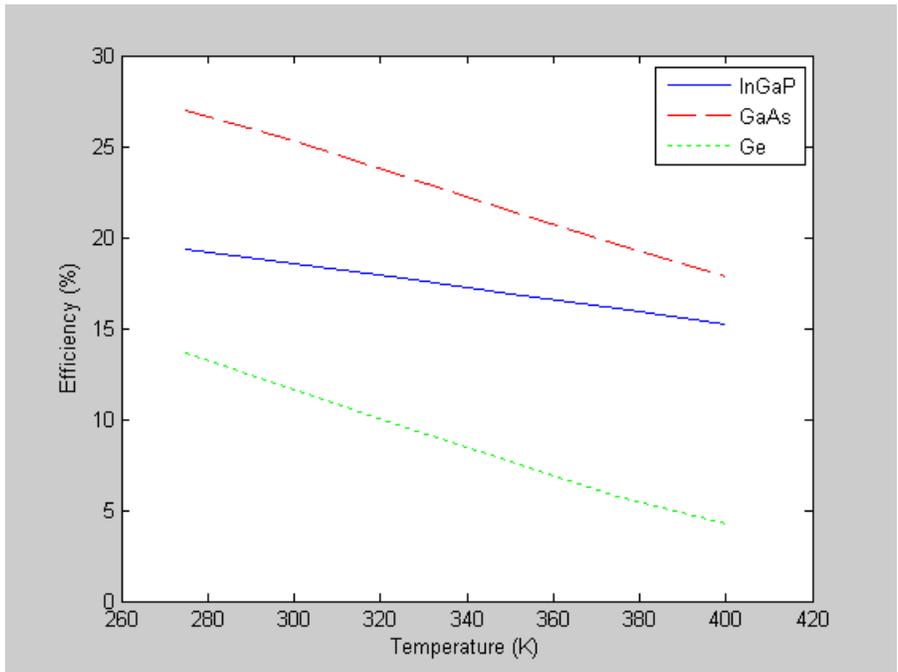
Figure 41.   Champion Subcells' Efficiency Versus Temperature.

These plots show that most of the parameters followed similar trends for each subcell with only a few differences. The differences are mainly due to one unusual J_{MAX} result for the Ge subcell, which causes a dip in the plot of this parameter at 375 K and affects the calculations for P_{MAX}, FF, and η for this subcell. A summary of how each parameter changes as the temperature increases for the triple-junction cell and each of its subcells is given in Table 10.

Table 10.   Triple and Subcell Temperature Coefficients.

Parameter Change	Rate of Decrease			
	Triple	InGaP	GaAs	Ge
V_{OC}(mV/K)	4.9	1.9	1.4	1.5
J_{SC}(μA/cm^2/K)	0.0	0.0	33.5	2.6
V_{MAX}(mV/K)	5.2	2.0	1.7	1.4
J_{MAX}(μA/cm^2/K)	3.8	4.8	45.5	69.5
P_{MAX}(μW/cm^2/K)	109.8	44.0	99.2	101.7
FF(%/K)	0.0005	0.0004	0.0008	0.0019
η(%/K)	0.0008	0.0003	0.0007	0.0008

The temperature coefficients in Table 10 are the average rate at which each parameter decreases as the temperature increases.

Although the changes in the parameters due to the temperature are similar for each cell, a few observations are worth mentioning. For the triple-junction cell, the drop in V_{OC} and V_{MAX} is approximately equal to the sum of the drops of each layer, as expected, since each layer contributes to the overall decrease. In another study of a triple-junction InGaP/InGaAs/Ge cell tested at 30°C to

79

100°C, the open-circuit voltage coefficient is 6.0 mV/K. For a similar temperature range, this is only a 1.1 mV/K difference, which is likely due to the actual design of the cells [20].

Although the decrease in J_{SC} for each layer and the triple-junction is negligible, the GaAs layer's J_{SC} decreases the most. This is significant because it suggests that improvement to the triple-junction cell may be made by improving this layer of the cell so that this current does not decrease as much. The maximum current and fill factor are two parameters that have similar trends: the triple-junction cell has the lowest and the rate of decrease increases from the top subcell to the bottom subcell. This is important because it implies that these parameters get worse as one gets deeper in the triple-junction cell. However, it may be that this is due to optimizing each layer to function as a part of the whole cell and the bottom layers are more affected when layers above them are removed. After all, the designer of this cell optimized it as a triple-junction cell and not as three individually optimized cells stacked on each other. The higher temperature coefficients for the middle and bottom layers may also indicate that these subcells have been designed to generate sufficient current at all temperatures in the range and not become the current limiting layer [21]. Since the decrease in the voltage remains almost constant for each layer, the increasing rate of decrease for P_{MAX} and, therefore, η is due to the increasing rate of J_{MAX} decrease. The rate of decrease of P_{MAX} for the triple-junction is more

80

than any one layer because the decrease in V_{MAX} for the triple-junction is a result of summing the voltage drop contribution made by each layer.

2.   Design Iterations of Triple-junction Subcells

Next, the author varied some of the design specifications to see how they affected the performance of each subcell. Specifically, the author varied the emitter thickness, emitter doping, base thickness, and base doping to generate multiple solar cell simulations at each temperature setting. Most of these design parameters are varied one decade above and below the original design thicknesses and doping levels to observe the temperature effect of these settings in the simulations.

Plots of the results from varying the design of the Ge subcell are given in Figures 42–49.
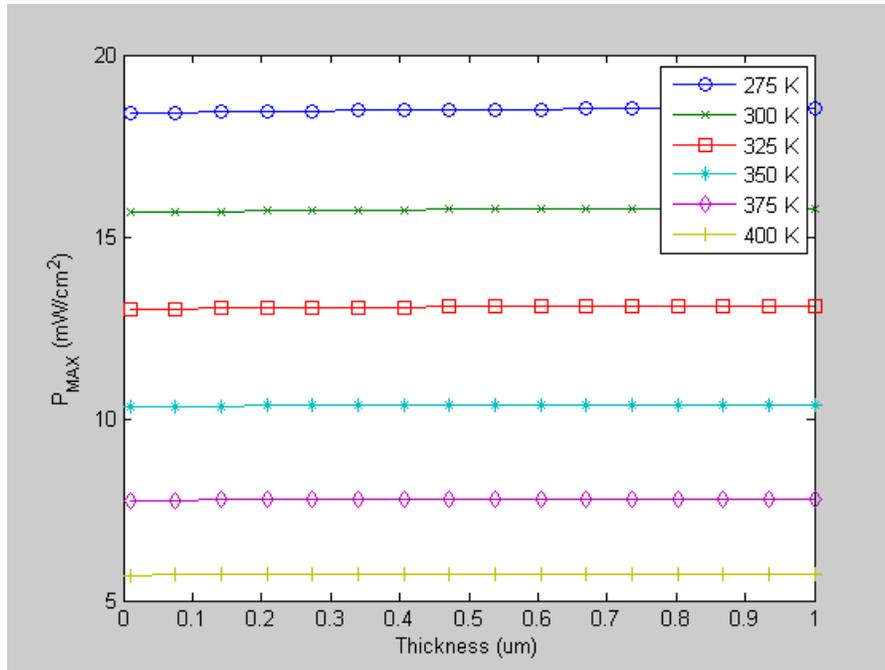
Figure 42.   Ge - P_{MAX} Versus Emitter Thickness (275–400 K).
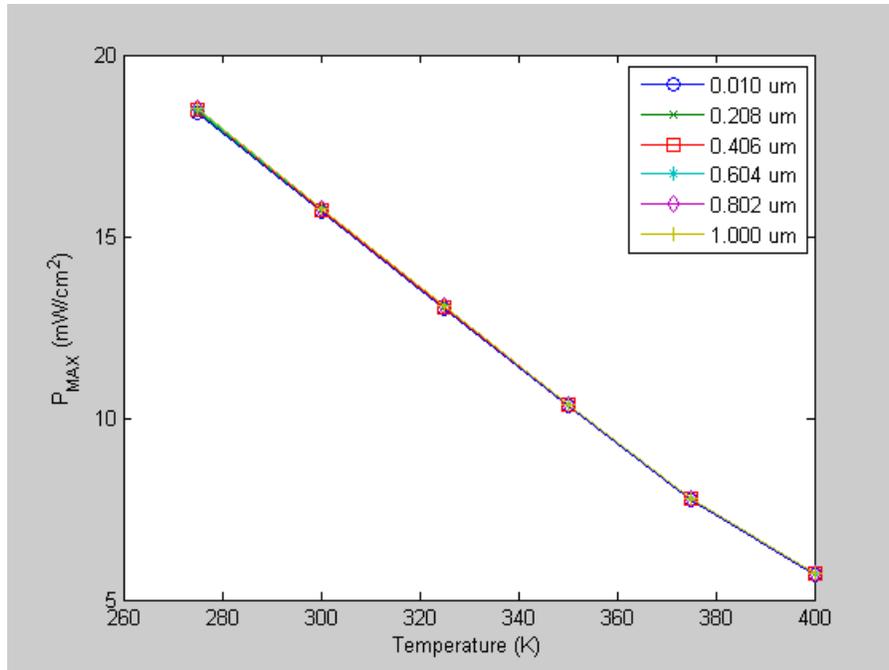
Figure 43. Ge Emitter Thickness - P_{MAX} Versus Temperature.

The two plots in Figures 42 and 43 indicate that the emitter thickness of Ge has little bearing on the performance of this subcell. However, as expected, the performance drops significantly, as the temperature increases.
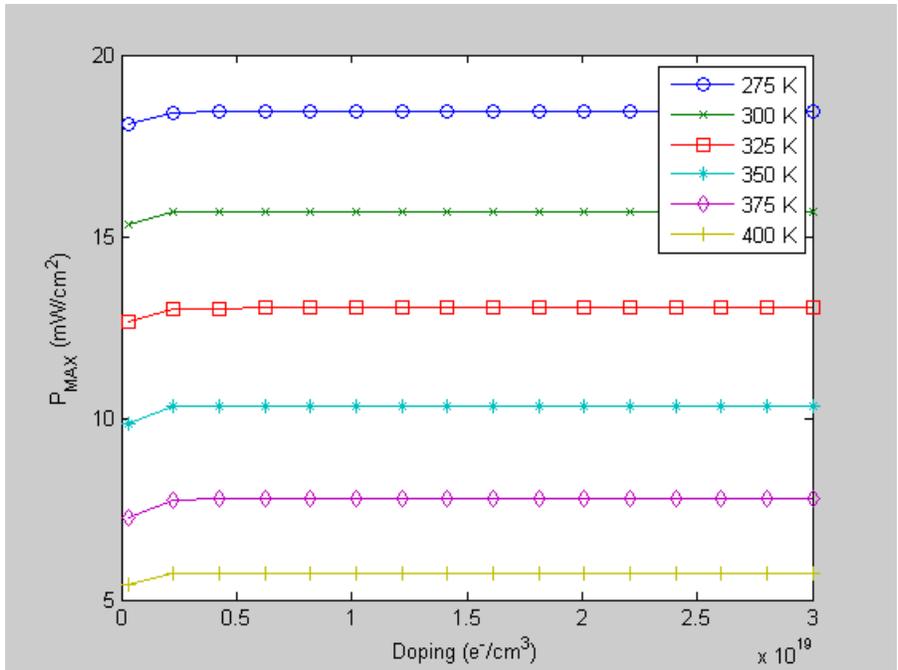
Figure 44. Ge - P_{MAX} versus Emitter Doping (275–400 K).
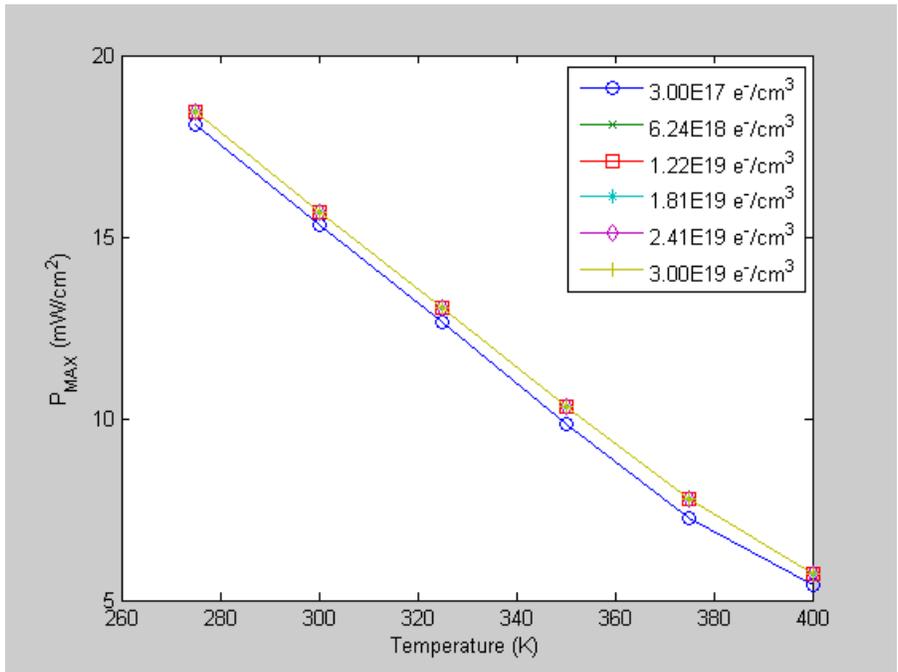
Figure 45. Ge Emitter Doping - P_{MAX} versus Temperature.

Like the emitter thickness of Ge, the Ge emitter doping has little effect based on the two plots presented in Figures 44 and 45, although the lowest doping tested gives a lower output than the other doping levels tested. Again, as expected, the temperature causes the performance to drop significantly.
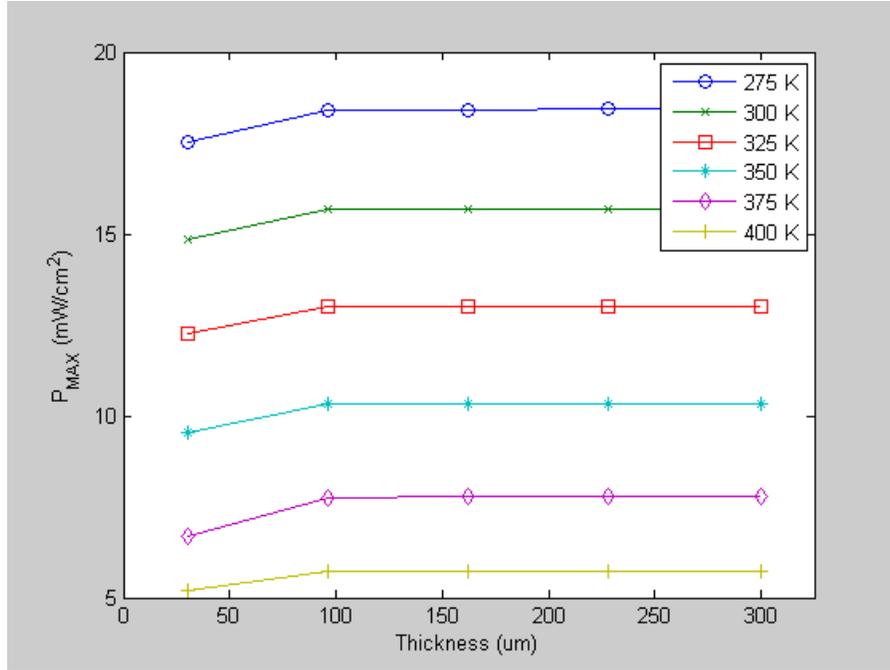
Figure 46. Ge - P_{MAX} versus Base Thickness (275-400 K).
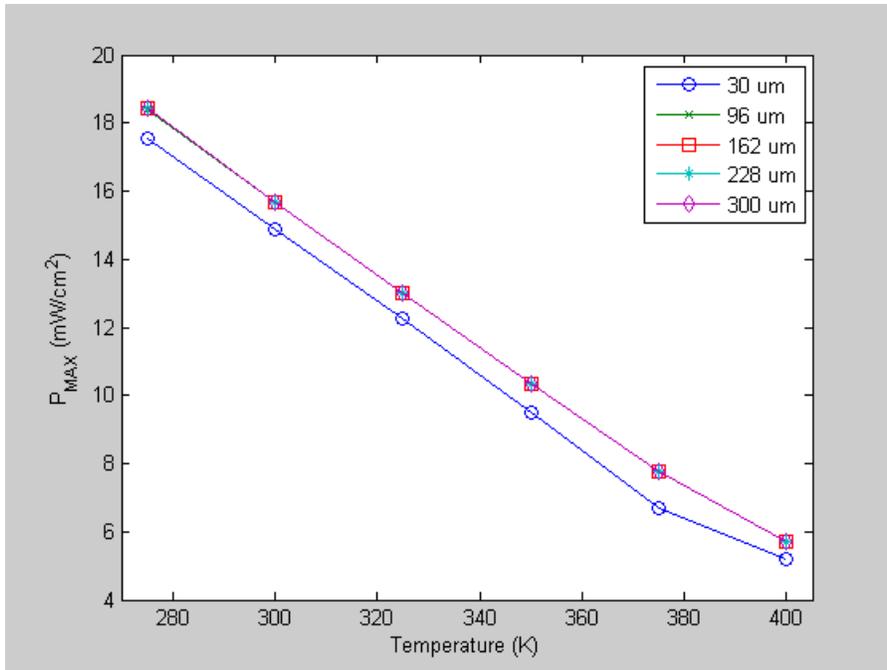
Figure 47.   Ge Base Thickness - P_{MAX} versus Temperature.

        Considering the plots in Figures 46 and 47, it is
clear that, like the two previous design parameters of Ge,
the Ge base thickness has little effect on the performance
of this subcell. However, the lowest thickness tested gives
a slightly lower output than the other thicknesses tested.
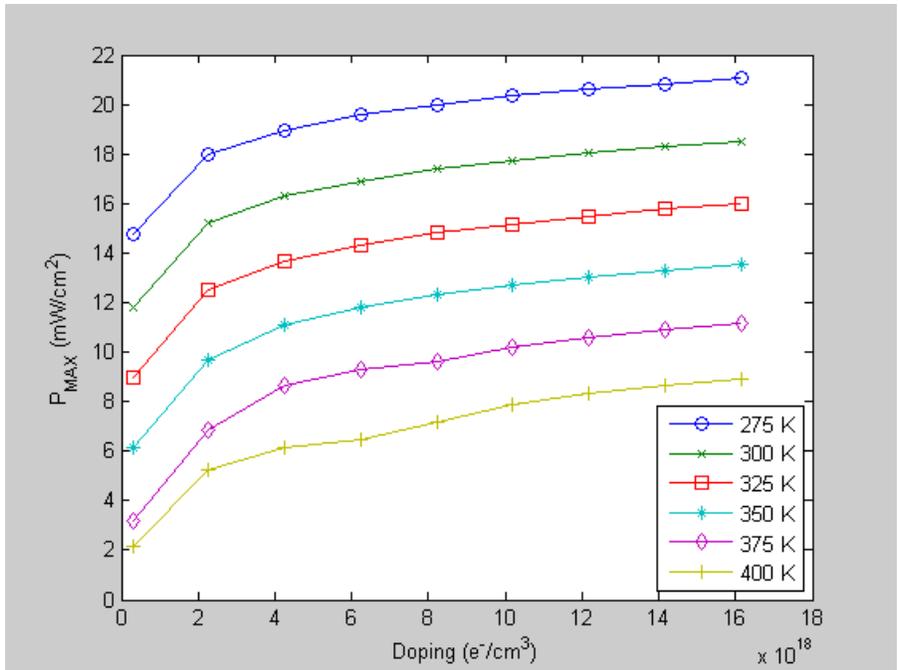
Figure 48.   Ge - P_{MAX} versus Base Doping (275-400 K).
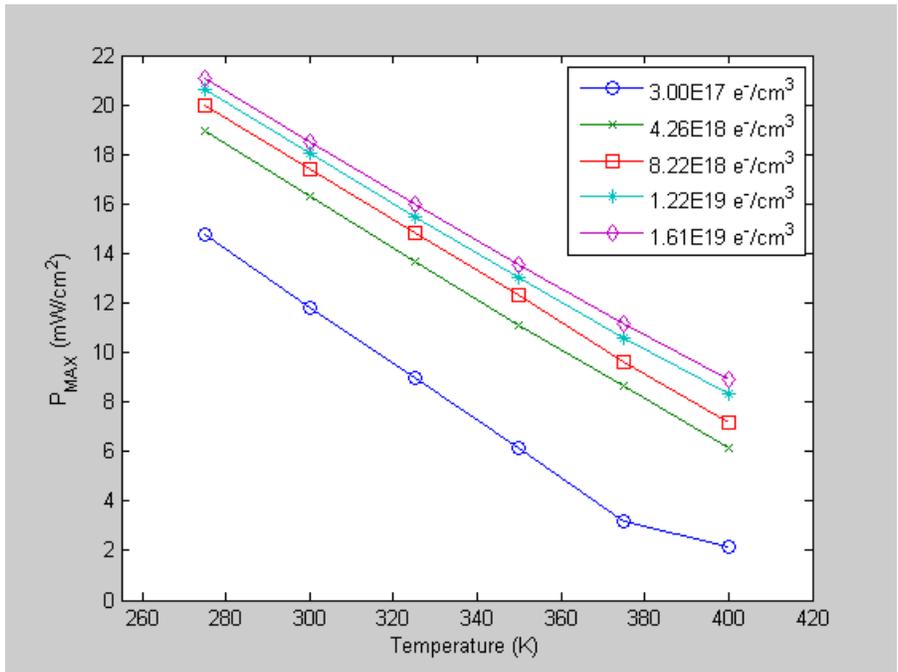
Figure 49.   Ge Base Doping - P_{MAX} versus Temperature.

Based on the two plots given in Figures 48 and 49, the Ge base doping is the only design parameter of this subcell tested that shows a significant effect on the cell's performance. The output of the cell increases as the base doping level increases. The temperature continues to cause a significant decrease in performance.

The plots of the results from varying the design of the GaAs subcell are presented in Figures 50–57.
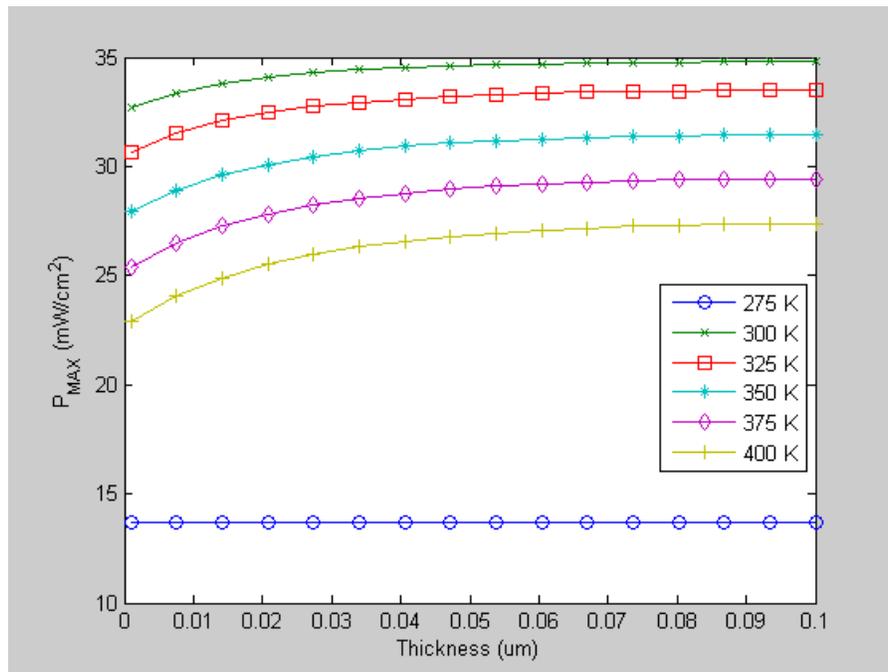
Figure 50.   GaAs – P_{MAX} versus Emitter Thickness (275-400 K).
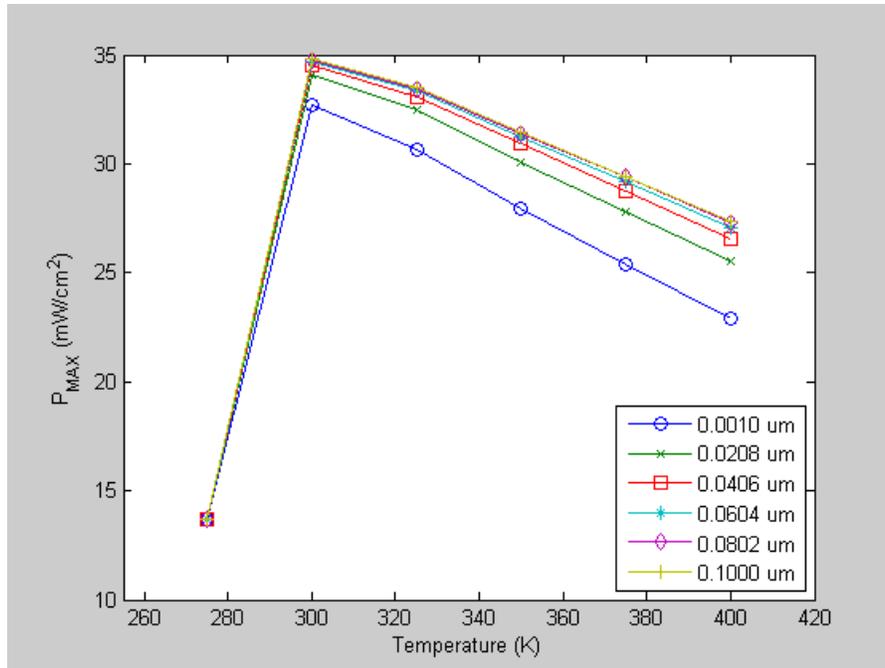
Figure 51. GaAs Emitter Thickness - P_{MAX} versus Temperature.

The results plotted in Figures 50 and 51 indicate that the emitter thickness of GaAs has some effect on the performance of this subcell, depending on the temperature. The plots show that the designer of this subcell optimizing its performance at 300 K, since the highest output is at this temperature and decreases as the temperature increases. However, the output is lowest at 275 K. Generally, as indicated in the plot shown in Figure 51, the thicker this subcell is the better it performs, but this effect is less significant at the greatest thicknesses tested.
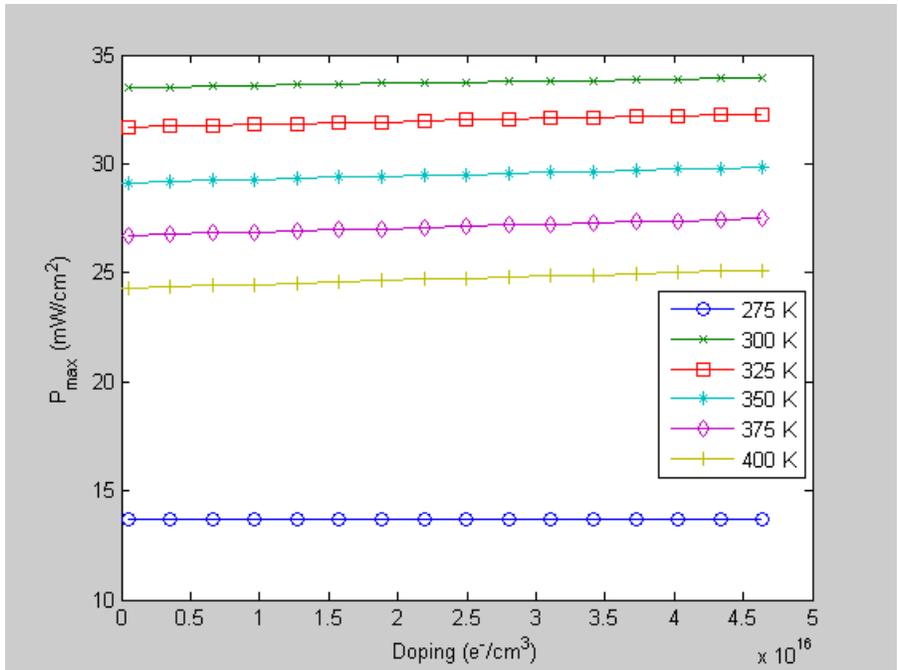
Figure 52.   GaAs - P$_{MAX}$ versus Emitter Doping (275-400 K).
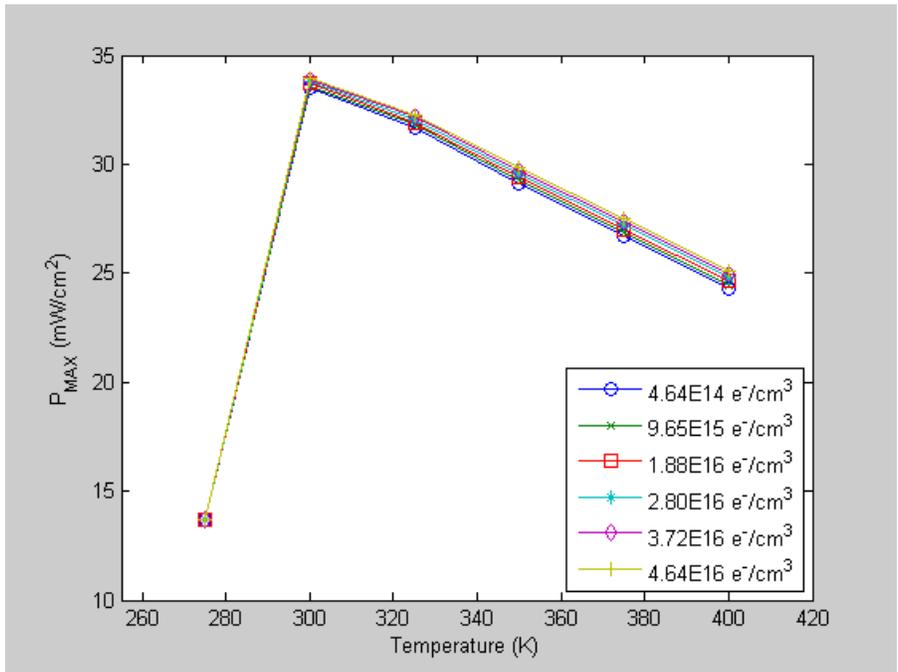
Figure 53.   GaAs Emitter Doping - P$_{MAX}$ versus Temperature.

Similar to the emitter thickness of GaAs, the results plotted in Figures 52 and 53 indicate that the emitter doping of GaAs has some effect on the performance of this subcell but a little less than the emitter thickness. As shown in these plots, the emitter doping contributes to this subcell's optimal design at 300 K. The output decreases as the temperature increases except at 275 K, where it is at its lowest. Generally, as with the emitter thickness, a higher doping level for this subcell means better performance; however, this effect is less significant than it is for the emitter thicknesses tested.
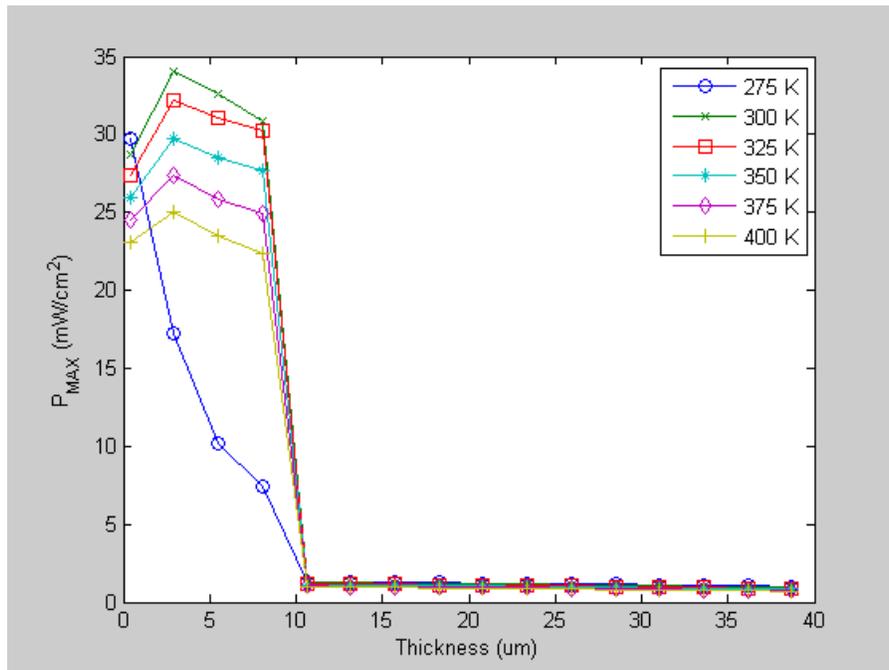
Figure 54.   GaAs – P_{MAX} versus Base Thickness (275–400 K).
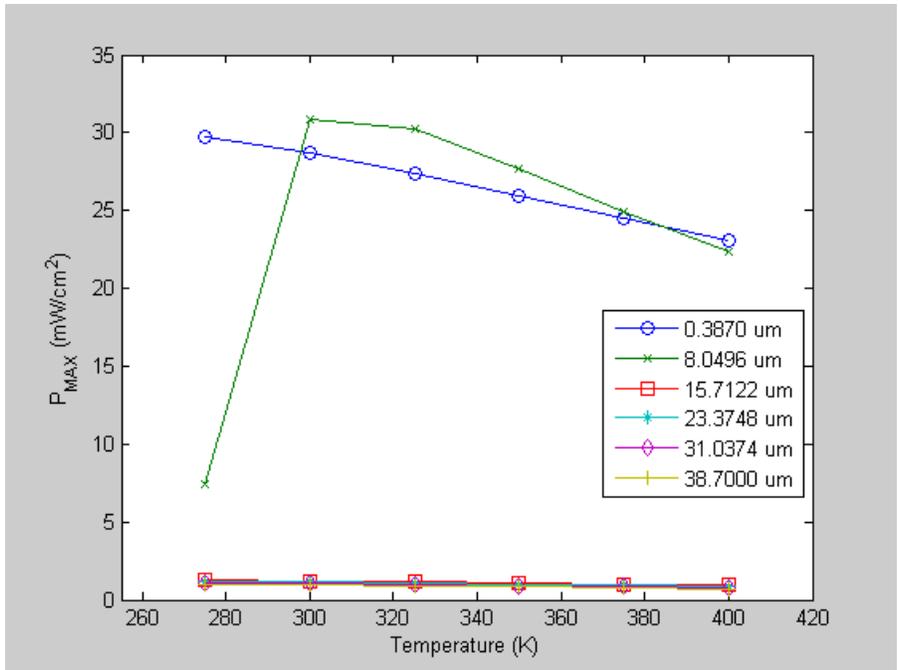
Figure 55. GaAs Base Thickness - P_{MAX} versus Temperature.

The GaAs base thickness is even more important in influencing the performance of this subcell. The results shown in Figures 54 and 55 indicate, again, that this subcell has been effectively optimized at 300 K. However, the results also indicate that the highest output occurs when it has a similar thickness at the higher temperatures.
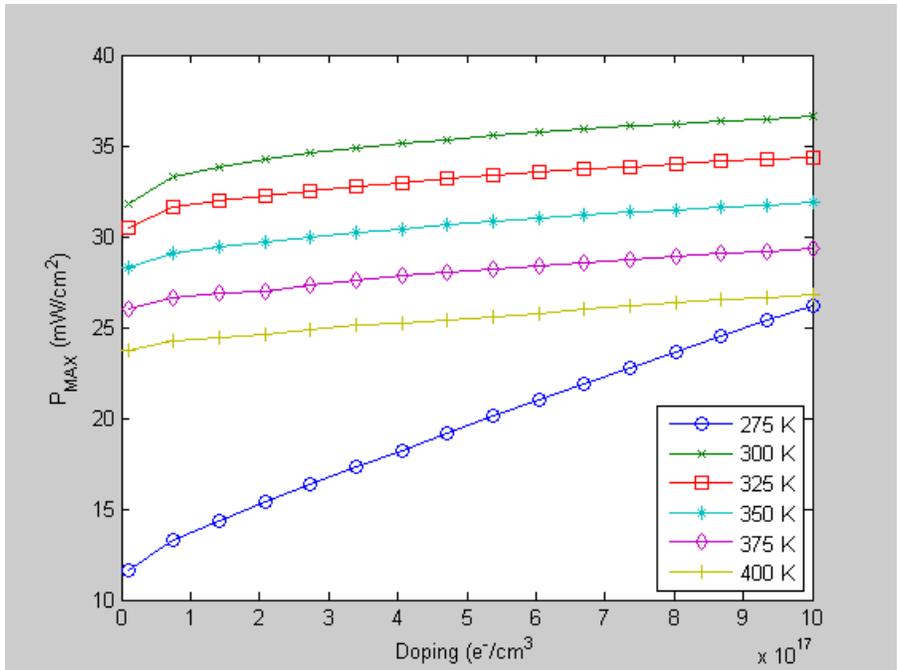
Figure 56.   GaAs - P_{MAX} versus Base Doping (275-400 K).
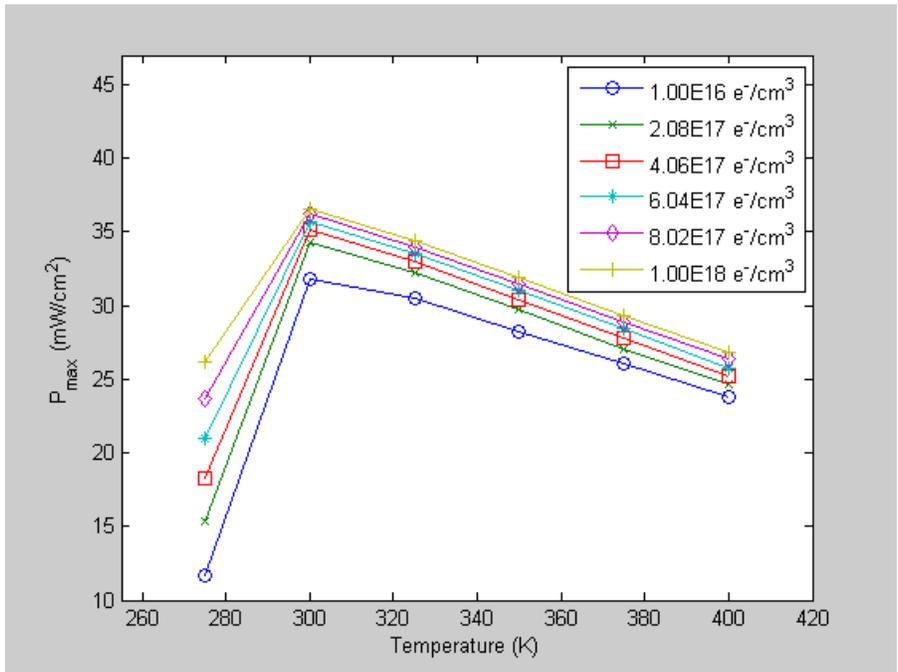
Figure 57.   GaAs Base Doping - P_{MAX} versus Temperature.

The GaAs output increases as the base doping level increases, as illustrated in the plots given in Figures 56 and 57. The temperature continues to cause a significant decrease in performance as it increases, although the output is lowest at 275 K.

The plots of the results from varying the design of the InGaP subcell are presented in Figures 58–68.
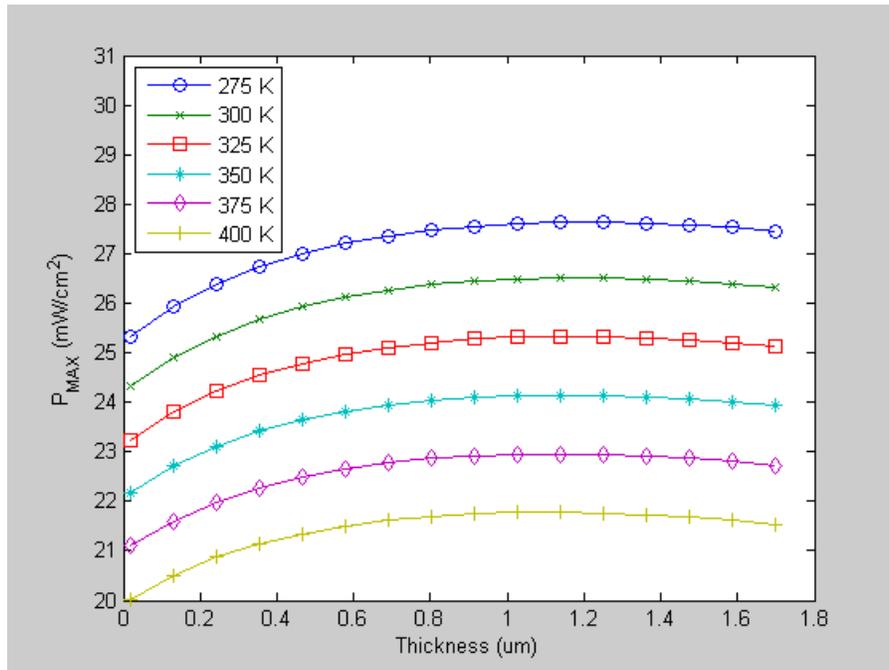
Figure 58.   InGaP – P_{MAX} versus Emitter Thickness (275–400 K).
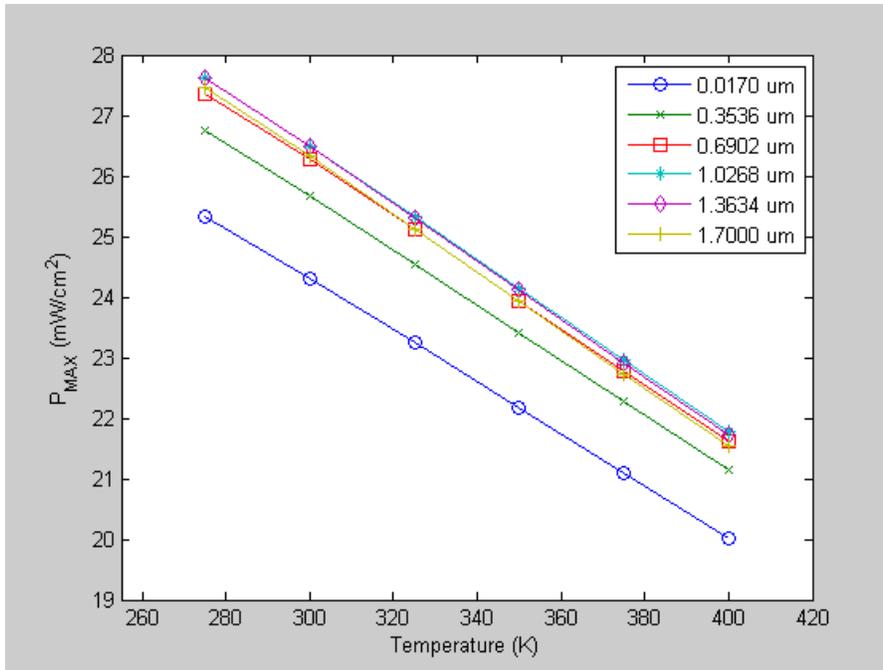
Figure 59.   InGaP Emitter Thickness - P_{MAX} versus Temperature.

The results plotted in Figures 58 and 59 indicate that the emitter thickness of InGaP also has some effect on the performance of this subcell, depending on the temperature. A close examination of the plots reveals that the most efficient cell has a thickness between 1.36 and 1.03 μm at every temperature tested, even though the output continues to decrease significantly as the temperature increases.
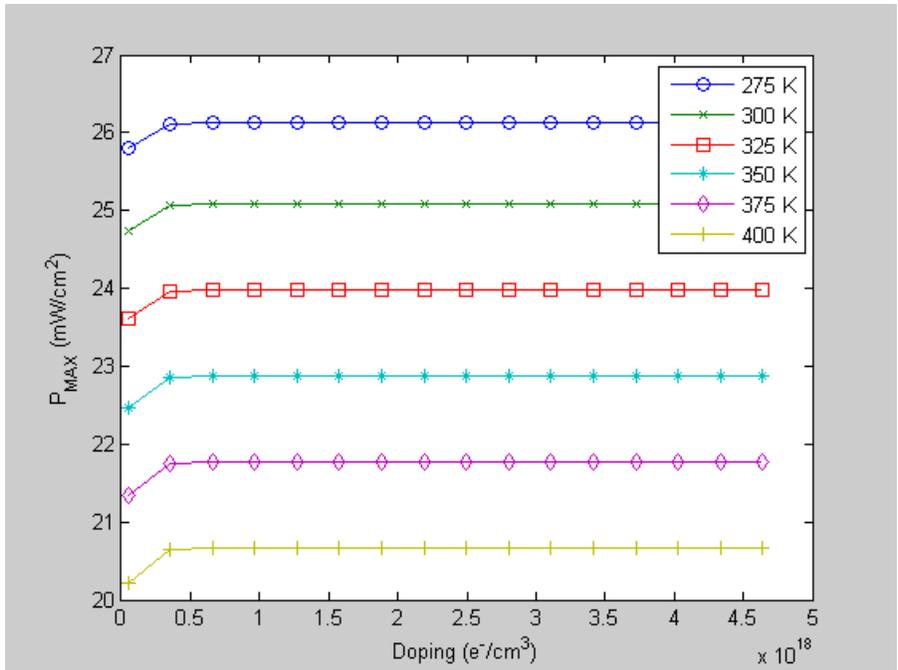
Figure 60.   InGaP - P_{MAX} Versus Emitter Doping (275-400 K).
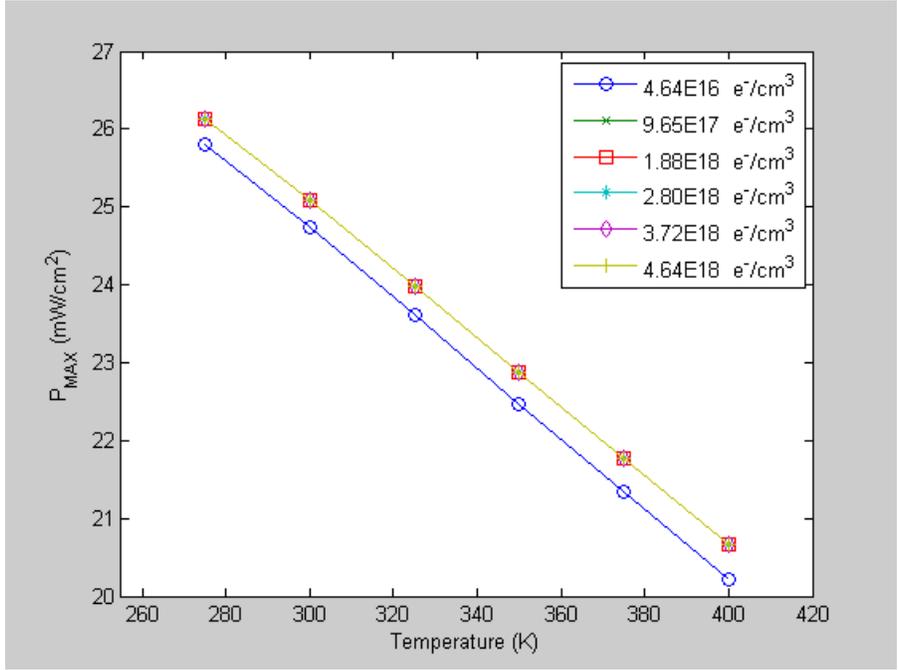
Figure 61.   InGaP Emitter Doping - P_{MAX} Versus Temperature.

The InGaP emitter doping has little effect based on the results plotted in Figures 60 and 61, although the lowest doping tested gives a lower output than the other doping levels tested. Again, as expected, the temperature causes the performance to drop significantly.
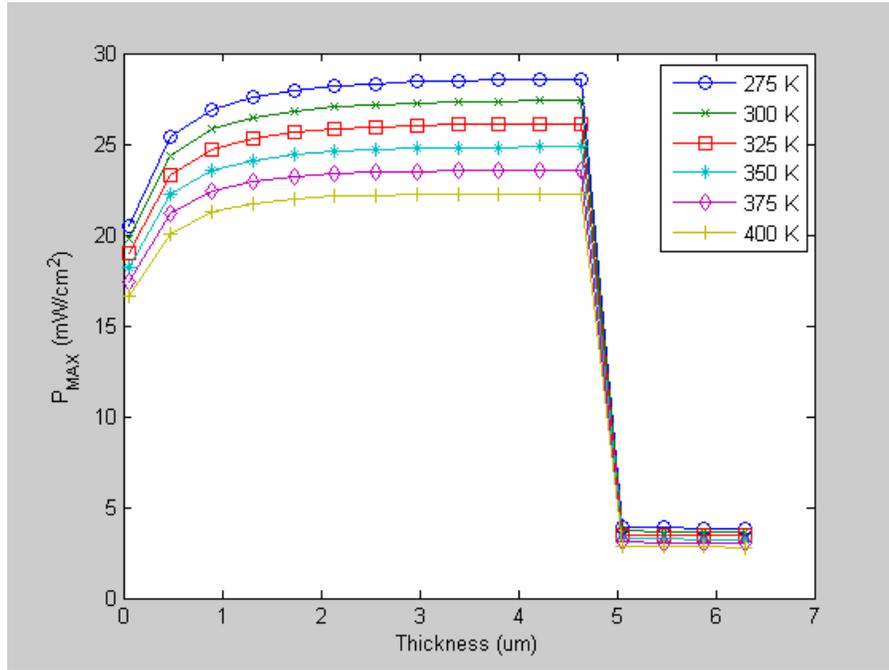
Figure 62.   InGaP - P_{MAX} versus Base Thickness (275-400 K).
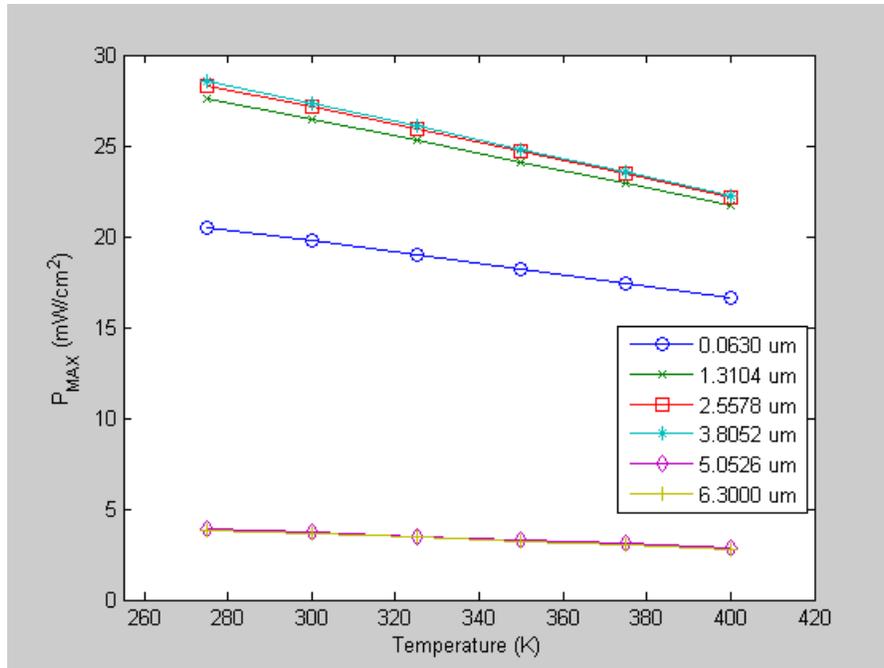
Figure 63. InGaP Base Thickness - P_{MAX} versus Temperature.

Similar to the emitter thickness, the InGaP base thickness has some effect based on the results plotted in Figures 62 and 63, although the greatest thicknesses tested give a much lower output than the other thicknesses tested. The GaAs subcell also gives a sudden drop in output, as demonstrated in Figures 54 and 55, when its base thickness reaches a specific amount. Maybe a certain material property of these two cells or something specific about their design is the cause of this behavior. As far as temperature is concerned, although it causes a decrease in output, the output does not drop for the InGaP base thickness as much as it does for many of the other design parameters.
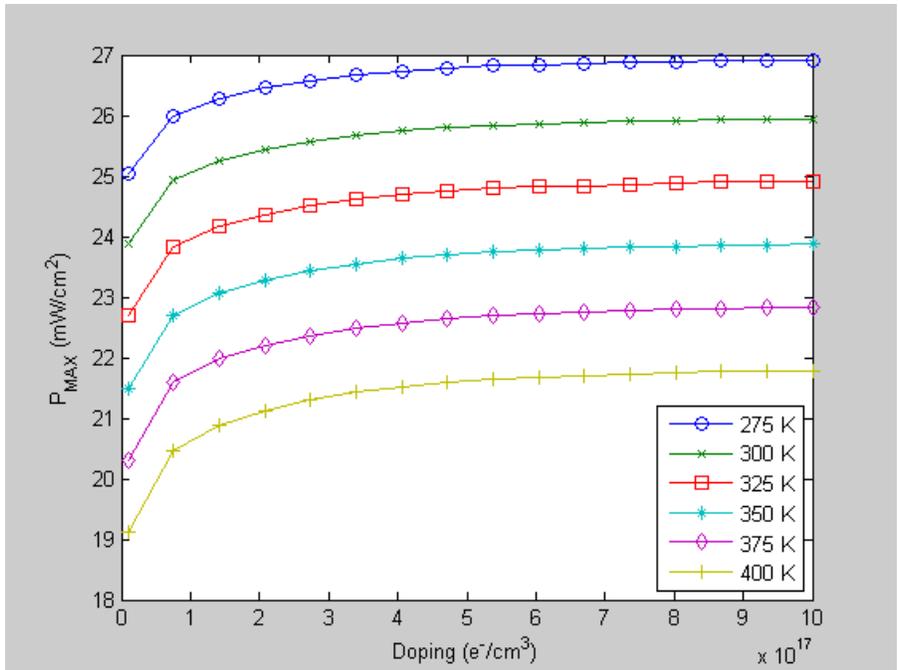
Figure 64.   InGaP – P_{MAX} versus Base Doping (275–400 K).
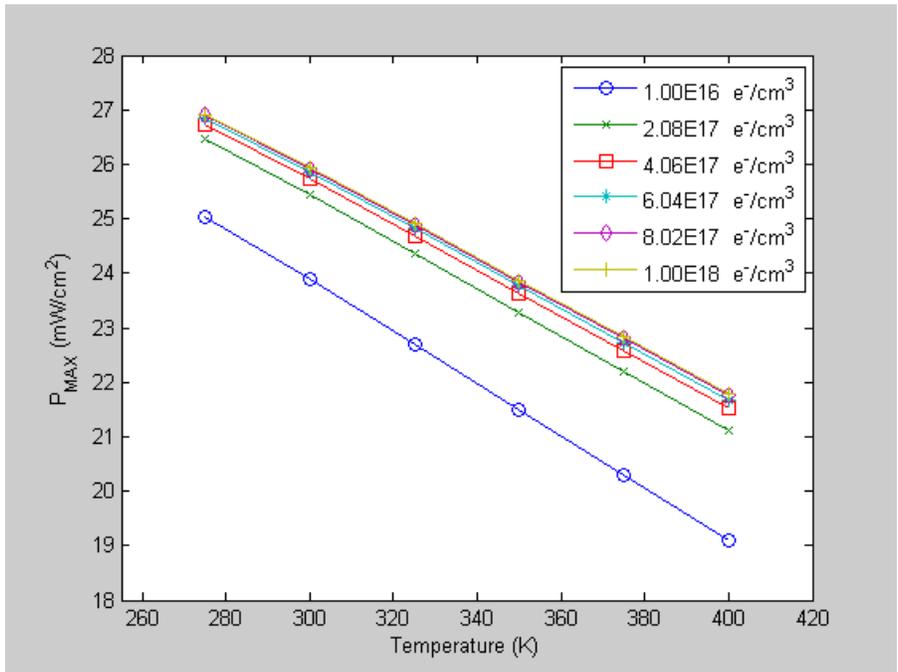
Figure 65.   InGaP Base Doping – P_{MAX} versus Temperature.

Based on the two plots presented in Figures 64 and 65, the InGaP base doping has some effect on the cell's performance. The output of the cell increases as the base doping level increases, although this improvement is much less as the doping level increases over the range tested. The temperature continues to cause a significant decrease in performance.

The author also iterated the mole fraction of InGaP from 0.48 to 0.70, which changed the band gap energy of the material, for each temperature setting to see the effect this had on the performance of the solar cell when exposed to light.
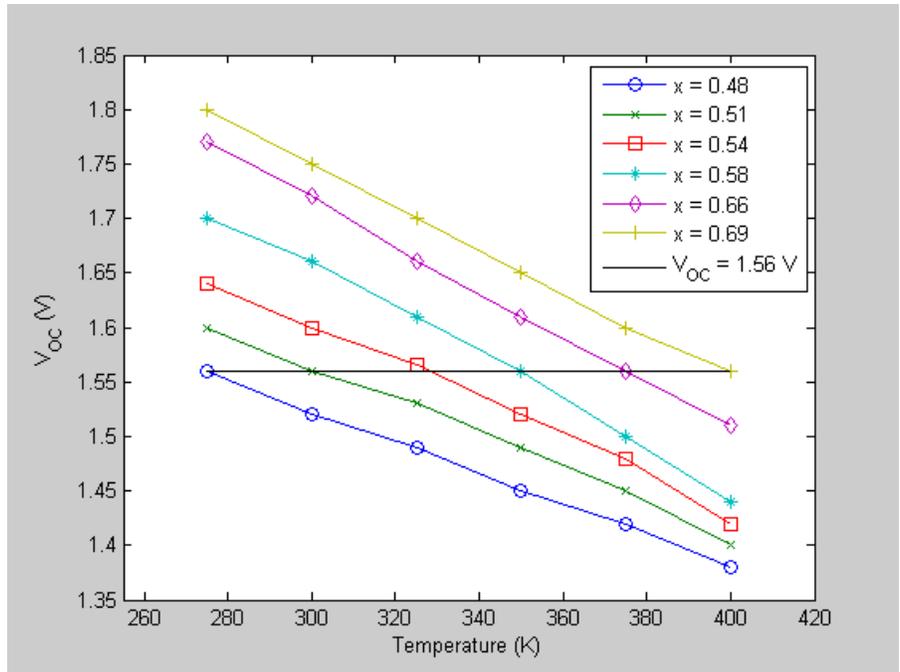
Figure 66. $In_{1-x}Ga_xP$ - V_{OC} versus Temperature.

Plots of V_{OC} versus temperature for these mole fractions are shown in Figure 66 because it is at these mole fractions that V_{OC} is approximately equal to V_{OC} at the original design of the triple-junction cell, which was

optimized at 300 K. This is illustrated by the sold line of V_{OC} = 1.56 V. As shown in the graph above, V_{OC} is 1.56 V at each temperature tested for the mole fractions plotted. This is significant because the decrease in voltage as the temperature increases has the most influence on the decrease in the efficiency of the solar cell. If V_{OC} can be matched at higher temperatures to the V_{OC} at lower temperatures, it may be possible to produce a solar cell that performs at higher temperatures as those optimally design at lower temperatures. However, the voltage is not the only consideration in this; for instance, the design would also need to improve the FF since it tends to decrease as the temperature increases.
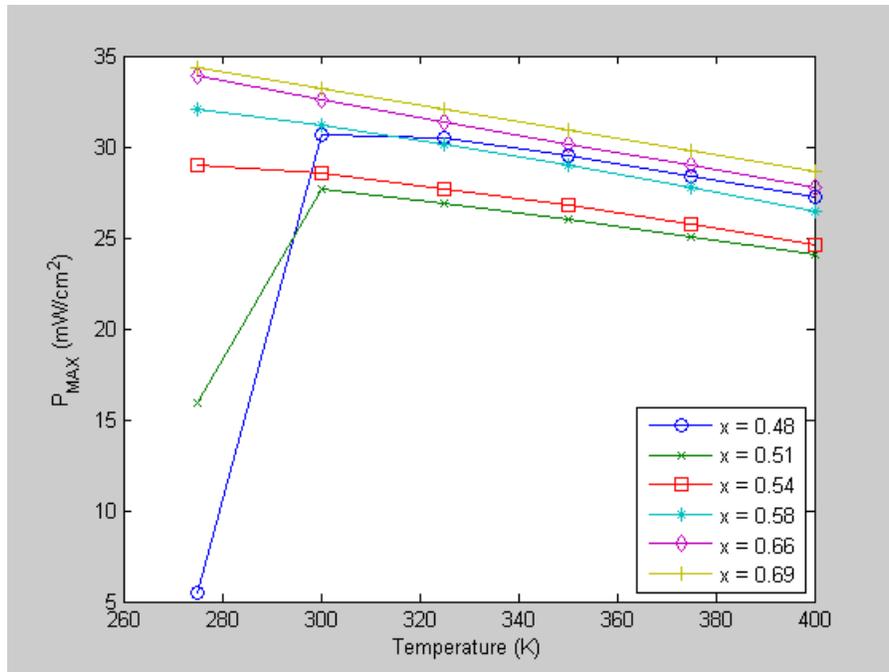
Figure 67. $In_{1-x}Ga_xP$ – P_{MAX} versus Temperature.

Generally, the higher the mole fraction is, the better performance is, as illustrated in Figure 67. However, the results for one mole fraction shown do not follow this

trend. This is for x = 0.48 where it gives better results than x = 0.51, 0.54, and 0.58 at the higher temperatures tested. Unfortunately, the same file used to represent the optical properties of InGaP for the different mole fractions could not be used as the mole fraction was varied. This may have contributed to the difference in the performance trends. Based on the mole fractions tested, the author used three different optical property files in the simulations: one for $0.48 \leq x \leq 0.50$, a second for $0.51 \leq x \leq 0.55$, and the third for $0.56 \leq x \leq 0.70$. These x values were determined to be the best fit for the optical properties files available. From the results, although one may conclude that a higher mole fraction produces better results for InGaP, the problem with choosing a high mole fraction for the InGaP subcell is being able to manufacture such a cell on top of the other subcells.

B.   APPLYING TEMPERATURE EFFECT ON DESIGN OF SUBCELLS

Based on the results of the design alterations simulated for the triple-junction subcells, the author re-examined the design of the original InGaP/GaAs/Ge triple-junction cell.

1.   Increasing InGaP Mole Fraction

Since improvements in the design of the $In_{1-x}Ga_xP$ subcell are achieved by increasing the mole fraction, x, similar iterations for this design parameter are made for the $In_{1-x}Ga_xP$/GaAs/Ge triple-junction cell. A plot of the results for this iterated simulation are presented in Figure 68.
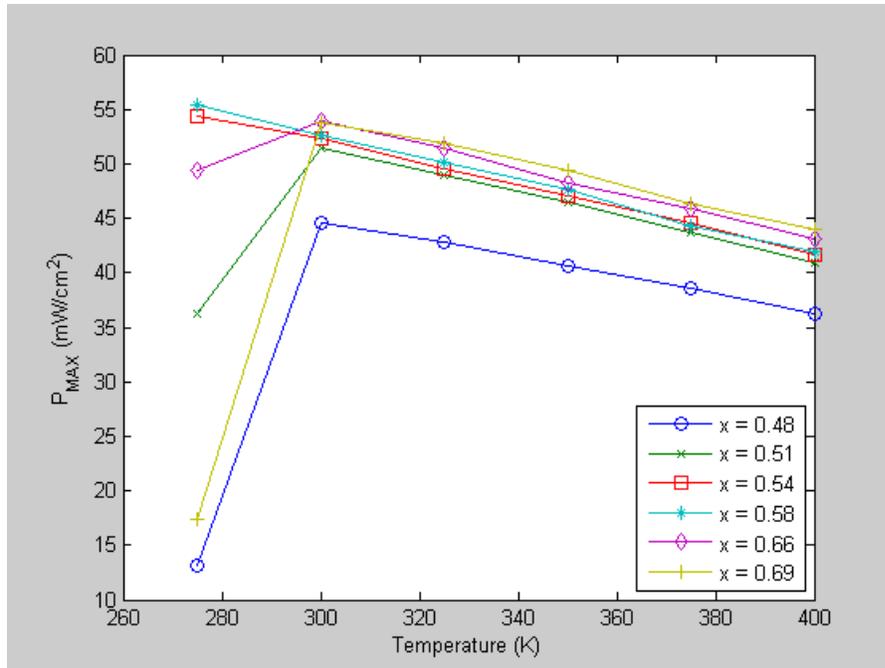
Figure 68. $In_{1-x}Ga_xP/GaAs/Ge-P_{MAX}$ versus Temperature.

As with the $In_{1-x}Ga_xP$ subcell, the triple-junction cell achieves its best performance at the highest mole fraction tested. Even at 300 K, it is better to use a higher mole fraction than the x = 0.51 that the original design uses. A comparison illustrating the improvement made by increasing the mole fraction is given in Table 11.

Table 11.   Mole Fraction Improvement Comparison.

Temperature (K)	Efficiency (%)		Percent Improvement
	x = 0.51	x = 0.69	
275	26.83	12.83	-52.2
300	38.00	39.72	4.53
325	36.14	38.37	6.17
350	34.32	36.52	6.41
375	32.33	34.21	5.82
400	30.28	32.53	7.43

It is clear from the results given in Table 11 that increasing the mole fraction increases the efficiency of the cell at the higher temperatures tested, although this is not true at the lowest temperature tested.

2.   Increasing Doping

Although increasing the thickness of the emitter and base of the top two subcells gives better performance for these subcells, any significant increase in these thicknesses causes the current to be limited in lower subcells. Since the subcells of a multijunction solar cell are connected in series, the lowest current is the limiting current of the overall cell. This means that for subcells that are current matched, as they are for this triple-junction cell, limiting the current any further in any subcell will reduce the output of the multijunction cell. Therefore, increasing the thicknesses is not a viable option.

However, increasing the doping levels of some of the layers of each subcell is beneficial. In particular, the output increases as the GaAs emitter doping and base doping of each subcell is increased. The efficiency of the triple-junction cell at the various temperatures tested is shown in Table 12 with the percent improvement over the original design.

103

Table 12. Doping Improvement Comparison.

Temperature (K)	Efficiency (%)		Percent Improvement
	Original Doping	*Increased Doping*	
275	26.83	28.33	5.59
300	38.00	39.71	4.50
325	36.14	38.02	5.20
350	34.32	36.27	5.68
375	32.33	34.56	6.90
400	30.28	32.67	7.89

Increasing the doping of the bases and the GaAs emitter improves the performance of the cell at every temperature tested.

3.   Increasing Doping and InGaP Mole Fraction

When the doping of each base and GaAs emitter and the mole fraction of InGaP are increased, additional improvements are achieved. The Silvaco code for simulating the performance of the cell with these design modifications is provided in Appendix A. The results of the simulations at 275 to 400 K are given in Table 13.

Table 13. Doping and Mole Fraction Improvement Comparison.

Temperature (K)	Efficiency (%)		Percent Improvement
	Original Doping and x = 0.51	*Increased Doping and x = 0.69*	
275	26.83	9.71	-63.81
300	38.00	40.42	6.37
325	36.14	39.60	9.57
350	34.32	37.77	10.05
375	32.33	35.73	10.52
400	30.28	33.89	11.92

The highest percent improvements are realized when both the doping and mole fraction are increased, except at 275 K. At this low temperature setting, the output is less when the doping and mole fraction are increased than it is when just the mole fraction is increased. However, since an operating temperature between 300 and 400 K is more likely, the results of these simulations support increasing both the mole fraction and the doping levels. The layers of the resulting triple-junction cell design with specific thicknesses and doping levels is shown in Figure 69.

Top InGaP Layer	Window	n+ AlInP	0.01 µm	5e19 cm^{-3}
	Emitter	n+ InGaP	0.17 µm	4.64e17 cm-3
	Base	p+ InGaP	0.63 µm	1e18 cm-3
	BSF	p+ InGaP	0.010 µm	5e19 cm-3
	Buffer	p+ AlInP	0.300 µm	1e18 cm-3
Tunnel Junction	Tunnel Emitter	p+ InGaP	0.015 µm	8e18 cm-3
	Tunnel Base	p+ InGaP	0.015 µm	1e19 cm-3
Middle GaAs Layer	Window	n+ AlInP	0.010 µm	4.64e17 cm-3
	Emitter	n+ GaAs	0.010 µm	4.64e16 cm-3
	Base	p+ GaAs	3.870 µm	1e18 cm-3
	BSF	p+ InGaP	0.010 µm	5e19 cm-3
	Buffer	p+ GaAs	0.300 µm	7e18 cm-3
Tunnel Junction	Tunnel Emitter	p+ GaAs	0.015 µm	8e18 cm-3
	Tunnel Base	n+ GaAs	0.015 µm	1e19 cm-3
Bottom Ge Layer	Window	n+ GaAs	0.050 µm	7e18 cm-3
	Emitter	n+ Ge	0.100 µm	3e18 cm-3
	Substrate	p+ Ge	300 µm	1.614e19 cm-3

Figure 69. Triple-junction Cell Redesign.

THIS PAGE INTENTIONALLY LEFT BLANK

VII. CONCLUSIONS AND RECOMMENDATIONS

A.   CONCLUSIONS

This thesis analyzes the effect of temperature on the performance of a typical multijunction solar cell. An InGaP/GaAs/Ge triple-junction cell serves as the model of this analysis. The author uses Silvaco's virtual wafer fabrication software called ATLAS to model this cell and simulate its behavior. The cell's performance is evaluated at 275, 300, 325, 350, 375, and 400 K. The emitter and base thickness and doping of each subcell and the mole fraction of InGaP is varied at these temperature settings to observe their effect on performance.

The simulations provide the results necessary to make some general observations about the effect of operating temperature and the variation of cell design on the performance of the multijunction cell. First, the obvious conclusion of these simulations is that as temperature increases, performance decreases no matter how the author varies the cell's design parameters. This is why it is important to have the expected operating temperature of a solar cell to guide the design process. Furthermore, even though efficiency drops as the temperature rises, improvements in efficiency at every temperature setting between 300 and 400 K occur when the GaAs emitter doping level; InGaP, GaAs, and Ge base doping levels; and InGaP mole fraction are increased. The greatest improvement in efficiency—11.92% at 400 K—is obtained when all of these doping levels and the mole fraction are increased.

B. RECOMMENDATIONS

Future work in the area of this research can be conducted in a number of ways. First, consider the optical characteristics of typical solar cell meaterials, which change with temperature. It would be beneficial to investigate whether those used in the models of this thesis are based on the most accurate data available. Are the methods and models used in making the calculations based on temperature the most ideal means to determine the effect of temperature on the solar cells? It is possible that improvements can be made based on investigating these factors affecting the simulations.

Another area that can be investigated further concerns the results presented in Figures 54, 55, 62, and 63 in Chapter VI. In these figures, the author shows that increasing the base thicknesses of GaAs and InGaP to a specific point improves the performance of these cells but increased thicknesses beyond this point cause a significant drop in output. An understanding of why this drop in output occurs is relevant and may lead to better cell designs. However, an investigation of this may also show that a problem exists in the code used to run the simulations, and this is why unusual results are obtained.

In addition, exploring the possibility of replacing the InGaP, GaAs, and Ge layers of the triple-junction cell analyzed in this thesis with alternate materials may prove beneficial in improving the cell for operation at higher temperatures. For instance, some in the solar cell industry are using InGaAs with a band gap energy of 0.7 and 1.0 eV as a substitute for Ge. Would this triple-junction cell

configuration handle increased temperatures better? Since adding a layer to a cell has the potential to increase output, can a layer be added to the InGaP/GaAs/Ge cell that would perform better at increased temperatures? One common material added between the GaAs and Ge layers is $In_{1-x}Ga_xN_{1-y}As_y$ that has a band gap energy of 1.0 eV. Investigating whether a quad-junction cell like that would operate better than the triple-junction at the higher temperatures could be another area of research for someone to pursue.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A.   ATLAS SOURCE CODE

This appendix provides the ATLAS input decks used in this thesis. Section A contains the original triple-junction solar cell design optimized by Tsutagawa. Section B contains the individual layers of the triple-junction solar cell simulated separately for the temperature analyses.

A.   MICHAEL TSUTAGAWA'S CHAMPION CELL

The original ATLAS input deck used in this thesis came from Michael Tsutagawa's triple-junction solar cell he optimized using Silvaco software. The following is the input deck for his InGaP/GaAs/Ge solar cell design.

```
#Brian Sullivan (2010)
#This solar cell design is Michael Tsutagawa's (16 Nov 2008) CHAMPION CELL,
# which is 36.28% efficient @ 300 K:
# At 300 K, using the AM0nrel.spec file, this 0.82InGaP (doped 5e19)/3.9umGaAs
# (doped 5e19)/ORIG300um Ge (Emitter & Substrate doped 3e18)cell has
# Voc=2.76425 V, Jsc=19.8531 mA/cm^2, P=0.05487893 mW/cm^2, Pmax=0.0490907
# mW/cm^2, Vmax=2.525 V, Jmax=19.4419 mA/cm^2, FF=0.894527, and Eff=0.362829

# Modifications from Tsutagawa's input deck have been made in order to change
# the design quickly using the set command and to run the simulations using
# Matlab. However, the integrity of the original design is maintained and the
# same results that Tsutagawa obtained are also obtained using this file.

# Main Silvaco ATLAS Structure
go atlas
# Definition of Constants
# Mesh
#    X-Mesh
#    Y-Mesh
# Regions
# Electrodes
# Doping
# Material Properties
# Models
# Light beams
# Solving

##################
# Constants
##################
#Filename of the IV curve output
set IVName=ChampionIVCurve.log
#Filename of the extracted P-V curve
set ExtractPowerFile=ChampionPower.log
```

```
#Fileanme of the extracted output
set ExtractDataFile=ChampionData.txt
#Filename of the structure
set StructFile=ChampionProfile.str

#Temperature
set tempature=300

#Baseline number of divisions/layer (10 is about the lowest that can be used)
set divs=10
#Width of cell (microns)
set cellWidth=500

#Cell width
set width3d=100e6/$cellWidth
set cellWidthHalf=$cellWidth/2
set cellWidthDiv1=$cellWidth/$divs
set cellWidthDiv2=$cellWidth/$divs/5

#Vacuum thickness (microns)
set vacuumThick=0.05

##First layer dimensions
#Window thickness (microns)
set windowThick1=0.01
#Window doping (atoms/cc)
set winDop1=5e+019
#Emitter thickness (microns)
set emitterThick1=0.17
#Emitter doping (atoms/cc)
set emitDop1=4.64e+017
#Base thickness (microns)
set baseThick1=0.63
#Base doping (atoms/cc)
set baseDop1=1e+017
#BSF thickness (microns)
set bsfThick1=0.01
#BSF doping (microns)
set bsfDop1=5e+019
#Buffer thickness (microns)
set bufferThick1=0.03
#Buffer doping (microns)
set bufferDop1=1e+018

##First tunnel junction
#Emitter thickness (microns)
set tunnelEmitThick1=0.015
#Emitter doping (atoms/cc)
set tunnelEmitDop1=8e18
#Base thickness (microns)
set tunnelBaseThick1=0.015
#Base doping (atoms/cc)
set tunnelBaseDop1=1e19

##Second layer
#Window thickness (microns)
set windowThick2=0.01
#Window doping (atoms/cc)
set winDop2=4.64e+017
#Emitter thickness (microns)
set emitterThick2=0.01
#Emitter doping (atoms/cc)
```

```
set emitDop2=4.64e+015
#Base thickness (microns)
set baseThick2=3.87
#Base doping (atoms/cc)
set baseDop2=1e+017
#BSF thickness (microns)
set bsfThick2=0.01
#BSF doping (microns)
set bsfDop2=5e19
#Buffer thickness (microns)
set bufferThick2=0.3
#Buffer doping (microns)
set bufferDop2=7e+018

##Second tunnel junction
#Emitter thickness (microns)
set tunnelEmitThick2=0.015
#Emitter doping (atoms/cc)
set tunnelEmitDop2=8e18
#Base thickness (microns)
set tunnelBaseThick2=0.015
#Base doping (atoms/cc)
set tunnelBaseDop2=1e19

##Third layer
#Window thickness (microns)
set windowThick3=0.05
#Window doping (atoms/cc)
set winDop3=7e+018
#Emitter thickness (microns)
set emitterThick3=0.1
#Emitter doping (atoms/cc)
set emitDop3=3e+018
#Base thickness (microns)
set baseThick3=300
#Base doping (atoms/cc)
set baseDop3=3e+018

#Dimensions based on other dimensions
set baseLo3=304.39
set baseHi3=$baseLo3-$baseThick3
set baseDiv3=$baseThick3/6
set emitterLo3=$baseHi3
set emitterHi3=$emitterLo3-$emitterThick3
set emitterDiv3=$emitterThick3/$divs
set windowLo3=$emitterHi3
set windowHi3=$windowLo3-$windowThick3
set windowDiv3=$windowThick3/$divs

set tunnelBaseLo2=$windowHi3
set tunnelBaseHi2=$tunnelBaseLo2-$tunnelBaseThick2
set tunnelBaseDiv2=$tunnelBaseThick2/$divs
set tunnelEmitterLo2=$tunnelBaseHi2
set tunnelEmitterHi2=$tunnelEmitterLo2-$tunnelEmitThick2
set tunnelEmitterDiv2=$tunnelEmitThick2/$divs

set bufferLo2=$tunnelEmitterHi2
set bufferHi2=$bufferLo2-$bufferThick2
set bufferDiv2=$bufferThick2/$divs/$divs
set bsfLo2=$bufferHi2
set bsfHi2=$bsfLo2-$bsfThick2
set bsfDiv2=$bsfThick2/$divs
```

```
set baseLo2=$bsfHi2
set baseHi2=$baseLo2-$baseThick2
set baseMid2=$baseLo2-$baseThick2/2
set baseMidDiv2=$baseThick2/$divs/$divs
set baseDiv2=$baseThick2/$divs/$divs/$divs
set emitterLo2=$baseHi2
set emitterHi2=$emitterLo2-$emitterThick2
set emitterDiv2=$emitterThick2/$divs
set windowLo2=$emitterHi2
set windowHi2=$windowLo2-$windowThick2
set windowDiv2=$windowThick2/$divs


set tunnelBaseLo1=$windowHi2
set tunnelBaseHi1=$tunnelBaseLo1-$tunnelBaseThick1
set tunnelBaseDiv1=$tunnelBaseThick1/$divs
set tunnelEmitterLo1=$tunnelBaseHi1
set tunnelEmitterHi1=$tunnelEmitterLo1-$tunnelEmitThick1
set tunnelEmitterDiv1=$tunnelEmitThick1/$divs


set bufferLo1=$tunnelEmitterHi1
set bufferHi1=$bufferLo1-$bufferThick1
set bufferDiv1=$bufferThick1/$divs
set bsfLo1=$bufferHi1
set bsfHi1=$bsfLo1-$bsfThick1
set bsfDiv1=$bsfThick1/$divs*2
set baseLo1=$bsfHi1
set baseHi1=$baseLo1-$baseThick1
set baseMid1=$baseLo1-$baseThick1/2
set baseMidDiv1=$baseThick1/$divs
set baseDiv1=$baseThick1/$divs/$divs
set emitterLo1=$baseHi1
set emitterHi1=$emitterLo1-$emitterThick1
set emitterDiv1=$emitterThick1/$divs
set windowLo1=$emitterHi1
set windowHi1=$windowLo1-$windowThick1
set windowDiv1=$windowThick1/$divs*2


set vacuumLo=$windowHi1
set vacuumHi=$vacuumLo-$vacuumThick
set vacuumDiv=$vacuumThick/$divs

##################
# Mesh
##################
#Define X-Mesh.-Must start from left to right or from lesser to greater values.
      #mesh width=200000
mesh width=$width3d
      #mesh space.mult=1
# X-mesh:  surface=500 um^2= 1/200000 cm^2
x.mesh loc=-$cellWidthHalf spac=$cellWidthDiv1
x.mesh loc=0              spac=$cellWidthDiv2
x.mesh loc=$cellWidthHalf  spac=$cellWidthDiv1

#Define the Y-Mesh.-Must start from surface of the device to the back, or go
#                   from lesser to greater value with up as negative.
# Vacuum
y.mesh loc=$vacuumLo   spac=$vacuumDiv
# Window (0.01 um)
y.mesh loc=$windowLo1  spac=$windowDiv1
# Emitter (0.17 um)
y.mesh loc=$emitterLo1  spac=$emitterDiv1
# Base (0.63 um)
```

114

```
y.mesh loc=$baseMid1  spac=$baseMidDiv1
y.mesh loc=$baseLo1  spac=$baseDiv1
# BSF (0.01 um)
y.mesh loc=$bsfLo1  spac=$bsfDiv1
# Buffer (0.03 um)
y.mesh loc=$bufferLo1  spac=$bufferDiv1

# Tunnel Emitter (0.015)
y.mesh loc=$tunnelEmitterLo1 spac=$tunnelEmitterDiv1
# Tunnel Base (0.015)
y.mesh loc=$tunnelBaseLo1   spac=$tunnelBaseDiv1

# Window (0.01 um)
y.mesh loc=$windowLo2   spac=$windowDiv2
# Emitter (0.01 um)
y.mesh loc=$emitterLo2   spac=$windowDiv2
# Base (3.87 um)
y.mesh loc=$baseMid2  spac=$baseMidDiv2
y.mesh loc=$baseLo2   spac=$baseDiv2
# BSF (0.01 um)
y.mesh loc=$bsfLo2   spac=$bsfDiv2
# Buffer (0.3 um)
y.mesh loc=$bufferLo2   spac=$bufferDiv2

# Tunnel Emitter (0.015)
y.mesh loc=$tunnelEmitterLo2  spac=$tunnelEmitterDiv2
# Tunnel Base (0.015)
y.mesh loc=$tunnelBaseLo2   spac=$tunnelBaseDiv2

# Window (0.05 um) - ORIGINAL Cell parameters
y.mesh loc=$windowLo3   spac=$windowDiv3
# Emitter (0.1 um)
y.mesh loc=$emitterLo3   spac=$emitterDiv3
# Substrate (300 um)
y.mesh loc=$baseLo3 spac=$baseDiv3

##################
# REGIONS
##################
# Vacuum
region num=1 material=Vacuum x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$vacuumHi y.max=$vacuumLo
#Window AlInP (=InAsP)
region num=2 material=InAsP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$windowHi1 y.max=$windowLo1
#Emitter
region num=3 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$emitterHi1 y.max=$emitterLo1
#Base
region num=4 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$baseHi1 y.max=$baseLo1
#BSF
region num=5 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$bsfHi1 y.max=$bsfLo1
#Buffer AlInP (=InAsP)
region num=6 material=InAsP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$bufferHi1 y.max=$bufferLo1

#Tunnel Emitter
region num=7 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$tunnelEmitterHi1 y.max=$tunnelEmitterLo1
#Tunnel Base
```

```
region num=8 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelBaseHi1 y.max=$tunnelBaseLo1


#Window AlInP (=InAsP)
region num=9 material=InAsP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$windowHi2 y.max=$windowLo2
#Emitter
region num=10 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$emitterHi2 y.max=$emitterLo2
#Base
region num=11 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$baseHi2 y.max=$baseLo2
#BSF
region num=12 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$bsfHi2 y.max=$bsfLo2
#Buffer
region num=13 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$bufferHi2 y.max=$bufferLo2


#Tunnel Emitter
region num=14 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelEmitterHi2 y.max=$tunnelEmitterLo2
#Tunnel Base
region num=15 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelBaseHi2 y.max=$tunnelBaseLo2


#Window
region num=16 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$windowHi3 y.max=$windowLo3
#Emitter
region num=17 material=Ge x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$emitterHi3 y.max=$emitterLo3
#Substrate
region num=18 material=Ge x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$baseHi3 y.max=$baseLo3


######################
# ELECTRODES - Use 2% of cell width for optimal grid contact width
######################     (500 x 0.02 = 10 um)
electrode name=cathode x.min=-5 x.max=5 y.min=$vacuumLo y.max=$vacuumLo
electrode name=tunneltop x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelEmitterHi1 y.max=$tunnelBaseLo1
electrode name=tunnelbot x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelEmitterHi2 y.max=$tunnelBaseLo2
electrode name=anode x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$baseLo3 y.max=$baseLo3


###################
# DOPING - n-emitter on p-base
###################
# Window - Realistic from 1e20
doping uniform region=2 n.type conc=$winDop1
# Emitter
doping uniform region=3 n.type conc=$emitDop1
# Base
doping uniform region=4 p.type conc=$baseDop1
# BSF - Realistic from 1e20
doping uniform region=5 p.type conc=$bsfDop1
# Buffer
doping uniform region=6 p.type conc=$bufferDop1


# Tunnel Emitter
```

```
doping uniform region=7 p.type conc=$tunnelEmitDop1
# Tunnel Base
doping uniform region=8 n.type conc=$tunnelBaseDop1

# Window
doping uniform region=9 n.type conc=$winDop2
# Emitter
doping uniform region=10 n.type conc=$emitDop2
# Base
doping uniform region=11 p.type conc=$baseDop2
# BSF - Realistic from 1e20
doping uniform region=12 p.type conc=$bsfDop2
# Buffer - May want to change to 1e18.
doping uniform region=13 p.type conc=$bufferDop2

# Tunnel Emitter
doping uniform region=14 p.type conc=$tunnelEmitDop2
# Tunnel Base
doping uniform region=15 n.type conc=$tunnelBaseDop2

# Window
doping uniform region=16 n.type conc=$winDop3
# Emitter - Changed from 2e18
doping uniform region=17 n.type conc=$emitDop3
# Substrate - Changed from 1e17
doping uniform region=18 p.type conc=$baseDop3

#####################
# MATERIAL PROPERTIES
#####################
material TAUN=1e-7 TAUP=1e-7 COPT=1.5e-10 AUGN=8.3e-32 AUGP=1.8e-31

# Vacuum
material material=Vacuum real.index=3.3 imag.index=0

# InGaP
material material=InGaP EG300=1.9 PERMITTIVITY=11.62 AFFINITY=4.16
material material=InGaP MUN=1945 MUP=141
material material=InGaP NC300=1.3e20 NV300=1.28e19
material material=InGaP index.file=InGaP-1.9.opt

# GaAs
material material=GaAs EG300=1.42 PERMITTIVITY=13.1 AFFINITY=4.07
material material=GaAs MUN=8800 MUP=400
material material=GaAs NC300=4.7e17 NV300=7e18
material material=GaAs index.file=GaAs.opt

# Ge
material material=Ge EG300=0.67 PERMITTIVITY=16 AFFINITY=4
material material=Ge MUN=3900 MUP=1800
material material=Ge NC300=1.04e19 NV300=6e18
material material=Ge index.file=Ge.opt

# AlInP (=InAsP)
material material=InAsP EG300=2.4 PERMITTIVITY=11.7 AFFINITY=4.2
material material=InAsP MUN=2291 MUP=142
material material=InAsP NC300=1.08e20 NV300=1.28e19
material material=InAsP index.file=AlInP.opt

#Gold
material material=Gold real.index=1.2 imag.index=1.8
```

```
####################
# MODELS
####################
models srh temp=$tempature print

####################
# CONTACTS
####################
contact name=tunneltop resist=1e17
contact name=tunnelbot resist=1e17

####################
# LIGHTBEAMS - Calculates the position of the light source and defines the
beam.
####################
beam num=1 x.origin=0 y.origin=-5 angle=90 back.refl \
        power.file=AM0nrel.spec wavel.start=0.12 wavel.end=2.4 wavel.num=1500

#Output and Display the Structure file
struct outfile=$StructFile
#tonyplot $StructFile

####################
# SOLVING I-V CURVE
####################
method gummel newton maxtraps=10 itlimit=25 climit=1 dvmax=0.1

solve initial
#Beam strength of 1.0 sun
solve b1=1.0

log outfile=$IVName

solve vanode=0.0 name=anode vstep=0.05 vfinal=2.4
solve vanode=2.4 name=anode vstep=0.005 vfinal=2.8

extract init infile="$'IVName'"
extract name="Jsc" abs(y.val from curve(v."anode," i."cathode") where \
        x.val=0.0) datafile="$'ExtractDataFile'"
extract name="Voc" x.val from curve(v."anode," i."cathode") where \
        y.val=0.0 datafile="$'ExtractDataFile'"
extract name="iv"  curve(v."anode," i."cathode") outfile="IVCurve.dat"

extract name="Power" curve(v."anode," (v."anode" * i."cathode")) \
        outf="$'ExtractPowerFile'"
extract name="Pmax" max(curve(v."anode," (v."anode" * i."cathode"))) \
        datafile="$'ExtractDataFile'"
extract name="Vmax" x.val from curve(v."anode," (v."anode"*i."cathode") ) \
        where y.val=-$"Pmax" datafile="$'ExtractDataFile'"
extract name="Jmax" $"Pmax"/$"Vmax" datafile="$'ExtractDataFile'"
extract name="FF" $"Pmax"/($"Jsc"*$"Voc") datafile="$'ExtractDataFile'"
#Use AMO of 0.1353 W/cm^2
extract name="Eff" $Pmax/0.1353 datafile="$'ExtractDataFile'"

#save outf=$IVName

################
#OUTPUT
################
#tonyplot IVCurve.dat
log outfile=done.log
log off
```

B. SINGLE-JUNCTION SOLAR CELLS

This section of this appendix contains the input decks for the individual layers of the multijunction solar cell analyzed in this thesis. These input decks allowed multiple simulations to be run while altering the design of the cells and changing the operating temperature. These input decks are written such that the design parameters can be changed quickly and the simulations can be conducted using Matlab.

1. InGaP

```
#Brian Sullivan (2010)
#This InGaP solar cell is based on the top layer of Michael Tsutagawa's (16 Nov
# 2008) CHAMPION CELL, which is 36.28% efficient @ 300 K:
# At 300 K, using the AM0nrel.spec file, his 0.82 InGaP (doped 5e19) / 3.9um
# GaAs (doped 5e19) / ORIG300um Ge (Emitter & Substrate doped 3e18) cell has
# Voc = 2.76425 V, Jsc = 19.8531 mA/cm^2, Pmax = 0.0490907 mW/cm^2, Vmax =
# 2.525 V, Jmax = 19.4419 mA/cm^2, FF = 0.894527, and Eff = 0.362829.

# Modifications from Tsutagawa's input deck have been made in order to change
# the design quickly using the set command and to run the simulations using
# Matlab. However, the integrity of the original design is maintained.


# Main Silvaco ATLAS Structure
go atlas
# Definition of Constants
# Mesh
#   X-Mesh
#   Y-Mesh
# Regions
# Electrodes
# Doping
# Material Properties
# Models
# Light beams
# Solving

##################
# Constants
##################
#Filename of the IV curve output
set IVName=InGaP_x_IVCurve.log
#Filename of the extracted P-V curve
set ExtractPowerFile=InGaP_x_Power.log
#Fileanme of the extracted output
set ExtractDataFile=InGaP_x_Data.txt
#Filename of the structure
set StructFile=InGaP_x_Profile.str

#Molar concentration of In(1-x)Ga(x)P
```

```
set x=0.49

#Temperature
set temperature=275

#Baseline number of divisions/layer (10 is about the lowest that can be used)
set divs=10
#Width of cell (microns)
set cellWidth=500

#Cell width
set width3d=100e6/$cellWidth
set cellWidthHalf=$cellWidth/2
set cellWidthDiv1=$cellWidth/$divs
set cellWidthDiv2=$cellWidth/$divs/5

#Vacuum thickness (microns)
set vacuumThick=0.05

##First layer dimensions
#Window thickness (microns)
set windowThick1=0.01
#Window doping (atoms/cc)
set winDop1=5e+019
#Emitter thickness (microns)
set emitterThick1=0.17
#Emitter doping (atoms/cc)
set emitDop1=4.64e+017
#Base thickness (microns)
set baseThick1=0.63
#Base doping (atoms/cc)
set baseDop1=1e+017
#BSF thickness (microns)
set bsfThick1=0.01
#BSF doping (microns)
set bsfDop1=5e+019
#Buffer thickness (microns)
set bufferThick1=0.03
#Buffer doping (microns)
set bufferDop1=1e+018

#Dimensions based on other dimensions
set bufferLo1=-0.02
set bufferHi1=$bufferLo1-$bufferThick1
set bufferDiv1=$bufferThick1/$divs
set bsfLo1=$bufferHi1
set bsfHi1=$bsfLo1-$bsfThick1
set bsfDiv1=$bsfThick1/$divs
set baseLo1=$bsfHi1
set baseHi1=$baseLo1-$baseThick1
set baseMid1=$baseLo1-$baseThick1/2
set bseMidDiv1=$baseThick1/$divs
set baseDiv1=$baseThick1/$divs/$divs
set emitterLo1=$baseHi1
set emitterHi1=$emitterLo1-$emitterThick1
set emitterDiv1=$emitterThick1/$divs/$divs
set windowLo1=$emitterHi1
set windowHi1=$windowLo1-$windowThick1
set windowDiv1=$windowThick1/$divs

set vacuumLo=$windowHi1
set vacuumHi=$vacuumLo-$vacuumThick
```

120

```
set vacuumDiv=$vacuumThick/$divs

set lightY=$windowHi1-5

###################
# Mesh
###################
#Define the X-Mesh.-Must start from left to right or lesser to greater values.
       #mesh width=200000
mesh width=$width3d
       #mesh space.mult=1
# X-mesh:  surface=500 um^2= 1/200000 cm^2
x.mesh loc=-$cellWidthHalf spac=$cellWidthDiv1
x.mesh loc=0              spac=$cellWidthDiv2
x.mesh loc=$cellWidthHalf  spac=$cellWidthDiv1

#Define the Y-Mesh.-Must start from surface of the device to the back, or go
#                  from lesser to greater value with up as negative.
# Vacuum
y.mesh loc=$vacuumLo  spac=$vacuumDiv
# Window (0.01 um)
y.mesh loc=$windowLo1  spac=$windowDiv1
# Emitter (0.17 um)
y.mesh loc=$emitterLo1  spac=$emitterDiv1
# Base (0.63 um)
y.mesh loc=$baseMid1  spac=$baseMidDiv1
y.mesh loc=$baseLo1  spac=$baseDiv1
# BSF (0.01 um)
y.mesh loc=$bsfLo1  spac=$bsfDiv1
# Buffer (0.03 um)
y.mesh loc=$bufferLo1  spac=$bufferDiv1

###################
# REGIONS
###################
# Vacuum
region num=1 material=Vacuum x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$vacuumHi y.max=$vacuumLo
#Window AlInP (=InAsP)
region num=2 material=InAsP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$windowHi1 y.max=$windowLo1
#Emitter
region num=3 material=InGaP x.comp=$x x.min=-$cellWidthHalf \
       x.max=$cellWidthHalf y.min=$emitterHi1 y.max=$emitterLo1
#Base
region num=4 material=InGaP x.comp=$x x.min=-$cellWidthHalf \
       x.max=$cellWidthHalf y.min=$baseHi1 y.max=$baseLo1
#BSF
region num=5 material=InGaP x.comp=$x x.min=-$cellWidthHalf \
       x.max=$cellWidthHalf y.min=$bsfHi1 y.max=$bsfLo1
#Buffer AlInP (=InAsP)
region num=6 material=InAsP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$bufferHi1 y.max=$bufferLo1

#####################
# ELECTRODES - Use 2% of cell width for optimal grid contact width
#####################     (500 x 0.02 = 10 um)
electrode name=cathode x.min=-5 x.max=5 y.min=$vacuumLo y.max=$vacuumLo
electrode name=anode x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$bufferLo1 y.max=$bufferLo1
```

```
####################
# DOPING - n-emitter on p-base
####################
# Window - Realistic from 1e20
doping uniform region=2 n.type conc=$winDop1
# Emitter
doping uniform region=3 n.type conc=$emitDop1
# Base
doping uniform region=4 p.type conc=$baseDop1
# BSF - Realistic from 1e20
doping uniform region=5 p.type conc=$bsfDop1
# Buffer
doping uniform region=6 p.type conc=$bufferDop1


####################
# MATERIAL PROPERTIES
####################
material TAUN=1e-7 TAUP=1e-7 COPT=1.5e-10 AUGN=8.3e-32 AUGP=1.8e-31

# Vacuum
material material=Vacuum real.index=3.3 imag.index=0

# In(1-x)Ga(x)P
material material=InGaP EG300=1.35+0.732*$x+0.7*$x^2
material material=InGaP PERMITTIVITY=12.5-1.4*$x AFFINITY=4.38-0.58*$x
material material=InGaP MUN=33000+(8500-33000)*$x+4600+ \
        (300-4600)*$x-(33000+(8500-33000)*$x)
material material=InGaP MUP=460+(400-460)*$x+150+ \
        (100-150)*$x-(460+(400-460)*$x)
#material material=InGaP NC300=1.3e20 NV300=1.28e19
material material=InGaP index.file=InGaP-0.51-0.49.opt

# AlInP (=InAsP)
material material=InAsP EG300=2.4 PERMITTIVITY=11.7 AFFINITY=4.2
material material=InAsP MUN=2291 MUP=142
material material=InAsP NC300=1.08e20 NV300=1.28e19
material material=InAsP index.file=AlInP.opt

#Gold
material material=Gold real.index=1.2 imag.index=1.8


####################
# MODELS
####################
models srh bgn temp=$temperature print

####################
# LIGHTBEAMS -Calculates the position of the light source and defines the beam.
####################
beam num=1 x.origin=0 y.origin=$lightY angle=90 back.refl \
        power.file=AM0nrel.spec wavel.start=0.12 wavel.end=2.4 wavel.num=1500

#Output and Display the Structure file
struct outfile=$StructFile
#tonyplot $StructFile


####################
# SOLVING I-V CURVE
####################
method gummel newton maxtraps=10 itlimit=25 climit=1 dvmax=0.1

solve initial
```

```
#Beam strength of 1.0 sun
solve b1=1.0

log outfile=$IVName

solve vanode=0.0 name=anode vstep=0.05 vfinal=1.2
solve vanode=1.3 name=anode vstep=0.005 vfinal=1.58

extract init infile="$'IVName'"
extract name="Jsc" y.val from curve(v."anode," i."cathode") where x.val=0.0 \
      datafile="$'ExtractDataFile'"
extract name="Voc" x.val from curve(v."anode," i."cathode") where y.val=0.0 \
      datafile="$'ExtractDataFile'"
extract name="iv"  curve(v."anode," i."cathode") outfile="IVCurve.dat"

extract name="Power" curve(v."anode," (v."anode" * i."cathode")) \
      outf="$'ExtractPowerFile'"
extract name="Pmax" max(curve(v."anode," (v."anode" * i."cathode"))) \
      datafile="$'ExtractPowerFile'"
extract name="Vmax" x.val from curve(v."anode," (v."anode"*i."cathode") ) \
      where y.val=$"Pmax" datafile="$'ExtractDataFile'"
extract name="Jmax" $"Pmax"/$"Vmax" datafile="$'ExtractDataFile'"
extract name="FF" $"Pmax"/($"Jsc"*$"Voc") datafile="$'ExtractDataFile'"
#Use AMO of 0.1353 W/cm^2
extract name="Eff" $Pmax/0.1353 datafile="$'ExtractDataFile'"

#save outf=$IVName

#################
#OUTPUT
#################
#tonyplot IVCurve.dat

log outfile=done.log
log off
```

2. GaAs

```
#Brian Sullivan (2010)
#This GaAs solar cell is based on the middle layer of Michael Tsutagawa's (16
# Nov 2008) CHAMPION CELL, which is 36.28% efficient @ 300 K:
# At 300 K, using the AM0nrel.spec file, his 0.82 InGaP (doped 5e19) / 3.9um
# GaAs (doped 5e19) / ORIG300um Ge (Emitter & Substrate doped 3e18) cell has
# Voc = 2.76425 V, Jsc = 19.8531 mA/cm^2, Pmax = 0.0490907 mW/cm^2, Vmax =
# 2.525 V, Jmax = 19.4419 mA/cm^2, FF = 0.894527, and Eff = 0.362829.

# Modifications from Tsutagawa's input deck have been made in order to change
# the design quickly using the set command and to run the simulations using
# Matlab. However, the integrity of the original design is maintained.

# Main Silvaco ATLAS Structure
go atlas
# Definition of Constants
# Mesh
#   X-Mesh
#   Y-Mesh
# Regions
# Electrodes
# Doping
# Material Properties
# Models
```

```
# Light beams
# Solving

##################
# Constants
##################
#Filename of the IV curve output
set IVName=GaAsIVCurve.log
#Filename of the extracted P-V curve
set ExtractPowerFile=GaAsPower.log
#Fileanme of the extracted output
set ExtractDataFile=GaAsData.txt
#Filename of the structure
set StructFile=GaAsProfile.str

#Temperature
set temperature=275

#Baseline number of divisions/layer (10 is about the lowest that can be used)
set divs=10
#Width of cell (microns)
set cellWidth=500

#Cell width
set width3d=100e6/$cellWidth
set cellWidthHalf=$cellWidth/2
set cellWidthDiv1=$cellWidth/$divs
set cellWidthDiv2=$cellWidth/$divs/5

#Vacuum thickness (microns)
set vacuumThick=0.05

##Second layer
#Window thickness (microns)
set windowThick2=0.01
#Window doping (atoms/cc)
set winDop2=4.64e+017
#Emitter thickness (microns)
set emitterThick2=0.01
#Emitter doping (atoms/cc)
set emitDop2=4.64e+015
#Base thickness (microns)
set baseThick2=3.87
#Base doping (atoms/cc)
set baseDop2=1e+017
#BSF thickness (microns)
set bsfThick2=0.01
#BSF doping (microns)
set bsfDop2=5e19
#Buffer thickness (microns)
set bufferThick2=0.3
#Buffer doping (microns)
set bufferDop2=7e+018

#Dimensions based on other dimensions
set bufferLo2=4.21
set bufferHi2=$bufferLo2-$bufferThick2
set bufferDiv2=$bufferThick2/$divs/$divs
set bsfLo2=$bufferHi2
set bsfHi2=$bsfLo2-$bsfThick2
set bsfDiv2=$bsfThick2/$divs*2
set baseLo2=$bsfHi2
```

124

```
set baseHi2=$baseLo2-$baseThick2
set baseMid2=$baseLo2-$baseThick2/2
set baseDiv2=$baseThick2/$divs
set baseDiv2b=$baseThick2/$divs/2
set emitterLo2=$baseHi2
set emitterHi2=$emitterLo2-$emitterThick2
set emitterDiv2=$emitterThick2/$divs
set windowLo2=$emitterHi2
set windowHi2=$windowLo2-$windowThick2
set windowDiv2=$windowThick2/$divs

set vacuumLo=$windowHi2
set vacuumHi=$vacuumLo-$vacuumThick
set vacuumDiv=$vacuumThick/$divs

##################
# Mesh
##################
#Define the X-Mesh.-Must start from left to right or lesser to greater values.
        #mesh width=200000
mesh width=$width3d
        #mesh space.mult=1
# X-mesh:  surface=500 um^2 = 1/200000 cm^2
x.mesh loc=-$cellWidthHalf spac=$cellWidthDiv1
x.mesh loc=0                spac=$cellWidthDiv2
x.mesh loc=$cellWidthHalf  spac=$cellWidthDiv1

#Define the Y-Mesh.-Must start from surface of the device to the back, or go
#                    from lesser to greater value with up as negative.
# Vacuum
y.mesh loc=$vacuumLo   spac=$vacuumDiv

# Window (0.01 um)
y.mesh loc=$windowLo2    spac=$windowDiv2
# Emitter (0.01 um)
y.mesh loc=$emitterLo2    spac=$emitterDiv2
# Base (3.87 um)
y.mesh loc=$baseMid2   spac=$baseDiv2b
y.mesh loc=$baseLo2    spac=$baseDiv2
# BSF (0.01 um)
y.mesh loc=$bsfLo2    spac=$bsfDiv2
# Buffer (0.3 um)
y.mesh loc=$bufferLo2    spac=$bufferDiv2

##################
# REGIONS
##################
# Vacuum
region num=1 material=Vacuum x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$vacuumHi y.max=$vacuumLo

#Window AlInP (=InAsP)
region num=2 material=InAsP  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$windowHi2 y.max=$windowLo2
#Emitter
region num=3 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$emitterHi2 y.max=$emitterLo2
#Base
region num=4 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$baseHi2 y.max=$baseLo2
#BSF
region num=5 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
```

125

```
            y.min=$bsfHi2    y.max=$bsfLo2
#Buffer
region num=6 material=GaAs x.min=-$cellWidthHalf x.max=$cellWidthHalf \
        y.min=$bufferHi2 y.max=$bufferLo2


#####################
# ELECTRODES - Use 2% of cell width for optimal grid contact width
#####################      (500 x 0.02 = 10 um)
electrode name=cathod x.min=-5 x.max=5 y.min=$vacuumLo y.max=$vacuumLo
electrode name=anode x.min=-$cellWidthHalf x.max=$cellWidthHalf \
        y.min=$bufferLo2 y.max=$bufferLo2


####################
# DOPING - n-emitter on p-base
####################
# Window
doping uniform region=2 n.type conc=$winDop2
# Emitter
doping uniform region=3 n.type conc=$emitDop2
# Base
doping uniform region=4 p.type conc=$baseDop2
# BSF - Realistic from 1e20
doping uniform region=5 p.type conc=$bsfDop2
# Buffer - May want to change to 1e18.
doping uniform region=6 p.type conc=$bufferDop2


####################
# MATERIAL PROPERTIES
####################
material TAUN=1e-7 TAUP=1e-7 COPT=1.5e-10 AUGN=8.3e-32 AUGP=1.8e-31

# Vacuum
material material=Vacuum real.index=3.3 imag.index=0

# InGaP
material material=InGaP EG300=1.9 PERMITTIVITY=11.62 AFFINITY=4.16
material material=InGaP MUN=1945 MUP=141
material material=InGaP NC300=1.3e20 NV300=1.28e19
material material=InGaP index.file=InGaP-1.9.opt

# GaAs
material material=GaAs EG300=1.42 PERMITTIVITY=13.1 AFFINITY=4.07
material material=GaAs MUN=8800 MUP=400
material material=GaAs NC300=4.7e17 NV300=7e18
material material=GaAs index.file=GaAs.opt

# AlInP (=InAsP)
material material=InAsP EG300=2.4 PERMITTIVITY=11.7 AFFINITY=4.2
material material=InAsP MUN=2291 MUP=142
material material=InAsP NC300=1.08e20 NV300=1.28e19
material material=InAsP index.file=AlInP.opt

#Gold
material material=Gold real.index=1.2 imag.index=1.8


####################
# MODELS
####################
models srh temp=$temperature print
```

```
#####################
# LIGHTBEAMS -Calculates the position of the light source and defines the beam.
#####################
beam num=1 x.origin=0 y.origin=-5 angle=90 back.refl \
       power.file=AM0nrel.spec wavel.start=0.12 wavel.end=2.4 wavel.num=1500

#Output and Display the Structure file
struct outfile=$StructFile
#tonyplot $StructFile

####################
# SOLVING I-V CURVE
####################
method gummel newton maxtraps=10 itlimit=25 climit=1 dvmax=0.1

solve initial
#Beam strength of 1.0 sun
solve b1=1.0

log outfile=$IVName

solve vanode=0.0 name=anode vstep=0.05 vfinal=0.85
solve vanode=0.85 name=anode vstep=0.005 vfinal=1.115

extract init infile="$'IVName'"
extract name="Jsc" y.val from curve(v."anode," i."cathode") where x.val=0.0 \
       datafile="$'ExtractDataFile'"
extract name="Voc" x.val from curve(v."anode," i."cathode") where y.val=0.0 \
       datafile="$'ExtractDataFile'"
extract name="iv"  curve(v."anode," i."cathode") outfile="IVCurve.dat"

extract name="Power" curve(v."anode," (v."anode" * i."cathode")) \
       outf="$'ExtractPowerFile'"
extract name="Pmax" max(curve(v."anode," (v."anode" * i."cathode"))) \
       datafile="$'ExtractPowerFile'"
extract name="Vmax" x.val from curve(v."anode," (v."anode"*i."cathode") ) \
       where y.val=$"Pmax" datafile="$'ExtractDataFile'"
extract name="Jmax" $"Pmax"/$"Vmax" datafile="$'ExtractDataFile'"
extract name="FF" $"Pmax"/($"Jsc"*$"Voc") datafile="$'ExtractDataFile'"
#Use AMO of 0.1353 W/cm^2
extract name="Eff" $Pmax/0.1353 datafile="$'ExtractDataFile'"

#save outf=$IVName

#################
#OUTPUT
#################
#tonyplot IVCurve.dat

log outfile=done.log
log off
```

3.   Ge

```
#Brian Sullivan (2010)
#This Ge solar cell is based on the bottom layer of Michael Tsutagawa's (16
# Nov 2008) CHAMPION CELL, which is 36.28% efficient @ 300 K:
# At 300 K, using the AM0nrel.spec file, his 0.82 InGaP (doped 5e19) / 3.9um
# GaAs (doped 5e19) / ORIG300um Ge (Emitter & Substrate doped 3e18) cell has
# Voc = 2.76425 V, Jsc = 19.8531 mA/cm^2, Pmax = 0.0490907 mW/cm^2, Vmax =
# 2.525 V, Jmax = 19.4419 mA/cm^2, FF = 0.894527, and Eff = 0.362829.
```

```
# Modifications from Tsutagawa's input deck have been made in order to change
# the design quickly using the set command and to run the simulations using
# Matlab. However, the integrity of the original design is maintained.

# Main Silvaco ATLAS Structure
go atlas
# Definition of Constants
# Mesh
#    X-Mesh
#    Y-Mesh
# Regions
# Electrodes
# Doping
# Material Properties
# Models
# Light beams
# Solving


##################
# Constants
##################
#Filename of the IV curve output
set IVName=GeIVCurve.log
#Filename of the extracted P-V curve
set ExtractPowerFile=GePower.log
#Fileanme of the extracted output
set ExtractDataFile=GeData.txt
#Filename of the structure
set StructFile=GeProfile.str

#Temperature
set temperature=275

#Baseline number of divisions/layer (10 is about the lowest that can be used)
set divs=10
#Width of cell (microns)
set cellWidth=500

#Cell width
set width3d=100e6/$cellWidth
set cellWidthHalf=$cellWidth/2
set cellWidthDiv1=$cellWidth/$divs
set cellWidthDiv2=$cellWidth/$divs/5

#Vacuum thickness (microns)
set vacuumThick=0.05

##Third layer
#Window thickness (microns)
set windowThick3=0.05
#Window doping (atoms/cc)
set winDop3=7e+018
#Emitter thickness (microns)
set emitterThick3=0.1
#Emitter doping (atoms/cc)
set emitDop3=3e+018
#Base thickness (microns)
set baseThick3=300
#Base doping (atoms/cc)
set baseDop3=3e+018
```

```
#Dimensions based on other dimensions
set baseLo3=304.39
set baseHi3=$baseLo3-$baseThick3
set baseMid3=$baseLo3-$baseThick3/2
set baseDiv3=$baseThick3/$divs
set emitterLo3=$baseHi3
set emitterHi3=$emitterLo3-$emitterThick3
set emitterDiv3=$emitterThick3/$divs
set windowLo3=$emitterHi3
set windowHi3=$windowLo3-$windowThick3
set windowDiv3=$windowThick3/$divs

set vacuumLo=$windowHi3
set vacuumHi=$vacuumLo-$vacuumThick
set vacuumDiv=$vacuumThick/$divs


##################
# Mesh
##################
#Define the X-Mesh.-Must start from left to right or lesser to greater values.
        #mesh width=200000
mesh width=$width3d
        #mesh space.mult=1
# X-mesh:  surface=500 um2= 1/200000 cm2
x.mesh loc=-$cellWidthHalf spac=$cellWidthDiv1
x.mesh loc=0               spac=$cellWidthDiv2
x.mesh loc=$cellWidthHalf  spac=$cellWidthDiv1

#Define the Y-Mesh.-Must start from surface of the device to the back, or go
#                   from lesser to greater value with up as negative.
# Vacuum
y.mesh loc=$vacuumLo  spac=$vacuumDiv

# Window (0.05 um) - ORIGINAL Cell parameters
y.mesh loc=$windowLo3    spac=$windowDiv3
# Emitter (0.1 um)
y.mesh loc=$emitterLo3   spac=$emitterDiv3
# Substrate (300 um)
y.mesh loc=$baseLo3 spac=$baseDiv3

##################
# REGIONS
##################
# Vacuum
region num=1 material=Vacuum x.min=-$cellWidthHalf x.max=$cellWidthHalf \
        y.min=$vacuumHi y.max=$vacuumLo

#Window
region num=2 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
        y.min=$windowHi3 y.max=$windowLo3
#Emitter
region num=3 material=Ge    x.min=-$cellWidthHalf x.max=$cellWidthHalf \
        y.min=$emitterHi3 y.max=$emitterLo3
#Substrate
region num=4 material=Ge x.min=-$cellWidthHalf x.max=$cellWidthHalf \
        y.min=$baseHi3 y.max=$baseLo3


#####################
# ELECTRODES - Use 2% of cell width for optimal grid contact width
#####################     (500 x 0.02 = 10 um)
electrode name=cathode x.min=-5 x.max=5 y.min=$vacuumLo y.max=$vacuumLo
```

```
electrode name=anode x.min=-$cellWidthHalf x.max=$cellWidthHalf \
        y.min=$baseLo3 y.max=$baseLo3


####################
# DOPING - n-emitter on p-base
####################
# Window
doping uniform region=2 n.type conc=$winDop3
# Emitter - Changed from 2e18
doping uniform region=3 n.type conc=$emitDop3
# Substrate - Changed from 1e17
doping uniform region=4 p.type conc=$baseDop3


####################
# MATERIAL PROPERTIES
####################
material TAUN=1e-7 TAUP=1e-7 COPT=1.5e-10 AUGN=8.3e-32 AUGP=1.8e-31

# Vacuum
material material=Vacuum real.index=3.3 imag.index=0

# GaAs
material material=GaAs EG300=1.42 PERMITTIVITY=13.1 AFFINITY=4.07
material material=GaAs MUN=8800 MUP=400
material material=GaAs NC300=4.7e17 NV300=7e18
material material=GaAs index.file=GaAs.opt

# Ge
material material=Ge EG300=0.67 PERMITTIVITY=16 AFFINITY=4
material material=Ge MUN=3900 MUP=1800
material material=Ge NC300=1.04e19 NV300=6e18
material material=Ge index.file=Ge.opt

#Gold
material material=Gold real.index=1.2 imag.index=1.8


####################
# MODELS
####################
models srh temp=$temperature print


####################
# LIGHTBEAMS -Calculates the position of the light source and defines the beam.
####################
beam num=1 x.origin=0 y.origin=-5 angle=90 back.refl \
        power.file=AM0nrel.spec wavel.start=0.12 wavel.end=2.4 wavel.num=1500

#Output and Display the Structure file
struct outfile=$StructFile
#tonyplot $StructFile


####################
# SOLVING I-V CURVE
####################
method gummel newton maxtraps=10 itlimit=25 climit=1 dvmax=0.1

solve initial
#Beam strength of 1.0 sun
solve b1=1.0

log outfile=$IVName
```

```
solve vanode=0.0 name=anode vstep=0.05 vfinal=0.18
solve vanode=0.18 name=anode vstep=0.005 vfinal=0.35

extract init infile="$'IVName'"
extract name="Jsc" y.val from curve(v."anode," i."cathode") where x.val=0.0 \
      datafile="$'ExtractDataFile'"
extract name="Voc" x.val from curve(v."anode," i."cathode") where y.val=0.0 \
      datafile="$'ExtractDataFile'"
extract name="iv"  curve(v."anode," i."cathode") outfile="IVCurve.dat"

extract name="Power" curve(v."anode," (v."anode" * i."cathode")) \
      outf="$'ExtractPowerFile'"
extract name="Pmax" max(curve(v."anode," (v."anode" * i."cathode"))) \
      datafile="$'ExtractPowerFile'"
extract name="Vmax" x.val from curve(v."anode," (v."anode"*i."cathode") ) \
      where y.val=$"Pmax" datafile="$'ExtractDataFile'"
extract name="Jmax" $"Pmax"/$"Vmax" datafile="$'ExtractDataFile'"
extract name="FF" $"Pmax"/($"Jsc"*$"Voc") datafile="$'ExtractDataFile'"
#Use AMO of 0.1353 W/cm^2
extract name="Eff" $Pmax/0.1353 datafile="$'ExtractDataFile'"

#save outf=$IVName

#################
#OUTPUT
#################
#tonyplot IVCurve.dat

log outfile=done.log
log off
```

C.   TRIPLE-JUNCTION CELL REDESIGNED

Increasing the GaAs emitter and base doping of each subcell along with the InGaP mole fraction in the triple-junction cell causes an increase in the cell's efficiency at 300 to 400 K. The Silvaco DeckBuild code for this redesigned cell is presented below.

```
#Brian Sullivan (Sep 2010)
#This solar cell design is based on Michael Tsutagawa's (16 Nov 2008) CHAMPION
# CELL, which is 36.28% efficient @ 300 K. The doping in the base of each
# subcell and the GaAs emitter have been increased and the InGaP mole fraction
# has been increased from x = 0.51 to x = 0.69. These changes result in a cell
# that is 40.42% efficient at 300 K and 33.89% efficient at 400 K.


# Main Silvaco ATLAS Structure
go atlas
# Definition of Constants
# Mesh
#   X-Mesh
#   Y-Mesh
# Regions
# Electrodes
# Doping
# Material Properties
```

131

```
# Models
# Light beams
# Solving

##################
# Constants
##################
#Filename of the IV curve output
set IVName=InGaP_x_GaAs_Ge_IVCurve.log
#Filename of the extracted P-V curve
set ExtractPowerFile=InGaP_x_GaAs_Ge_Power.log
#Fileanme of the extracted output
set ExtractDataFile=InGaP_x_GaAs_Ge_Data.txt
#Filename of the structure
set StructFile=InGaP_x_GaAs_Ge_Profile.str

#Molar concentration of In(1-x)Ga(x)P
set x=0.69

#Temperature
set temperature=275

#Baseline number of divisions/layer (10 is about the lowest that can be used)
set divs=10
#Width of cell (microns)
set cellWidth=500

#Cell width
set width3d=100e6/$cellWidth
set cellWidthHalf=$cellWidth/2
set cellWidthDiv1=$cellWidth/$divs
set cellWidthDiv2=$cellWidth/$divs/5

#Vacuum thickness (microns)
set vacuumThick=0.05

##First layer dimensions
#Window thickness (microns)
set windowThick1=0.01
#Window doping (atoms/cc)
set winDop1=5e+019
#Emitter thickness (microns)
set emitterThick1=0.17
#Emitter doping (atoms/cc)
set emitDop1=4.64e+017
#Base thickness (microns)
set baseThick1=0.63
#Base doping (atoms/cc) 1e+017
set baseDop1=1e+018
#BSF thickness (microns)
set bsfThick1=0.01
#BSF doping (microns)
set bsfDop1=5e+019
#Buffer thickness (microns)
set bufferThick1=0.03
#Buffer doping (microns)
set bufferDop1=1e+018

##First tunnel junction
#Emitter thickness (microns)
set tunnelEmitThick1=0.015
#Emitter doping (atoms/cc)
```

```
set tunnelEmitDop1=8e18
#Base thickness (microns)
set tunnelBaseThick1=0.015
#Base doping (atoms/cc)
set tunnelBaseDop1=1e19

##Second layer
#Window thickness (microns)
set windowThick2=0.01
#Window doping (atoms/cc)
set winDop2=4.64e+017
#Emitter thickness (microns)
set emitterThick2=0.01
#Emitter doping (atoms/cc) 4.64e+015
set emitDop2=4.64e+016
#Base thickness (microns)
set baseThick2=3.87
#Base doping (atoms/cc) 1e+017
set baseDop2=1e+018
#BSF thickness (microns)
set bsfThick2=0.01
#BSF doping (microns)
set bsfDop2=5e19
#Buffer thickness (microns)
set bufferThick2=0.3
#Buffer doping (microns)
set bufferDop2=7e+018

##Second tunnel junction
#Emitter thickness (microns)
set tunnelEmitThick2=0.015
#Emitter doping (atoms/cc)
set tunnelEmitDop2=8e18
#Base thickness (microns)
set tunnelBaseThick2=0.015
#Base doping (atoms/cc)
set tunnelBaseDop2=1e19

##Third layer
#Window thickness (microns)
set windowThick3=0.05
#Window doping (atoms/cc)
set winDop3=7e+018
#Emitter thickness (microns)
set emitterThick3=0.1
#Emitter doping (atoms/cc)
set emitDop3=3e+018
#Base thickness (microns)
set baseThick3=300
#Base doping (atoms/cc) 3e+018
set baseDop3=1.614e+019

#Dimensions based on other dimensions
set baseLo3=304.39
set baseHi3=$baseLo3-$baseThick3
        #set baseMid3=$baseLo3-$baseThick3/2
        #set baseDiv3=$baseThick3/$divs
set emitterLo3=$baseHi3
set emitterHi3=$emitterLo3-$emitterThick3
        #set emitterDiv3=$emitterThick3/$divs
set windowLo3=$emitterHi3
set windowHi3=$windowLo3-$windowThick3
```

```
        #set windowDiv3=$windowThick3/$divs*2

set tunnelBaseLo2=$windowHi3
set tunnelBaseHi2=$tunnelBaseLo2-$tunnelBaseThick2
        #set tunnelBaseDiv2=$tunnelBaseThick2/$divs
set tunnelEmitterLo2=$tunnelBaseHi2
set tunnelEmitterHi2=$tunnelEmitterLo2-$tunnelEmitThick2
        #set tunnelEmitterDiv2=$tunnelEmitThick2/$divs

set bufferLo2=$tunnelEmitterHi2
set bufferHi2=$bufferLo2-$bufferThick2
        #set bufferDiv2=$bufferThick2/$divs
set bsfLo2=$bufferHi2
set bsfHi2=$bsfLo2-$bsfThick2
        #set bsfDiv2=$bsfThick2/$divs*2
set baseLo2=$bsfHi2
set baseHi2=$baseLo2-$baseThick2
set baseMid2=$baseLo2-$baseThick2/2
        #set baseDiv2=$baseThick2/$divs
set emitterLo2=$baseHi2
set emitterHi2=$emitterLo2-$emitterThick2
        #set emitterDiv2=$emitterThick2/$divs
set windowLo2=$emitterHi2
set windowHi2=$windowLo2-$windowThick2
        #set windowDiv2=$windowThick2/$divs*2

set tunnelBaseLo1=$windowHi2
set tunnelBaseHi1=$tunnelBaseLo1-$tunnelBaseThick1
        #set tunnelBaseDiv1=$tunnelBaseThick1/$divs
set tunnelEmitterLo1=$tunnelBaseHi1
set tunnelEmitterHi1=$tunnelEmitterLo1-$tunnelEmitThick1
        #set tunnelEmitterDiv1=$tunnelEmitThick1/$divs

set bufferLo1=$tunnelEmitterHi1
set bufferHi1=$bufferLo1-$bufferThick1
        #set bufferDiv1=$bufferThick1/$divs
set bsfLo1=$bufferHi1
set bsfHi1=$bsfLo1-$bsfThick1
        #set bsfDiv1=$bsfThick1/$divs*2
set baseLo1=$bsfHi1
set baseHi1=$baseLo1-$baseThick1
set baseMid1=$baseLo1-$baseThick1/2
        #set baseDiv1=$baseThick1/$divs
set emitterLo1=$baseHi1
set emitterHi1=$emitterLo1-$emitterThick1
        #set emitterDiv1=$emitterThick1/$divs
set windowLo1=$emitterHi1
set windowHi1=$windowLo1-$windowThick1
        #set windowDiv1=$windowThick1/$divs*2

set vacuumLo=$windowHi1
set vacuumHi=$vacuumLo-$vacuumThick
        #set vacuumDiv=$vacuumThick/$divs

set lightY=$windowHi1-5

##################
# Mesh
##################
#Define X-Mesh. Must start from left to right or from lesser to greater values.
        #mesh width=200000
mesh width=$width3d
```
134

```
        #mesh space.mult=1
# X-mesh:  surface=500 um^2= 1/200000 cm^2
x.mesh loc=-$cellWidthHalf spac=$cellWidthDiv1
x.mesh loc=0              spac=$cellWidthDiv2
x.mesh loc=$cellWidthHalf  spac=$cellWidthDiv1

#Define Y-Mesh.  Must start from surface of the device to the back or go from #
                 lesser to greater value with up as negative.
# Vacuum
y.mesh loc=$vacuumLo  spac=0.003
# Window (0.01 um)
y.mesh loc=$windowLo1  spac=0.003
# Emitter (0.17 um)
y.mesh loc=$emitterLo1  spac=0.003
# Base (0.63 um)
y.mesh loc=$baseMid1  spac=0.03
y.mesh loc=$baseLo1  spac=0.003
# BSF (0.01 um)
y.mesh loc=$bsfLo1  spac=0.003
# Buffer (0.03 um)
y.mesh loc=$bufferLo1  spac=0.002

# Tunnel Emitter (0.015)
y.mesh loc=$tunnelEmitterLo1 spac=0.001
# Tunnel Base (0.015)
y.mesh loc=$tunnelBaseLo1    spac=0.001

# Window (0.01 um)
y.mesh loc=$windowLo2   spac=0.003
# Emitter (0.01 um)
y.mesh loc=$emitterLo2   spac=0.003
# Base (3.87 um)
y.mesh loc=$baseMid2  spac=0.03
y.mesh loc=$baseLo2    spac=0.003
# BSF (0.01 um)
y.mesh loc=$bsfLo2    spac=0.003
# Buffer (0.3 um)
y.mesh loc=$bufferLo2    spac=0.002

# Tunnel Emitter (0.015)
y.mesh loc=$tunnelEmitterLo2  spac=0.001
# Tunnel Base (0.015)
y.mesh loc=$tunnelBaseLo2    spac=0.001

# Window (0.05 um) - ORIGINAL Cell parameters
y.mesh loc=$windowLo3    spac=0.01
# Emitter (0.1 um)
y.mesh loc=$emitterLo3    spac=0.01
# Substrate (300 um)
y.mesh loc=$baseLo3 spac=50

#################
# REGIONS
#################
# Vacuum
region num=1 material=Vacuum x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$vacuumHi   y.max=$vacuumLo
#Window AlInP (=InAsP)
region num=2 material=InAsP  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
      y.min=$windowHi1  y.max=$windowLo1
#Emitter
region num=3 material=InGaP  x.comp=$x  x.min=-$cellWidthHalf \
```
135

```
           x.max=$cellWidthHalf  y.min=$emitterHi1  y.max=$emitterLo1
#Base
region num=4 material=InGaP  x.comp=$x  x.min=-$cellWidthHalf \
       x.max=$cellWidthHalf  y.min=$baseHi1  y.max=$baseLo1
#BSF
region num=5 material=InGaP  x.comp=$x  x.min=-$cellWidthHalf \
       x.max=$cellWidthHalf  y.min=$bsfHi1  y.max=$bsfLo1
#Buffer AlInP (=InAsP)
region num=6 material=InAsP  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$bufferHi1  y.max=$bufferLo1


#Tunnel Emitter
region num=7 material=InGaP  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelEmitterHi1  y.max=$tunnelEmitterLo1
#Tunnel Base
region num=8 material=InGaP  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelBaseHi1  y.max=$tunnelBaseLo1


#Window AlInP (=InAsP)
region num=9 material=InAsP  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$windowHi2  y.max=$windowLo2
#Emitter
region num=10 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$emitterHi2  y.max=$emitterLo2
#Base
region num=11 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$baseHi2   y.max=$baseLo2
#BSF
region num=12 material=InGaP x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$bsfHi2    y.max=$bsfLo2
#Buffer
region num=13 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$bufferHi2   y.max=$bufferLo2


#Tunnel Emitter
region num=14 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelEmitterHi2   y.max=$tunnelEmitterLo2
#Tunnel Base
region num=15 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelBaseHi2  y.max=$tunnelBaseLo2


#Window
region num=16 material=GaAs  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$windowHi3  y.max=$windowLo3
#Emitter
region num=17 material=Ge    x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$emitterHi3  y.max=$emitterLo3
#Substrate
region num=18 material=Ge    x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$baseHi3  y.max=$baseLo3


####################
# ELECTRODES - Use 2% of cell width for optimal grid contact width
#            (500x0.02 = 10um)
####################
electrode name=cathode    x.min=-5 x.max=5 y.min=$vacuumLo y.max=$vacuumLo
electrode name=tunneltop  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelEmitterHi1    y.max=$tunnelBaseLo1
electrode name=tunnelbot  x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$tunnelEmitterHi2    y.max=$tunnelBaseLo2
electrode name=anode      x.min=-$cellWidthHalf x.max=$cellWidthHalf \
       y.min=$baseLo3            y.max=$baseLo3
```

```
####################
# DOPING - n-emitter on p-base
####################
# Window - Realistic from 1e20
doping uniform region=2 n.type conc=$winDop1
# Emitter
doping uniform region=3 n.type conc=$emitDop1
# Base
doping uniform region=4 p.type conc=$baseDop1
# BSF - Realistic from 1e20
doping uniform region=5 p.type conc=$bsfDop1
# Buffer
doping uniform region=6 p.type conc=$bufferDop1

# Tunnel Emitter
doping uniform region=7 p.type conc=$tunnelEmitDop1
# Tunnel Base
doping uniform region=8 n.type conc=$tunnelBaseDop1

# Window
doping uniform region=9 n.type conc=$winDop2
# Emitter
doping uniform region=10 n.type conc=$emitDop2
# Base
doping uniform region=11 p.type conc=$baseDop2
# BSF - Realistic from 1e20
doping uniform region=12 p.type conc=$bsfDop2
# Buffer - May want to change to 1e18.
doping uniform region=13 p.type conc=$bufferDop2

# Tunnel Emitter
doping uniform region=14 p.type conc=$tunnelEmitDop2
# Tunnel Base
doping uniform region=15 n.type conc=$tunnelBaseDop2

# Window
doping uniform region=16 n.type conc=$winDop3
# Emitter - Changed from 2e18
doping uniform region=17 n.type conc=$emitDop3
# Substrate - Changed from 1e17
doping uniform region=18 p.type conc=$baseDop3

####################
# MATERIAL PROPERTIES
####################
material TAUN=1e-7 TAUP=1e-7 COPT=1.5e-10 AUGN=8.3e-32 AUGP=1.8e-31

# Vacuum
material material=Vacuum real.index=3.3 imag.index=0

# In(1-x)Ga(x)P
material material=InGaP EG300=1.35+0.732*$x+0.7*$x^2
material material=InGaP PERMITTIVITY=12.5-1.4*$x AFFINITY=4.38-0.58*$x
material material=InGaP \
    MUN=33000+(8500-33000)*$x+4600+(300-4600)*$x-(33000+(8500-33000)*$x)
material material=InGaP \
    MUP=460+(400-460)*$x+150+(100-150)*$x-(460+(400-460)*$x)
#    material material=InGaP NC300=1.3e20 NV300=1.28e19
material material=InGaP index.file=InGaP-0.4-0.6.opt

# GaAs
```

```
material material=GaAs EG300=1.42 PERMITTIVITY=13.1 AFFINITY=4.07
material material=GaAs MUN=8800 MUP=400
material material=GaAs NC300=4.7e17 NV300=7e18
material material=GaAs index.file=GaAs.opt

# Ge
material material=Ge EG300=0.67 PERMITTIVITY=16 AFFINITY=4
material material=Ge MUN=3900 MUP=1800
material material=Ge NC300=1.04e19 NV300=6e18
material material=Ge index.file=Ge.opt

# AlInP (=InAsP)
material material=InAsP EG300=2.4 PERMITTIVITY=11.7 AFFINITY=4.2
material material=InAsP MUN=2291 MUP=142
material material=InAsP NC300=1.08e20 NV300=1.28e19
material material=InAsP index.file=AlInP.opt

#Gold
material material=Gold real.index=1.2 imag.index=1.8


####################
# MODELS
####################
models srh bgn temp=$temperature print


####################
# CONTACTS
####################
contact name=tunneltop resist=1e17
contact name=tunnelbot resist=1e17


####################
# LIGHTBEAMS - Calculates position of light source and defines beam.
####################
beam num=1 x.origin=0 y.origin=$lightY angle=90 back.refl \
      power.file=AM0nrel.spec wavel.start=0.12 wavel.end=2.4 wavel.num=1500

#Output and Display the Structure file
struct outfile=$StructFile
#tonyplot $StructFile


####################
# SOLVING I-V CURVE
####################
method gummel newton maxtraps=50 itlimit=100 climit=1 dvmax=0.1

solve initial
#Beam strength of 1.0 sun
solve b1=1.0

log outfile=$IVName

solve vanode=0.0 name=anode vstep=0.1 vfinal=2.8
solve vanode=2.8 name=anode vstep=0.01 vfinal=3.1

extract init infile="$'IVName'"
extract name="Jsc" y.val from curve(v."anode," i."cathode") where x.val=0.0 \
      datafile="$'ExtractDataFile'"
extract name="Voc" x.val from curve(v."anode," i."cathode") where y.val=0.0 \
      datafile="$'ExtractDataFile'"
extract name="iv"  curve(v."anode," i."cathode") outfile="IVCurve.dat"
```

```
extract name="Power" curve(v."anode," (v."anode" * i."cathode")) \
        outf="$'ExtractPowerFile'"
extract name="Pmax" max(curve(v."anode," (v."anode" * i."cathode"))) \
        datafile="$'ExtractPowerFile'"
extract name="Vmax" x.val from curve(v."anode," (v."anode"*i."cathode") ) \
        where y.val=$"Pmax" datafile="$'ExtractDataFile'"
extract name="Jmax" $"Pmax"/$"Vmax" datafile="$'ExtractDataFile'"
extract name="FF" $"Pmax"/($"Jsc"*$"Voc") datafile="$'ExtractDataFile'"
#Use AM0 of 0.1353 W/cm^2
extract name="Eff" $Pmax/0.1353 datafile="$'ExtractDataFile'"

#save outf=$IVName

#################
#OUTPUT
#################
#tonyplot IVCurve.dat

log outfile=done.log
log off
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B.   MATLAB SOURCE CODE

This appendix contains the Matlab script and function files used in this thesis. Section A has an example of the script used to run Silvaco input decks. Section B contains the function files used in the script.

A.   MATLAB RUNNING ATLAS

The Matlab script atlusrun2_parm1_parm2.m presented below is useful for running multiple iterations of Silvaco ATLAS. In this script, one parameter is iterated for each iteration of another parameter. This particular example of the Matlab scripts used in this thesis opens and runs the Silvaco input deck InGaP_x_GaAs_Ge.in changing the temperature from 275 to 400 K in 25-K increments for 20 iterations of x, the mole fraction of InGaP.

```matlab
%atlasrun_InGaP_x/GaAs/Ge_MoleFrac&Temp.m
%
%This script runs iterations of ATLAS. It is based on the atlasrun.m
%and atlasrun_parm1_parm2.m scripts developed by Davenport and
%Canfield, respectively. Like Canfield's script, it has an inner and
%outer loop of iteration. This script uses ivmaxp.m modified to
%calculate open-circuit voltage, short-circuit current, maximum
%voltage, maximum current, maximum power, fill factor, and efficiency.
%The function filerw.m is called to iterate the ATLAS input deck. The
%function time.m is called to calculate the time of each iteration in
%hours, minutes, and seconds. Davenport's original version iterated
%only upon one parameter and this parameter had to be specified in
%filerw.m. Like Canfield's script file, the parameters do not have to
%be specified in filerw.m and two variables are iterated with the
%resulting data saved in a unique spreadsheet on different tabs.

close all
clear all
clc

%Specifies the number of iterations to perform for the outer loop
ITERATE = 20;
CNT     = 1;
%Specifies the range of values to iterate between for the outer loop
INIT    = 0.48; %Starting InGaP mole fraction (x)
INTR    = 0.48;
```

```matlab
RST      = 0.51; %Defined for resetting "INIT" using filerw.m
FIN      = 0.68; %Stopping InGaP mole fraction (x)
STP      = (FIN-INIT)/ITERATE;
B        = INIT;


%This will iterate once each outer loop
firststring = 'set x=%g';
%This will iterate once each inner loop
secondstring = 'set temperature=%g';
%Starting outer loop to iterate first parameter
for yy=1:1:ITERATE+1
    filerw('InGaP_x_GaAs_Ge.in',firststring,INIT,B)
    INIT=B  %Saving the old x value
    B=B+STP;  %Iterating to the next x value

    %Specifies the number of iterations to perform for the inner loop
    iterations=5;
    count=1;
    %Specifies the range of values to iterate for the inner loop
    old=275; %Starting temperature
    ol=275;
    rst=275; %Defined for resetting "old" using filerw.m
    final=400; %Stopping temperature
    step=(final-old)/iterations;

    k=0;
    totalruntime=0;
    %Starting the inner loop to iterate the second parameter
    for a=ol:step:final
        tic
        k=k+1;
        fidres=fopen('atres.txt','a');
        filerw('InGaP_x_GaAs_Ge.in',secondstring,old,a)
        delete done.log
        delete atlaslog.log

        %Here atlas is called by Matlab to run the specified input deck
        !C:\Silvaco\etc\GuiAppStarter.exe -lib-dir-name deckbuild -exe-
name Deckbld -run InGaP_x_GaAs_Ge.in -outfile atlaslog.log

        sprintf('Executing run %u\nStandby for results',k)
        xy=-1;

        while(xy==-1)
            xy=fopen('done.log');
            pause(3)
        end

        %Here the *.log file created by ATLAS is read and open-circuit
        %voltage, short-circuit current, maximum voltage, maximum
        %current, maximum power, fill factor and efficiency are
        %returned.

[Voc,Jsc,Vmax,Jmax,Pmax,FF,Eff]=ivmaxp3('InGaPGaAsGeIVCurve.log','n');
```

```matlab
        data(k,:)=[a Voc Jsc Vmax Jmax Pmax FF Eff];
        %The previously calculated data is written to file.
        fprintf(fidres,'%f %f %f %f %f %f %f %f\n',data(k,:));

        %After each iteration, all calculated data is printed to the
        %screen including the time it took to run the iteration. A plot
        %of efficiency is updated after each iteration.
        clc
        data
        thisruntime=toc;T1=time(toc);
        totalruntime=totalruntime+thisruntime;T2=time(totalruntime);
        averageruntime=totalruntime/k;T3=time(averageruntime);
        estimateruntime=averageruntime*(iterations+1);
        T4=time(estimateruntime);
        sprintf('This run took %s\nTotal run time so far is %s\nAverage
run time is %s\nEstimated total run time is %s',T1,T2,T3,T4)

        old=a
        count=count+1;
        fclose('all')
        pause(3)
        plot(data(:,1),data(:,8))
        title('Efficiency')
    end

    %The information vector contains the iterated outer loop variable
    %as well as emitter doping and thickness and base doping and
    %thickness.
    Information = [INIT, 0.17, 4.64e17, 0.63, 1e17];
    %Write data to a different sheet for each loop
    sheet=sprintf('Sheet%g',yy);

xlswrite('C:\Silvaco\Work\InGaPGaAsGe_MoleFrac&Temp.xls',Information,sh
eet,'A2:E2')

xlswrite('C:\Silvaco\Work\InGaPGaAsGe_MoleFrac&Temp.xls',data,sheet,'A4
:H9')

    filerw('InGaP_x_GaAs_Ge.in',secondstring,old,ol)
end

%Reseting the first iterated value to its original value.
filerw('InGaP_x_GaAs_Ge.in',firststring,INIT,RST)
%Reseting the second iterated value to its original value.
filerw('InGaP_x_GaAs_Ge.in',secondstring,old,rst)
```

B.   FUNCTIONS

The Matlab file filerw.m contains the function created by Davenport and used in the atlasrun2_parm1_parm2.m file presented in the previous section of this appendix. A modified version of the ivmaxp.m function created by

Davenport and used in the atlasrun2_parm1_parm2.m file is presented in the second and third sections. The second section contains the version of the ivmaxp.m file used for single-junction solar cell simulations and the third section contains the version used for triple-junction simulations.

1.   filerw.m

```
function filerw(file,string,old,new)
%filerw(file,old,new)
%This program opens an infile, "file," and writes over the
%"old" "string" with the "new" "string"
%file must be in '' with extension if running standalone.
%old and new are the old and new values of string
%This same function was also used to iterate upon any variable by
%replacing "firststring" and "secondstring" in atlasrun.m with the
%names of the variable to be iterated.

fidr=fopen(file,'r');
a=fscanf(fidr,'%c');
fclose(fidr);
fidw=fopen(file,'w');
a=strrep(a,sprintf(string,old),sprintf(string,new));
fwrite(fidw,a);
fclose(fidw);
```

2.   ivmaxp1.m

```
function [Voc,Jsc,Vmax,Jmax,Pmax,FF,Eff]=ivmaxp(varargin)
%[Voc,Jsc,Vmax,Jmax,Pmax,FF,Eff]=ivmaxp(file,type)
%Finds Voc, Jsc, Vmax, Jmax, Pmax, FF, and Efficiency from an ATLAS
%*.log file
%file and type must be in ''
%Type is the doping of the top layer, n or p

%THIS FUNCTION HAS BEEN MODIFIED TO WORK WITH SINGLE-LAYER SOLAR CELLS
%(n-type). - BRIAN SULLIVAN, 2010 (see comments below)

format long

if(nargin==1)
    file=char(varargin(1));
    type='P';
elseif(nargin==2)
    file=char(varargin(1))
    type=char(varargin(2))
else 'Type must be either "N," "n," "P," or "p"'
    Voc=NaN;
```

144

```matlab
    Jsc=NaN;
    Pmax=NaN;
    FF=NaN;
    return
end
if(type=='N'||type=='n')
    [I V]=textread(file,'%*s %*f %*f %*f %*f %*f %*f %*f %*f %*f %*f %f
%*f %*f %*f','headerlines',20);
%The above line has to be changed based on the format of the *.log file
%which changes based on the number of layers in the solar cell design.

elseif(type=='P'||type=='p')
    [V I]=textread(file,'%*s %*f %*f %*f %*f %*f %*f %*f %*f %f %*f %*f
%*f %*f %f','headerlines',20);
%The above line has to be changed based on the format of the *.log file
%which changes based on the number of layers in the solar cell design.

else
    'Type must be either "N," "n," "P," or "p"'
    Voc=NaN;
    Jsc=NaN;
    Pmax=NaN;
    FF=NaN;
    return
end
xi=0;
Voc=interp1(I,V,xi,'pchip');
Jsc=1000*interp1(V,I,xi,'pchip');
P=I.*V;
[a b]=max(P);
Pmax=1000*V(b)*I(b);
Vmax=V(b);
Jmax=1000*I(b);
FF=Pmax/(Voc*Jsc);
Eff=100*Pmax/(135.3);

x=1.1*Voc;
y=1.1*Jsc;
II=1000*I;
IIb=1000*I(b);
plot(V,II,V(b),IIb,'rh')
xlabel('Voltage (V)')
ylabel('Current (mA/cm^2)')
title('I-V Curve')
axis([0 x 0 y])
```

3. ivmaxp3.m

```matlab
function [Voc,Jsc,Vmax,Jmax,Pmax,FF,Eff]=ivmaxp(varargin)
%[Voc,Jsc,Vmax,Jmax,Pmax,FF,Eff]=ivmaxp(file,type)
%Finds Voc, Jsc, Vmax, Jmax, Pmax, FF, and Efficiency from an ATLAS
%*.log file
%file and type must be in ''
%Type is the doping of the top layer, n or p
```

145

```matlab
%THIS FUNCTION HAS BEEN MODIFIED TO WORK WITH TRIPLE-JUNCTION SOLAR
%CELLS (n-type). - BRIAN SULLIVAN, 2010 (see comments above)

format long

if(nargin==1)
    file=char(varargin(1));
    type='P';
elseif(nargin==2)
    file=char(varargin(1))
    type=char(varargin(2))
else 'Type must be either "N," "n," "P," or "p"'
    Voc=NaN;
    Jsc=NaN;
    Pmax=NaN;
    FF=NaN;
    return
end
if(type=='N'||type=='n')
    [I V]=textread(file,'%*s %*f %*f %*f %*f %*f %*f %*f %*f %*f %*f %f
%*f %*f %*f %*f %*f %*f %f %*f %*f','headerlines',20);
%The above line has to be changed based on the format of the *.log file
%which changes based on the number of layers in the solar cell design.
elseif(type=='P'||type=='p')
    [V I]=textread(file,'%*s %*f %*f %*f %*f %*f %*f %*f %*f %f %*f %*f
%*f %*f %f','headerlines',20);
%The above line has to be changed based on the format of the *.log file
%which changes based on the number of layers in the solar cell design.
else
    'Type must be either "N," "n," "P," or "p"'
    Voc=NaN;
    Jsc=NaN;
    Pmax=NaN;
    FF=NaN;
    return
end
xi=0;
Voc=interp1(I,V,xi,'pchip');
Jsc=1000*interp1(V,I,xi,'pchip');
P=I.*V;
[a b]=max(P);
Pmax=1000*V(b)*I(b);
Vmax=V(b);
Jmax=1000*I(b);
FF=Pmax/(Voc*Jsc);
Eff=100*Pmax/(135.3);

x=1.1*Voc;
y=1.1*Jsc;
II=1000*I;
IIb=1000*I(b);
plot(V,II,V(b),IIb,'rh')
xlabel('Voltage (V)')
ylabel('Current (mA/cm^2)')
```

146

```
title('I-V Curve')
axis([0 x 0 y])
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1]   P. Michalopoulous, "A novel approach for the development and optimization of state-of-the-art photovoltaic devices using Silvaco," Master's thesis, Naval Postgraduate School, 2002.

[2]   M. Green, "The verification of Silvaco as a solar cell simulation tool and the design and optimization of a four-junction solar cell," Master's Thesis, Naval Postgraduate School, 2002.

[3]   J. Lavery, "Quantum tunneling model of a p-n junction in Silvaco," Master's thesis, Naval Postgraduate School, 2008.

[4]   A. Bates, "Novel optimization techniques for mulitjunction solar cell design using Silvaco ATLAS," Master's thesis, Naval Postgraduate School, 2004.

[5]   J. Utsler, "Genetic algorithm based optimization of advanced solar cell designs modeled in Silvaco ATLAS," Master's thesis, Naval Postgraduate School, 2006.

[6]   M. Tsutagawa, "Triple junction InGaP/GaAs/Ge solar cell optimization using Silvaco ATLAS," Master's thesis, Naval Postgraduate School, 2008.

[7]   B. Canfield, "Advanced modeling of high temperature performance of indium gallium arsenide thermophotovoltaic cells," Master's thesis, Naval Postgraduate School, 2005.

[8]   A. Crespin, "Modeling the effects of 1 MeV electron radiation in GaAs solar cells with Silvaco," Master's thesis, Naval Postgraduate School, 2004.

[9]   R. F. Pierret, *Semiconductor Device Fundamentals*. Reading, MA: Addison-Wesley Publishing Company, 1996.

[10]   "Space Power and Radiation Effects," class notes for EC3230, Department of Electrical and Computer Engineering, Naval Postgraduate School, Spring 2009.

[11]   G. B. Stringfellow. "The importance of lattice mismatch in the growth of $Ga_xIn_{1-x}P$ epitaxial crystalls," J. Appl. Phys. 43, 8, pp. 3455–3460, August 1972.

[12]   T. Prutski, P. Diaz-Arencibia, R. A. Brito-Orta, A. Mintairov, T. Kosel, and J. Merz. "Luminescense anisotropy of InGaP layers grown by liquid phase epitaxy," J. Appl. Phys. 37, pp. 1563–1568, May 2004.

[13]   *ATLAS User's Manual: Device Simulation Software*. Santa Clara, CA: SILVACO International, 2009.

[14]   B. Davenport, "Advanced thermophotovoltaic cells modeling, optimized for use in radioisotope thermoelectric generators (RTGs) for Mars and deep space missions," Master's thesis, Naval Postgraduate School, 2004.

[15]   T. Agui, T. Takamoto, E. Ikeda, and H. Kurita, "High-efficient dual-junction InGaP/GaAs solar cells with improved tunnel interconnect," Indium Phosphide and Related Materials, 1998 International Conference on …, pp 203–206, 1998.

[16]   Emcore Photovoltaics, "EMCORE T1000 cell – triple-junction high-efficiency solar cells for terrestrial concentrated photovoltaic applications," Emcore datasheet, March 2006.

[17]   D. King, J. Kratochvil, and W. Boyson, "Temperature coefficients for PV modules and arrays: measurement methods, difficulties, and results," presented at the 26th IEEE Photovoltaic Specialists Conference, Anaheim, California, September 29–October 3, 1997.

[18]   M. Patel, *Spacecraft Power Systems*. Washington, D.C.: CRC Press, 2005, pp. 135-180.

[19]   G. Landis, D. Merritt, R. Raffaelle, and D. Scheiman, "High-temperature solar cell development." Available: http://gltrs.grc.nasa.gov/reports/2005/CP-2005-213431/27Landis.pdf. Accessed August, 19, 2010.

[20]   K. Nishioka, T. Takamoto, T. Agui, M. Kaneiwa, Y. Uraoka, and T. Fuyuki, "Evaluation of temperature characteristics of high-efficiency InGaP/InGaAs/Ge triple-junction solar cell under concentration," *Solar Energy Materials & Solar Cells,* vol 85, pp. 429-436, 2005.

[21]   D. Aiken, M. Stan, C. Murray, P. Sharps, J. Hills, and B. Clevenger. "Temperature dependent spectral response measurements for III-V multi-junction solar cells," presented at the 29th IEEE Photovoltaic Specialists Conference, New Orleans, Louisiana, May 20-24, 2002.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
    Ft. Belvoir, Virginia

2. Dudley Knox Library
    Naval Postgraduate School
    Monterey, California

3. Marine Corps Representative
    Naval Postgraduate School
    Monterey, California

4. Director, Training and Education,
    MCCDC, Code C46
    Quantico, Virginia

5. Director, Marine Corps Research Center
    MCCDC, Code C40RC
    Quantico, Virginia

6. Marine Corps Tactical Systems Support Activity
    (Attn: Operations Officer)
    Camp Pendleton, California

7. Dr. R. Clark Robertson, Chairman
    Department of Electrical and Computer Engineering
    Naval Postgraduate School
    Monterey, California

8. Dr. Sherif Michael
    Department of Electrical and Computer Engineering
    Naval Postgraduate School
    Monterey, California

9. Dr. Todd Weatherford
    Department of Electrical and Computer Engineering
    Naval Postgraduate School
    Monterey, California