2013 IEEE 7th International Power Engineering and Optimization Conference (PEOCO2013), Langkawi, Malaysia. 3-4 June 2013

Implementation of Field Programmable Gate Array based Maximum Power Point Tracking Controller of Photovoltaic System

Mukarram A.M. Al-Mohaya, Abd Kadir Mahamad and Sharifah Saon
Faculty of Electrical and Electronic Engineering
University Tun Hussein Onn Malaysia, Johor, Malaysia
kadir@uthm.edu.my, sharifa@uthm.edu.my, almohia82@yahoo.com

Abstract – Photovoltaic (PV) cell is one of the electrical part to convert photo-light into electricity in order to generate the electrical power. This paper presents the implementation of Field Programmable Gate Array (FPGA) based Maximum Power Point Tracking (MPPT) controller in evaluating the Maximum Power Point (MPP) output voltage of PV system. Matlab Simulink and Quartus II VHDL software tools are used as simulator of this FPGA, while Altera DE2 board is used as a controller. Both simulator and hardware had shown the same results, which prove that the designed system has been successfully extracting the MPP. The system has been evaluated in sunny day and partially shaded conditions to analyze the respective outputs.

***Keywords* –** PV, FPGA, MPPT, MPP

I. INTRODUCTION

Conventionally, the PV output voltage is inconstant and insufficient for direct uses. Thus, need of designing a controller which can calculate and extract the MPP at any instant from the solar cells is important. There are a few numbers of proposed control strategies in the literature: which are perturb and observe, fuzzy logic controller (FLC), artificial intelligence techniques, constant voltage, parasitic capacitance and etc.

Perturb and observe generally guarantee of acceptable performances and easy to implement MPPT. Whereas FLC provides a mean of converting a linguistic control strategy based on expert knowledge into an automatic strategy [1].

FPGA based systems provide a number of run-time advantages over the sequential machines such as a microcontroller. Moreover, concurrent operations can be operated continuously and simultaneously faster than Digital Signal Processing device (DSP). Since, the functions of components can be integrated onto the small chip, FPGAs offer lower production cost than DSPs that can perform only DSP-related computations. Beside robustness, the capability to reprogram can provide a high degree of flexibility, which allows MPPT control system to be easily updated or modified even when it is running [2].

II. METHODOLOGY

There are several things should be considered in development of this project. Methodology is a process that has to be followed during analyzes data and designing a project. The process include the method, technique, and the tools/equipments.

A. PV Cell Model

The simple PV cell model is consists of parallels diode and current source. Current source is directly proportional to the solar radiation and diode represents PN junction of a solar cell. Therefore, ideal solar cell, $I = I_{PH} - I_D$ can be represented as in Eqn. (1).

$$I = I_{PH} - I_s\left[\exp\left(\frac{qV}{kTcA}\right) - 1\right] \tag{1}$$

where I_{PH} is photo current, I_D is diode current , q is the elementary charge $1.6\text{x}10^{-19}$ Coulombs, k is a constant of $1.38\text{x}10^{-23}$J/K, Tc is the cell temperature in Kelvin, A is the idea factor and dependent on PV technology, and V is the measured cell voltage that is either produced (power quadrant) or applied (voltage bias).

Furthermore, a more accurate model will involved two diode terms, however, in this project just to concentrate on a single diode model. By expanding the PV cell, as shown in Figure 1, the Eqn. (1) can be expanded as:

$$I = I_{PH} - I_s\left[\exp\left(\frac{q(V + IR_S)}{kTcA}\right) - 1\right] - \left(\frac{V + IR_S}{R_{SH}}\right) \tag{2}$$

where R_S and R_{SH} are series and shunt resistances, respectively.

During operation, the efficiency of solar cells is reduced by the dissipation of power across internal resistances. On the other hand, for an ideal cell, R_{SH} would be approach infinity. Thus, Eqn. (2) can be represented as:

$$I = I_{PH} - I_s\left[\exp\left(\frac{q(V + IR_S)}{kTcA}\right) - 1\right] \tag{3}$$

Besides, a small variation in R_S will significantly affect the PV output power. The appropriate PV solar cell model with suitable complexity is shown in Figure 2.

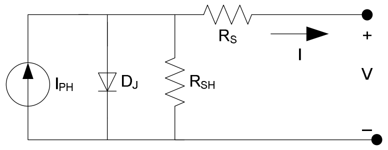

Figure 1: General Model of PV Cell

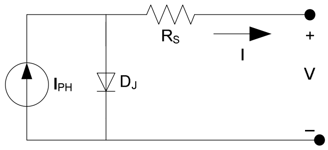

Figure 2: Equivalent Circuit Modules of PV System

In this paper the SX -150 is been used as photovoltaic module. The typical electrical characteristics of SX-150 PV modules are shown as in Table 1.

Table 1: SX-150 PV characteristic

Electric parameter	BPSX150
Maximum power, P_{max}	150 *W*
Maximum current (short circuit), I_{mp}	4.35 *A*
Maximum voltage (open circuit), V_{mp}	34.5 *V*
Short circuit current, I_{SC}	4.75 *A*
Open circuit voltage, V_{SC}	43.5 *V*
Temp. coefficient: short-circuit current	(0.065 ± 0.015)%/°C
Temp. coefficient: open-circuit voltage	–(160 ± 20)mV/°C

B.MPPT Algorithms

The MPPT algorithms are designed to dynamically extract the MPP from the PV panels. Figure 3 shows the flow chart of the operation of MPPT controller using perturb and observation (P&O) method.

978-1-4673-5074-7/13/$31.00 ©2013 IEEE

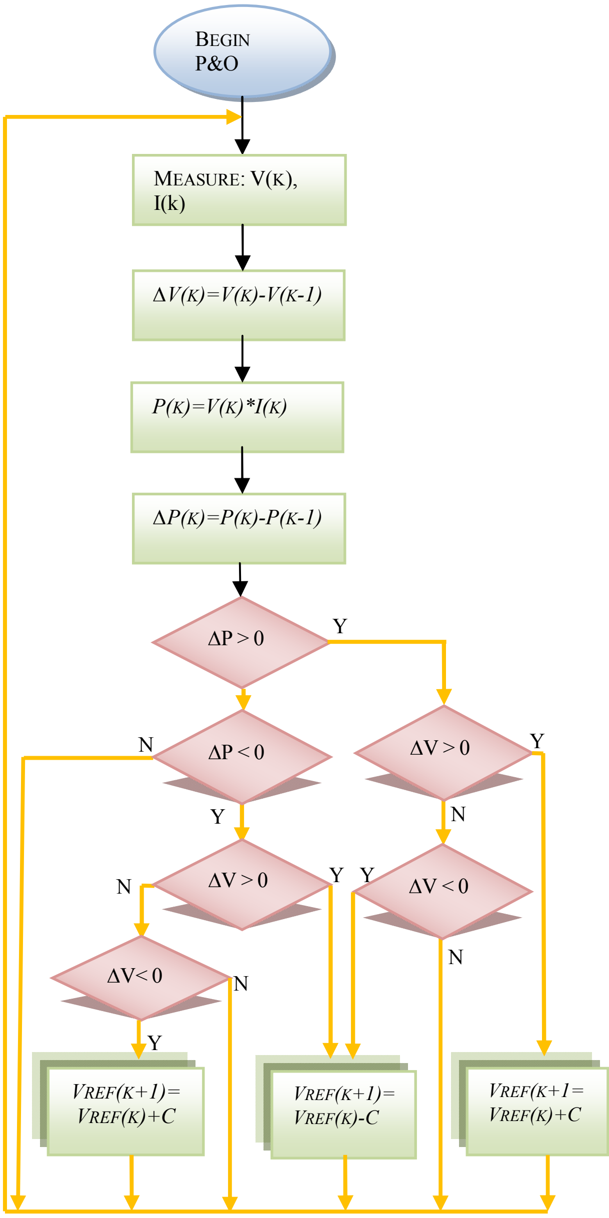

Figure 3: P&O MPPT Operation Flow Chart

1 .MPPT with Fuzzy Logic Controller

In order to increase the performance of the P&O method, a FLC is used to dynamically modify the step size, based on the changes in power and reference voltage of the solar panel. The aim of the FLC is to let the P&O method reach MPPT faster.

There are four basic elements needed in order to design FLC, namely fuzzification, rule base, inference engine and defuzzification. Fuzzification is used to convert the solar panel parameters into fuzzy sets using a pre-determined fuzzy membership function. The membership function is the curve that decides the membership value of each range of input signal values. The fuzzy rule base is a collection of if-then rules which controls what the output should be given fuzzy values of input. They are set according to experience and operation of the system to be controlled.

The fuzzy inference engine is used to generate an output based on the given inputs and fuzzy rule base. The fuzzy inference engine will generate a logical decision based on the fuzzy rule setting. The defuzzifier is then used to convert the fuzzy output back into an actual value. In this project, the difference in power (delta P) and difference in current (delta I) are used as an input to the fuzzy controller, while the output will be the change in voltage reference (delta V_{ref}) to be sent to the solar panel.

2. P&O MPPT using VHDL

The architecture of the MPPT controller consists of a structure that facilitates efficient data flow from starting point. The library declaration contains list of all libraries to be used in the design. It functions for resolving and translating the language within the body of the program. The most common library used in programming is IEEE library because it can specify a multi-level logic system. The entity declaration defines the inputs and outputs of the design. It lists a specification of all input and output pins of the circuit [3].

The entity declaration provides an external view of a component but does not provide information on how a component is implemented. It specifies the number of ports, direction of the ports, and types of ports.

The architecture body defines the relationship between the inputs and outputs of a design entity which expressed in terms of behavioural style, data-flow style and structural style. This part will describe the behaviour of the ENTITY that had already declared. It consists of a declaration section where components, types, signals, constants, and subprograms are declared, then followed by a collection of concurrent statements [4].

The P&O algorithm is resides on the MPPT controller. This controller in first state receives values for voltage and current, In second state, its calculate the differentials in voltage, current and power. In third state its check all requirements of parameter and flags changes, forth state is to check if MPPT could be applied and which direction increase or decrease of V_{ref}. In fifth state, its perform bounds checking on new V_{ref} and difference between the voltage and V_{ref} system exceeds to a certain threshold V_{ref} min. In the sixth state, its perform low current comparing with I_{min} which enables the direction of MPPT, The seventh state is to update V_{ref} and save the values for next iteration. After calculating all the necessary differentials in six states, its send updates to V_{ref} that ready out has been performed in last state. The P&O algorithm stores the power, voltage, and current values from the previous iteration in each state. Figure 4 summarize overall process of P&O MPPT in form of finite state machine (FSM).

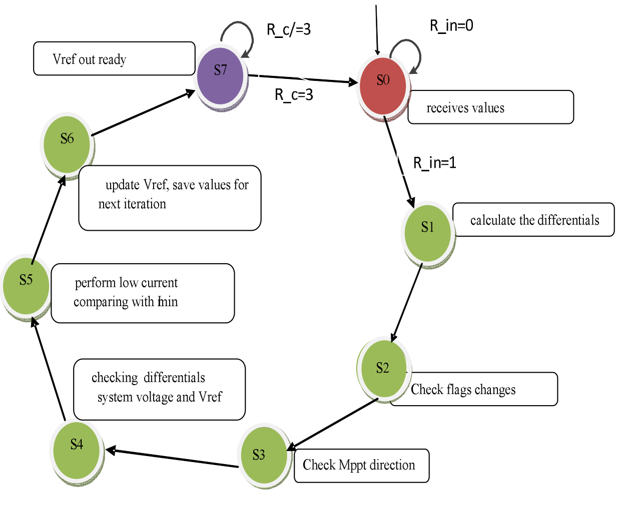

Figure 4: FSM of P&O MPPT

3. P&O MPPT using Quartus II

After the design process using VHDL is done, the simulation part of the waveform will be simulate using Vector Waveform File (.vwf). By using the waveform editor, the .vwf can be easily viewed and edited.

III. RESULTS AND DISCUSSIONS

First section is about the rated values of PV module which has been used in this study. These include the temperature and irradiance to obtain a certain value of voltage, current and power. The graphical representations are taking into account to support the respective findings.

Second section describes the assessment of the variation in the values of irradiance and temperature. These inputs are assumed as variables to mimic a full one day conditions. Meanwhile, the output results after the implementation of MPPT controller is shown later. A comparison has also been made to evaluate the output of MATLAB simulation as compared to the QUARTUS II simulation.

A. The Input graph for PV module by Matlab Simulink in the Normal Condition with variable input

Figure 5 and 6 show the graph for input variable of PV module which are irradiance and cell temperature. As been shown, the red lines of the graph are representing sunny day. While, the purple lines of the graph is presenting partially shaded condition. During the simulation, only one switch can be turned on in one time.

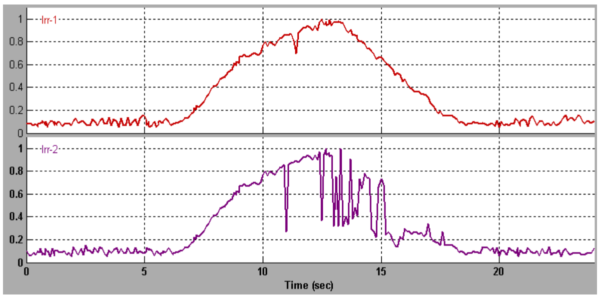

Figure 5 : Input Irradiance for PV Module

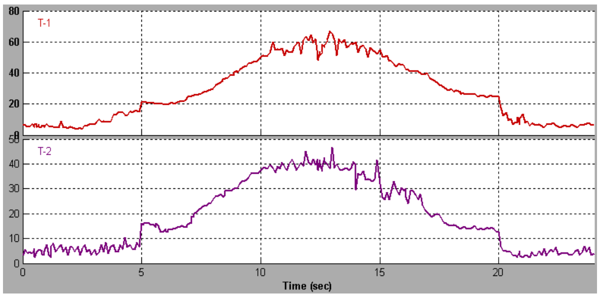

Figure 6 : Input Cell Temperature for PV Module

B . The MPPT controller module by Quartus II Simultion

Figure 7 shows the simulation of MPPT controller based on P&O algorithm as implemented in Quartus II. It can be found from this simulated output that the didv_test is flag for variation of current and voltage at the same time. While the graph for the dp_test shows the variation in the power output. The graph for dv_test presents the change in voltage output from PV module. It can be observed from the simulated output of the mppt_dir graph show the 1 and 0 values which represent the direction of the MPPT. If the output is 1, that means V_{ref} is increased while if the output is 0, it shows the decrement in the V_{ref} value. Now, the ovr flag plays the role of short circuit current protection and open circuit voltage protection. If any of these values try to overshoot from the maximum values of optimal PV module, the value of ovr flag becomes 1 to signal the MPPT controller in order to decrease the V_{ref} values, accordingly. Skip_mppt graph depicts that the MPPT controller is skipped in the areas where the derivative of power is equal to zero. Ready_in and ready_out shows that the values from PV are ready for input in to MPPT controller and output from MPPT, respectively.

C. Comparison results of MPPT between Matlab simulink and Quartus II

Figure 8, 9 and 10, show the comparison of the simulated current, reference power and reference voltage output from the MATLAB Simulink and Quartus II simulation, respectively. These graphs show the current variation for 24 hours. The values in the graph have been scaled to a ratio of 1:331 for 1 second. It can be observed from the graphs that both of these simulations when fed with the same input values produce almost the same output after simulation.

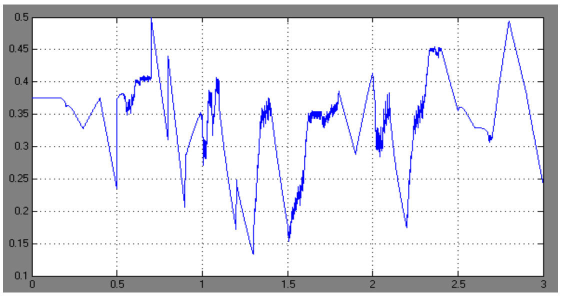

Figure 8(a): MPPT Input Current from Matlab Simulink

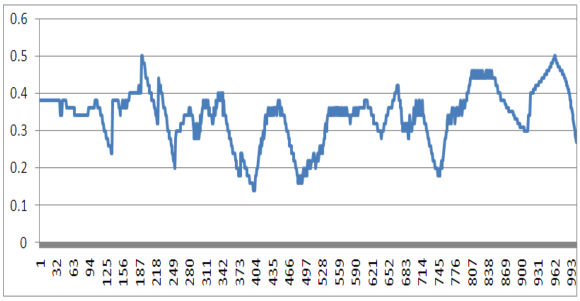

Figure 8(b): MPPT Input Current from Quartus II

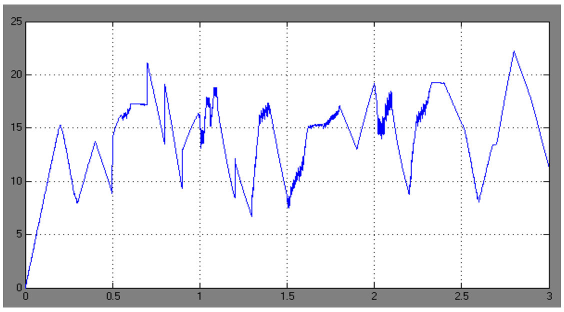

Figure 9(a): MPPT Output power from Matlab Simulink

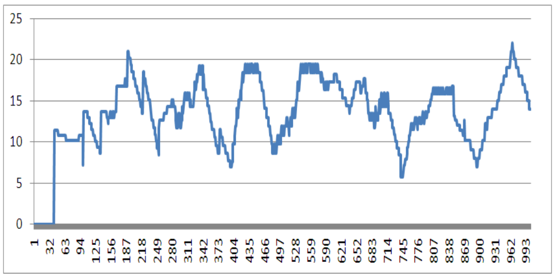

Figure 9(b): MPPT Output power from simulation Quartus II

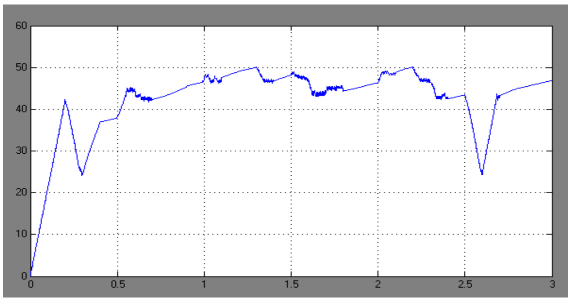

Figure 10(a): MPPT Output Voltage reference from Matlab Simulink

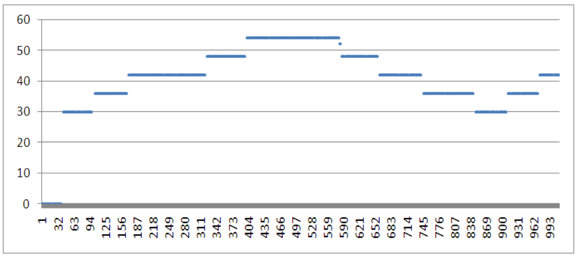

Figure 10(b): MPPT Output Voltage reference from Simulation Quartus II

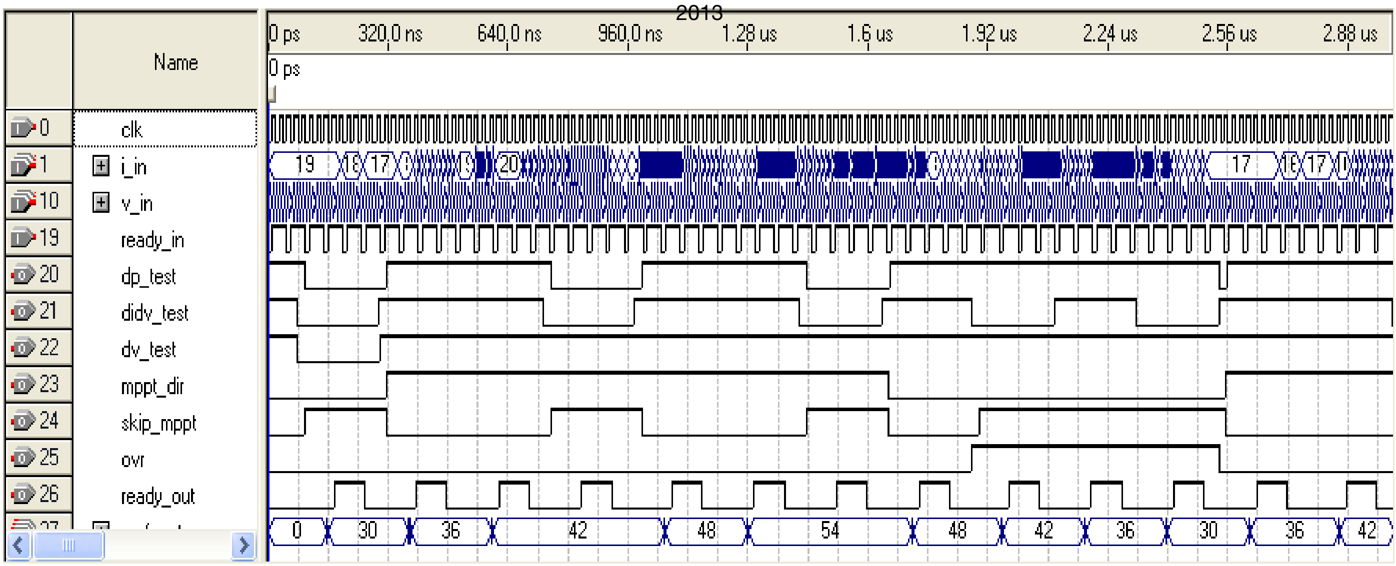

Figure 7: Simulation results of MPPT controller by simulation using Quartus II

IV. CONCLUSIONS

This paper has presented the modelling of PV module and the development of the MPPT controller. In particular, the performances of the controllers are analyzed for variable temperature and irradiation. The proposed system is simulated by using MATLAB Simulink. Based on the MATLAB Simulink and Quartus II simulation result, the work has successfully achieved the objective of this paper..

This study proposes an FPGA-based implementation of a MPPT controller. The P&O algorithm is used to track the MPP. The algorithm runs on an Altera DE2 FPGA board. Integrating FPGA in MPPT control system provides numerous advantages. To meet performance requirements, FPGAs are desirable since their performance can easily surpass the performance of microcontrollers and DSPs. In addition, given their re-programmability, FPGAs can be used to conduct in-circuit experimentation, testing and optimization of various parameters that affect the performance of the MPPT control system [5][6].

The system comprising of PV module and the MPPT controller has been implemented in this study to propose and validate an efficient MPPT system based on P&O algorithm. This system is being simulated using MATLAB SIMULINK to get an idea of the working of this system with different input values under various conditions, including sunny day and partially shaded conditions. After the clarification and validation of the proposed setup by using MATLAB Simulink, the system is then implemented by using QUARTUS II simulation to extract the same output values as in MATLAB[7][8].

The scheme proved to work effectively and the output from the QUARTUS II simulation is observed to be approximately same as in MATLAB.

This paper shows how FPGAs can be used to build cost-effective and highly adaptable implementations of MPPT control systems; for future research in FPGA implementations of MPPT control systems.

Hence, based on the simulation results and the respective analysis and findings, it can be concluded that the aim and the objectives of this paper are successfully achieved.

REFERENCES

[1] Lee, C. C. (1990). Fuzzy-logic in control-systems: Fuzzy logic controller, Part I. IEEE Trans Syst Man Cybern, 20(2), 404-418.

[2] Messai, A., et al. "Maximum power point tracking using a GA optimized fuzzy logic controller and its FPGA mplementation." Solar energy 85.2 (2011): 265-277.

[3] Volnei A. Pedroni. "Circuit Design With VHDL". United State of America, Massachusetts Institute Technology, 2004.

[4] TJUKUP MARNOTO, Tjukup, et al. "Mathematical model for determining the performance characteristics of multi-crystalline photovoltaic modules." Proc. of the 9th WSEAS Int. Conf. on Mathematical and Computational Methods in Science and Engineering. WSEAS Conferences, Trinidad and Tobago, 2007.

[5] Pongsakor Takun, Somyot Kaitwanidvilai and Chaiyan Jettanasen, "Maximum Power Point Tracking using Fuzzy Logic Control for Photovoltaic Systems"

[6] Koutroulis, Eftichios, Kostas Kalaitzakis, and Vasileios Tzitzilonis. "Development of an FPGA-based system for real-time simulation of photovoltaic modules." Microelectronics Journal 40.7 (2009): 1094-1102.

[7] B. Amrouche, M. Belhamel and A. Guessoum, "Artificial Intelligence Based P&O MPPT Method for Photovoltaic Systems" Revue des Engergies Renouvelables ICRESD-07 Telmcen (2007) 11 -16.

[8] Giraud, Francois, and Ziyad M. Salameh. "Analysis of the effects of a passing cloud on a grid-interactive photovoltaic system with battery storage using neural networks." Energy Conversion, IEEE Transactions on 14.4 (1999): 1572-1577.