CHAPTER III

RESEARCH METHODOLOGY

This chapter describes three parts, the research instrument, calculation of sun's position and fine-tuning of solar tracking using genetic algorithms.

1. The Research Instrument

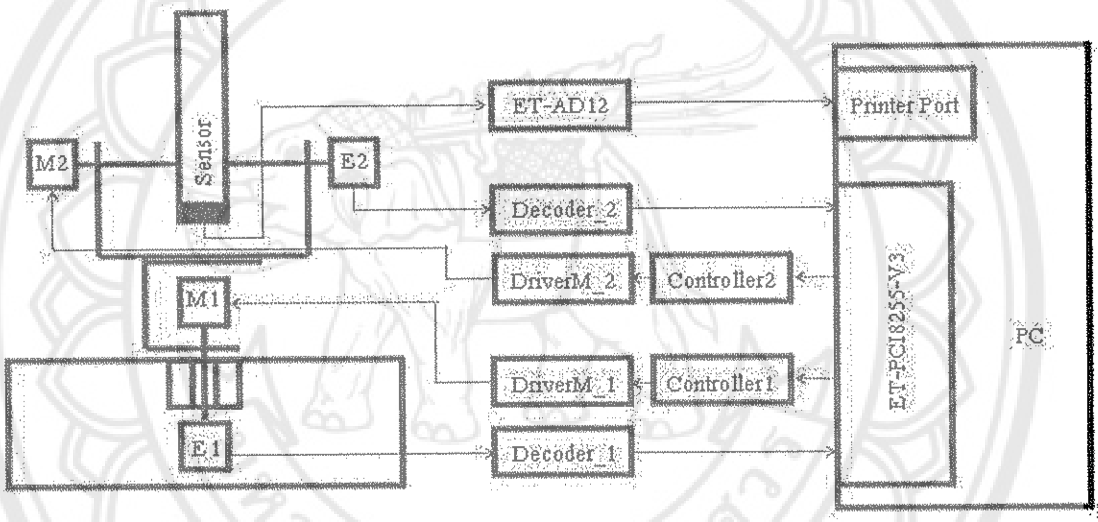

Figure 9 Solar Tracking Machine Model (STMM) Diagram

Process of Solar Tracking Machine Model (STMM)

Solar tracking in this research is altitude over azimuth axis type which has 2 basic axle-turns. Each axle uses DC motors (M1, M2) to driving and they are controlled speed and rotate direction by driver motor (DriverM_1, DriverM_2).

The controllers (Controller1, Controller2) send Pulse Width Modulation (PWM) signal to the driver motor for control speed and send direction-signal to control rotate direction of motors which the controllers received commands from program in PC via interface card (ET-PCI8255-V3).

Decoder and Counter (Decoder_1, Decoder_2) will decrypt signal from optical encoder (E1, E2) into current degree-position of the axle turns then report to PC via interface card (ET-PCI8255-V3).

While STMM is working, the program will order turn to go to still each position, and then it measures voltage from sensor which connect to A/D Converter (ET-AD12), for convert signal from analog to digital, 12 bits, finally send the data into program via printer port of PC.

1.1 Solar Tracking Machine Model (STMM)

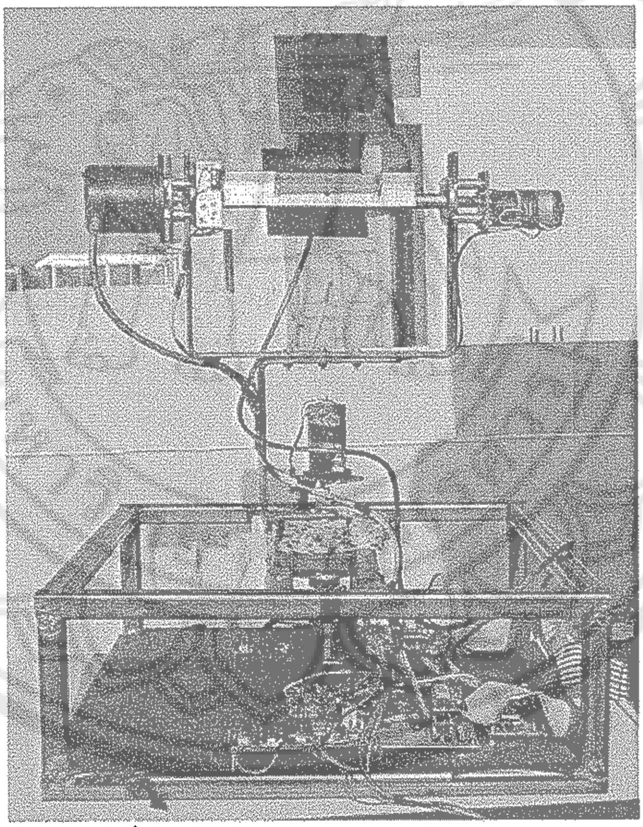

Figure 10 Solar Tracking Machine Model (STMM)

Specification of Solar Tracking Machine Model (STMM)

Mass	:	15 kg
Physical dimensions	:	500 x 550 x 500 mm
Configuration	:	Altitude over azimuth axis
Motor	:	Brushless DC Motor on each axis
Gear	:	Maxon gear ratio 57:1
Position encoder	:	Increment encoder 2000 pulses per 360° for azimuth axis and 2500 pulses per 360° for altitude axis.
Resolution	:	8000 cnts/360° corresponding to a resolution of 0.045° for azimuth axis and 10000 cnts/360° corresponding to a resolution of 0.036° for altitude axis.

1.2 ET-AD12

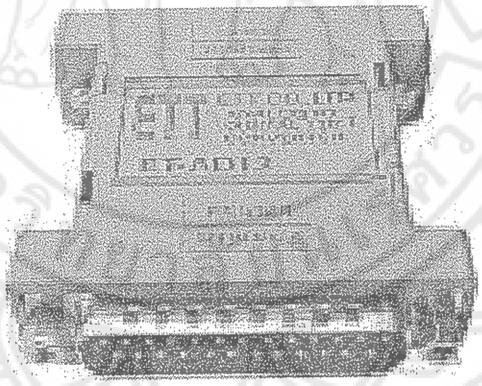

Figure 11 ET-AD12

ET-AD12 is a board for convert analog signal to digital signal with resolution 12 bits. ET–AD12 is designing for comfortable with small size and use to join with personal computer by parallel with printer port that make to easily. Develop program on ET-AD12 can improve thought personal computer as well many computer language for example C, C++, Pascal, Visual Basic etc.

Specification of ET-AD12

Conversion time	:	60 µs
Sampling rate	:	11.1 KHz
ADC/Channel	:	2 channels
Gain error	:	+/-2 LSB
Analog input range	:	-0.05V to +5.05V
+Vcc supply	:	+9Vdc

1.3 ET-PCI8255-V3

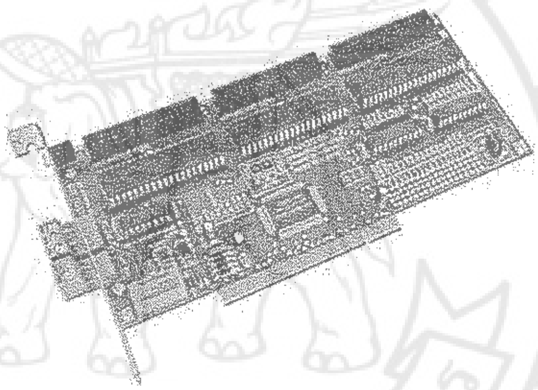

Figure 12 ET-PCI8255-V3 Card

ET-PCI8255-V3 card is programmable input / output card type TTL Logic with 72 bit size. It was improved and developed to be connectable with personal computer by PCI Bus system through PCI SLOT. Ett teams are selected to succeed IC Chip that was improved and developed for connection to particularly PCI Bus system. TIGER320'Chip from Tiger Jet Network Inc is selected that to be perfectly connectable to PCI bus V2.2 and tested and accepted from PCI-SIG test and approve institute so show to sureness performance of TIGER320'Chip with PCI Bus system.

1.4 DriverM_1 and DriverM_2

In this research used two ET-OPTO DC Motor Drivers for drive each motor M1 and M2.

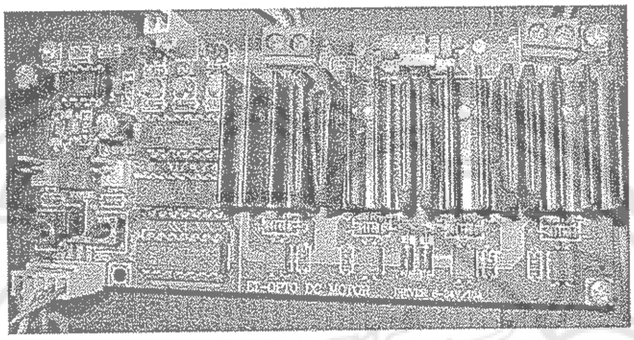

Figure 13 ET-OPTO DC Motor Driver

Specification of ET-OPTO DC Motor Driver

1. MOSFET is the drive can make a force to drive DC MOTOR for 6V -24V with electric current at 5 Amp.

2. Opto is the separations groundless between the electronic sources to 2 parts for protect equipment on the broad.

3. Separate the electronic sources to 2 parts. First, the electronic source with DC 5 V for IC, And Second, the electronic source with DC 6 – 24 V to the DC MOTOR.

4. Change the rotate the DC MOTOR form SW LEFT, SW RIGHT on the board

5. This board has the design to stop the rotation of DC MORTOR with 2 types. They are fast stop type and slow stop type.

6. Container control the velocity of DC MOTOR by PWM signal form board directly or outside.

7. Include 5 roasts to join the Microcontroller for send signal form outside by control the direction and speed of the DC MOTOR.

1.5 Controller1 and Controller2

In this research used two ET-ISP89S52 controllers for control the DriverM_1 and DriverM_2.

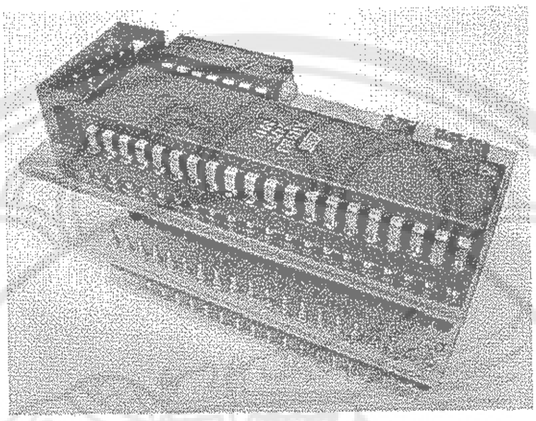

Figure 14 ET-ISP89S52

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8K bytes of in-system programmable Flash memory. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the indus-try-standard 80C51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory pro-grammer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S52 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications. The AT89S52 provides the following standard features: 8K bytes of Flash, 256 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset.

Specification of ET-ISP89S52

1. Using CPU Atmel AT89S52
2. Compatible with MCS®-51 Products
3. 8K Bytes of In-System Programmable (ISP) Flash Memory – Endurance: 1000 Write/Erase Cycles
4. 4.0V to 5.5V Operating Range
5. Fully Static Operation: 11.059 MHz
6. Three-level Program Memory Lock
7. 256 x 8-bit Internal RAM
8. 32 Programmable I/O Lines
9. Three 16-bit Timer/Counters
10. Eight Interrupt Sources
11. Full Duplex UART Serial Channel
12. Low-power Idle and Power-down Modes
13. Interrupt Recovery from Power-down Mode

1.6 Sensor

Sensor in this research made of poly crystalline solar cell. It changes solar energy transform to electricity voltage.

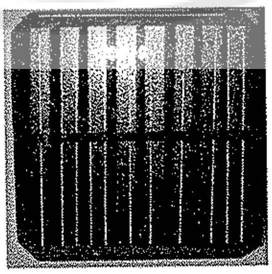

Figure 15 Poly Crystalline Solar Cell

Specification of solar cell

1.Power Max	= 0.32 Wp.
2.Open Circuit Voltage (Voc)	= 6.5V
3.Maximum Voltage (Vmp)	= 5.1V
4.Maximum Circuit (Imp)	= 62mA
5.Short Circuit Current (Isc)	= 92mA
6.Module size(mm)	= 60x60x3mm
7.Weight	= 0.03kg

The factors that affect to the solar cell efficiency are light intensity and temperature. The electric current is directly depending on light intensity, but the electric current no take effect from temperature changes. And the voltage will be decrease when temperature increases.

A solar cell is a device that converts light energy into electrical energy. Sometimes solar cell is reserved for devices intended specifically to capture energy from sunlight.

1.7 Square Pipe

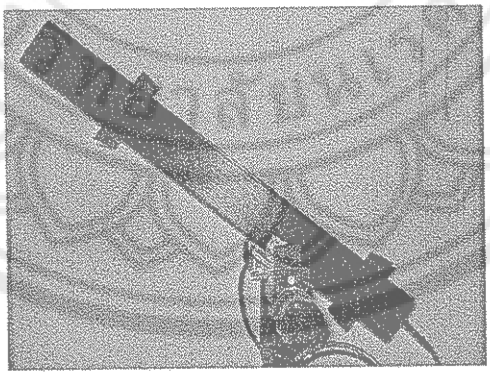

Figure 16 Square Pipe

Solar cell was to be used to assess the tracking. The solar cell is installed in 60*60*600 millimeter square pipe at length 340 millimeter of the pipe for assign upper bound and lower bound each axis equal ±10 degrees see equation (10).

$$X = \frac{60}{\tan 10°} \qquad \text{(millimeter)} \qquad (10)$$

where X is length of the position of solar cell

The pipe is reducing the size of the image of the sun equal to the figure of the pipe. The pipe and solar cell assembly is fixed to the system at an angle perpendicular to the concentrator plane. The pipe function is to allow only the direct parallel rays to be exposed on the screen of the sensors .The voltage output from sensor be maximum if sensor stay in perpendicular position accurately with the sun.

1.8 Personal Computer

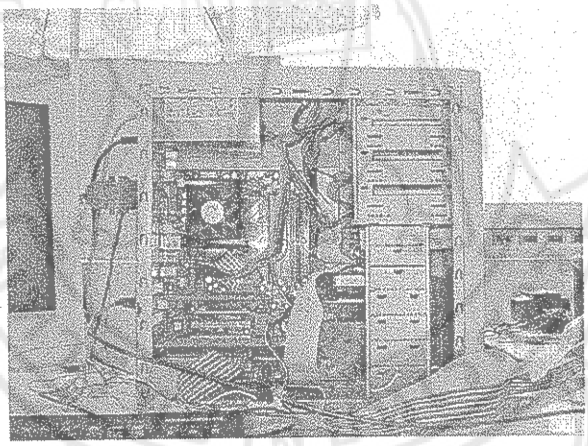

Figure 17 Personal Computer

Specification of Personal Computer

CPU	:	Intel Pentium IV 2.4 GHz
Mother board	:	Asrock P4VBB
RAM	:	DDR 256 MB PC3200
VGA	:	GF MX4000 128 MB
Hard disk	:	40 GB 7200 RPM
OS	:	Microsoft Window XP Professional SP1
Software	:	Visual basic and Minitab

2. Calculation of Sun's Position

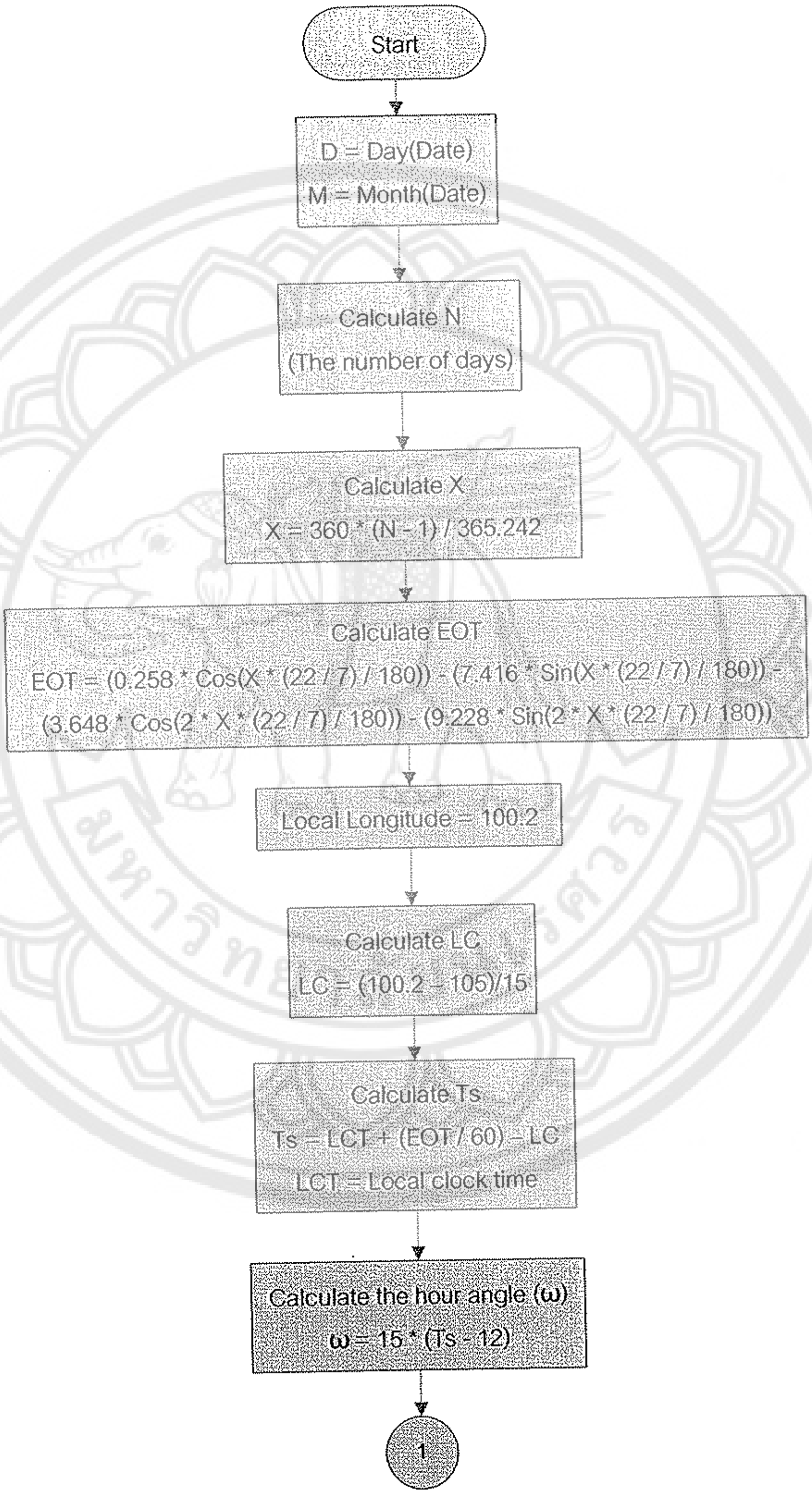

Figure 18 Flow Chart for Calculate Sun's Position

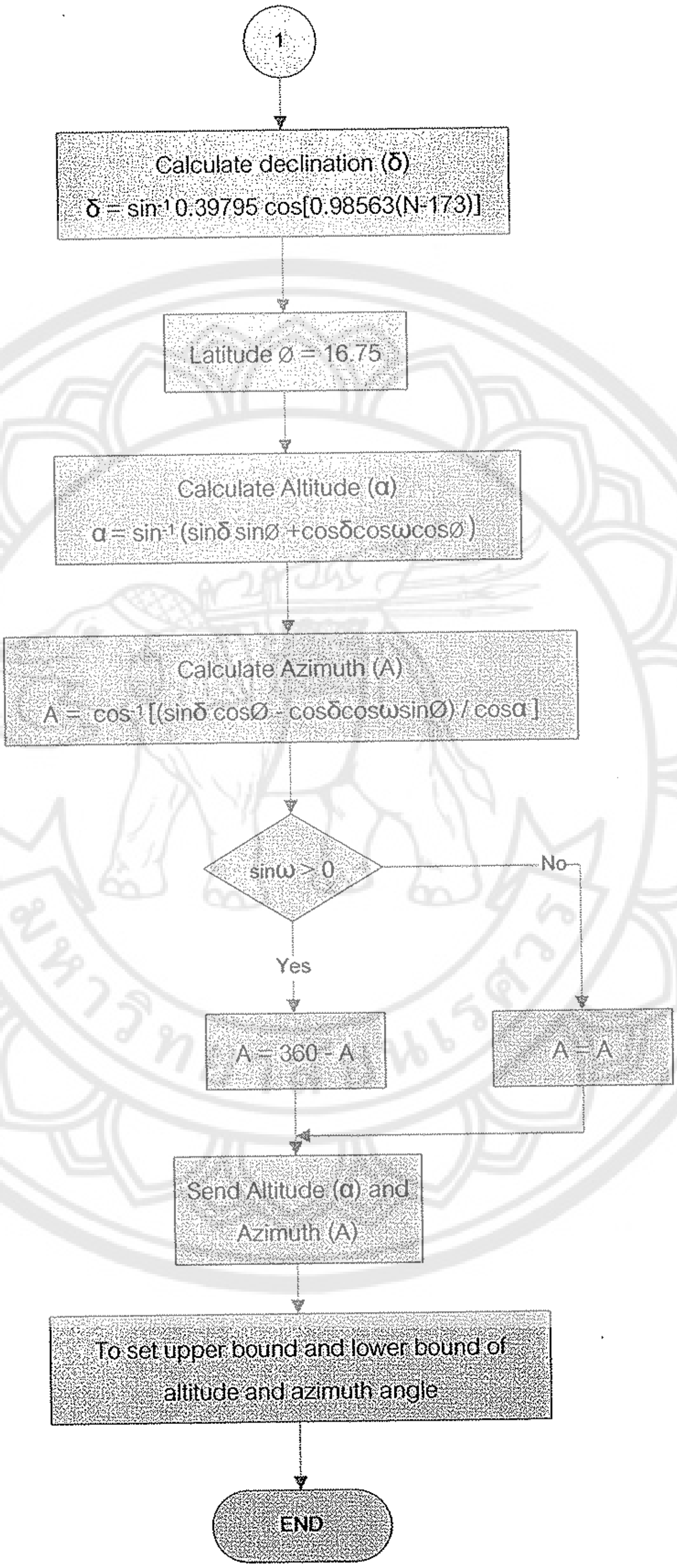

Figure 18 (Cont.)

From figure 18 to calculate sun's position by a formula that uses equation of time by Woolf (1968). Initial with assign date and month thereafter calculate the order day (N) follows the table 1.

Suppose calculate date 20 February, will get the result N equal to 51 then calculate, function of the day number (x), from equations 3rd as follows

$$x = \frac{360(51-1)}{365.242} \quad \text{(degrees)}$$

Thus, $x = 49.2824^{\circ}$

Then calculate Equation of Time (EOT) from equations 2nd as follows.

EOT = 0.258 cos(49.2824) – 7.416 sin(49.2824) – 3.648 cos2(49.2824) – 9.228 sin 2(49.2824)

= 0.1683 - 5.6208 – (-0.5433) – 9.1251

= -14.0343 (minutes)

This experimentation has done at faculty of engineering in Naresuan University, Phitsanulok. It is on Longitude 100.2° and Latitude = 16.75°. The equation 5th can calculate the Longitude Correction (LC) value, as follow.

$$LC = (100.2 - 105) / 15 = -0.32 \quad \text{(hours)}$$

Form equation 4th can calculate Solar time (t_s) by assume calculate at 11.35.12 o'clock, as follow.

$$t_s = 11.5867 + (-14.0343/60) - (-0.32) = 11.6728 \quad \text{(hours)}$$

So, calculate hour angle (ω) form equation 1st.

ω = 15 * (11.6728-12) = -4.908 (degrees)

From that time do seeking Declination angle (δ) from equation 6th, as follow.

$$\delta = \sin^{-1} 0.39795 \cos[0.98563(51-173)]$$
$$= \sin^{-1}(-0.2)$$
$$= -11.5370 \quad \text{(degrees)}$$

From position experiment location has Latitude (Φ) = 16.75°

then equation 8th can calculate to seek the value Altitude angle (α) , as follows.

$$\alpha = \sin^{-1}[\sin(-11.5370)\sin(16.75) + \cos(-11.5370)\cos(-4.908)\cos(16.75)]$$
$$= \sin^{-1}[(-0.2) * (0.2882) + (0.9798) * (0.9963) * (0.9576)]$$
$$= \sin^{-1}[0.8771]$$
$$= 61.2945 \quad \text{(degrees)}$$

And do the calculations seek the value Azimuth angle (A), from equation 9th, as follows.

$$A = \cos^{-1}\frac{[\sin(-11.5370)\cos(16.75)\ \ \cos(-11.5370)\cos(-4.908)\sin(16.75)]}{\cos(61.2945)}$$

$$= \cos^{-1}\frac{[(-0.2)*(0.9576) - (0.9798)*(0.9963)*(0.2882)]}{(0.4803)}$$

$$= \cos^{-1}(-0.9845) \quad = 169.8990 \quad \text{(degrees)}$$

$\sin\omega$ = sin(-4.908) = -0.0856

Thus $\sin\omega < 0$

So, A = 169.8990 (degrees)

Thus form calculate get

Altitude angle (α) = 61.2945 (degrees)

Azimuth angle (A) = 169.8990 (degrees)

In this experiment will guide the Altitude angle value (α) and Azimuth angle value (A) to go use in limits seeking specification by fix the value at ±10 degrees (see equation 10), from the value that can calculate.

Lower bound of Altitude angle (A_{j2}) = 51.2945 (degrees)

Upper bound of Altitude angle (B_{j2}) = 71.2945 (degrees)

Lower bound of Azimuth angle (A_{j1}) = 159.8990 (degrees)

Upper bound of Azimuth angle (B_{j1}) = 179.8990 (degrees)

3. Fine-tuning of Solar Tracking using Genetic Algorithms

First, we need to encode decision variables into binary strings. The length of the string depends on the required precision in this work.

This experiment is assigned fine-tuning precinct with ±10 form azimuth angle value and altitude angle value. These values are calculating form the equation (10). So, GA has the range for find the position to receive maximum of sunlight in other axis with 20 degree. Azimuth axis contain the resolution at 0.045 degree it can explain position at 20/0.045 about 445 point in 20 degree that uses 9 bits binary string. Altitude axis have the resolution at 0.036 degree it can explain position at 20/0.036 about 667 point in 20 degree that uses 10 bits binary string. Also throughout experiment must use at 19 bits.

m1 = 9 and m2 = 10

Where, m is the number of bits.

The total of length of a chromosome is 19 bits which can represented as follows:

← 19 bits →

V_j = 001110000 1000110101

← 9 bits → ← 10 bits →

Azimuth (X_1) Altitude(X_2)

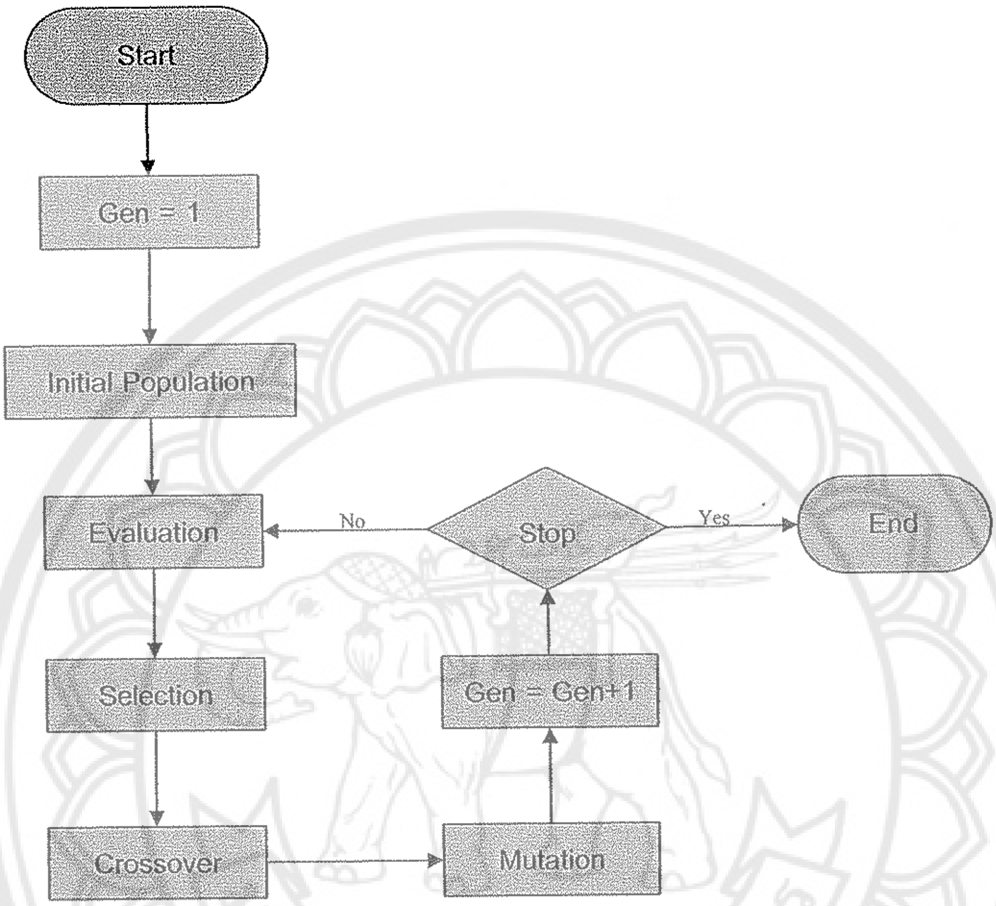

Figure 19 Flow Chart for Solar Tracking using Genetic Algorithms

Suppose calculate the value Azimuth angle = 150 degrees and Altitude angle = 60 degrees thus, Lower limits and upper limits of Azimuth angle are:

A_j1 (Lower bound of Azimuth angle)= 140 degrees

B_j1 (Upper bound of Azimuth angle)= 160 degrees

Lower limits and upper limits of Altitude angle are:

A_j2 (Lower bound of Altitude angle) = 50 degrees

B_j2 (Upper bound of Altitude angle) = 70 degrees

The corresponding values for variables X_1 and X_2 are given below:

	Binary Number	Decimal Number
X_1	001110000	112
X_2	1000110101	565

The mapping from a binary string to a real number for variable X_j is straight forward and completed as follows:

$$X_j = A_j + \text{decimal}(\text{substring}_j)^* (B_j - A_j)/2^{mj}-1 \quad (11)$$

where decimal (substring$_j$) represents the decimal value of substring-j for decision variable X_j.

thus,

$X_1 = 140 + 112 * (160\text{-}140)/ 2^9 -1 \quad = 144.38 \quad$ degrees

$X_2 = 50 + 565 * (70\text{-}50)/ 2^{10} -1 \quad = 61.05 \quad$ degrees

From figure 19 Flow chart for solar tracking using genetic algorithms is described following.

3.1 Initial population

Initial population is randomly generated as follows:

V_1 = [1110001100111100100]

V_2 = [0100111000011110101]

V_3 = [0001001111000010101]

V_4 = [0010111001111000111]

V_5 = [1100010011110001011]

V_6 = [0110001110000011000]

V_7 = [0101110001111100010]

V_8 = [1011000111010100011]

V_9 = [0011110001010111100]

V_{10} = [1110100011100001110]

3.2 Evaluation

The process of evaluating the fitness of a chromosome consists of the following three steps:

Step 1. Convert the chromosome's genetype to phenotype. Here, this means converting binary string into relative real values $X^k = (X^k_1, X^k_2)$, $k = 1,2,\ldots,pop\_size$.

So, (X^k_1, X^k_2) = (Azimuth angle^k, Altitude angle^k).

Binary Number	Decimal	Number
V_1 = [1110001100111100100]	454	484
V_2 = [0100111000011110101]	156	245
V_3 = [0001001111000010101]	39	533
V_4 = [0010111001111000111]	92	967
V_5 = [1100010011110001011]	393	907
V_6 = [0110001110000011000]	199	24
V_7 = [0101110001111100010]	184	994
V_8 = [1011000111010100011]	355	675
V_9 = [0011110001010111100]	120	700
V_{10} = [1110100011100001110]	465	782

The corresponding decimal values are

$V_1 = [X_1,X_2] = [157.78 , 59.46]$

$V_2 = [X_1,X_2] = [146.11 , 54.79]$

$V_3 = [X_1,X_2] = [141.53 , 60.42]$

$V_4 = [X_1,X_2] = [143.60 , 68.90]$

$V_5 = [X_1,X_2] = [155.38 , 67.73]$

$V_6 = [X_1,X_2] = [147.79 , 50.47]$

$V_7 = [X_1,X_2] = [147.20 , 69.43]$

$V_8 = [X_1,X_2]= [153.89 , 63.20]$

$V_9= [X_1,X_2] = [144.70 , 63.69]$

$V_{10} = [X_1,X_2] = [158.20 , 65.29]$

Step 2. Evaluate the objective function $f(X^k)$. In this work evaluate the objective function by measure the voltage from solar sensor through ET-AD12.

Step 3. Convert the value of objective function into fitness. For the maximization problem, the fitness is simply equal to the value of objective function.

$$eval(V_k) = f(X^k),\ k = 1,2,\ldots,pop\_size. \tag{12}$$

An evaluation function plays the role of the environment, and it rates chromosomes in terms of their fitness.

For the clearness in value calculation most then has led the value that will have minuses 2700 (estimate minimum value of ET-AD12 in experimental).

The fitness function values of above chromosomes are as follows:

$eval(V_1)$ = $f(157.78, 59.46)$ = 3012 – 2700 = 312

$eval(V_2)$ = $f(146.11, 54.79)$ = 3108 – 2700 = 408

$eval(V_3)$ = $f(141.53, 60.42)$ = 3095 – 2700 = 395

$eval(V_4)$ = $f(143.60, 68.90)$ = 2902 – 2700 = 202

$eval(V_5)$ = $f(155.38, 67.73)$ = 2956 – 2700 = 256

$eval(V_6)$ = $f(147.79, 50.47)$ = 3031 – 2700 = 331

$eval(V_7)$ = $f(147.20, 69.43)$ = 2827 – 2700 = 127

$eval(V_8)$ = $f(153.89, 63.20)$ = 3126 – 2700 = 426

$eval(V_9)$ = $f(144.70, 63.69)$ = 3021 – 2700 = 321

$eval(V_{10})$ = $f(158.20, 65.29)$ = 3081 – 2700 = 381

It is clear that chromosome V_8 is the strongest one and that chromosome V_7 is the weakest one.

3.3 Selection

In most practices, a roulette wheel approach is adopted as the selection procedure; it belongs to the fitness-proportional selection and can select a new population with respect to the probability distribution based on fitness values. The roulette wheel can be constructed as follows;

1.Calculate the fitness value $eval(V_k)$ for each chromosome V_k:

$$eval(V_k) = f(X),\ k = 1,2,\ldots,pop\_size \tag{13}$$

2.Calculate the total fitness for the population :

$$F = \sum_{k=1}^{pop\_size} eval\ (V_k) \tag{14}$$

3.Calculate selection probability p_k for each chromosome V_k:

$$p_k = \frac{eval(\mathrm{Vk})}{\mathrm{F}},\qquad k = 1,2,\ldots,pop\_size \tag{15}$$

4.Calculate cumulative probability q_k for each chromosome V_k:

$$q_k = \sum_{j=1}^{k} p_j,\qquad k = 1,2,\ldots,pop\_size \tag{16}$$

The selection process begins by spinning the roulette wheel *pop_size* times: each time, a single chromosome is selected for a new population in the following way:

Procedure: Selection

Step 1. Genarate a random number r from the range [0 ,1].

Step 2. If $r \le q_1$, then select the first chromosome V_1; otherwise, select the kth chromosome $V_k (2 \le k \le pop\_size)$ such that $q_{k-1} < r \le q_k$.

The total fitness F of the population is

$$F = \sum_{k=1}^{10} eval\ (V_k) = 3159 \qquad (17)$$

The probability of a selection p_k for each chromosome $V_k (k = 1,\ldots,10)$ is as follows:

$p_1 = 0.0988$ $p_2 = 0.1292$ $p_3 = 0.1250$

$p_4 = 0.0639$ $p_5 = 0.0810$ $p_6 = 0.1048$

$p_7 = 0.0402$ $p_8 = 0.1349$ $p_9 = 0.1016$

$p_{10} = 0.1206$

The cumulative probabilities q_k for each chromosome $V_k (k = 1,\ldots,10)$ is as follows:

$q_1 = 0.0988$ $q_2 = 0.2279$ $q_3 = 0.3530$

$q_4 = 0.4169$ $q_5 = 0.4979$ $q_6 = 0.6027$

$q_7 = 0.6429$ $q_8 = 0.7778$ $q_9 = 0.8794$

$q_{10} = 1.0000$

Now we are ready to spin the roulette wheel 10 times, and each time we select a single chromosome for a new population. Let us assume that a random sequence of 10 numbers from the range [0, 1] is as follows:

0.3014	0.3221	0.7665	0.8819
0.3509	0.5834	0.1776	0.3432
0.0327	0.1976		

The first number $r_1 = 0.3014$ is greater than q_2 and smaller than q_3, meaning that the chromosome V_3 is selected for the new population; the second number $r_2 = 0.3221$ is greater than q_2 and smaller than q_3, meaning that the chromosome V_3 is selected for the new population; and so on. Finally, the new population consists of the following chromosomes:

V'_1 = [000100111100001010I] (V_3)

V'_2 = [0001001111000010101] (V_3)

V'_3 = [1011000111010100011] (V_8)

V'_4 = [1110100011100001110] (V_{10})

V'_5 = [0001001111000010101] (V_3)

V'_6 = [0110001110000011000] (V_6)

V'_7 = [0100111000011110101] (V_2)

V'_8 = [0001001111000010101] (V_3)

V'_9 = [1110001100111100100] (V_1)

V'_{10} = [0100111000011110101] (V_2)

3.4 Crossover

Crossover used here is one-cut-point method, which randomly select on cut-point and exchanges the right parts of two parents to generate offspring. Consider two chromosomes as follows, and the cut-point is randomly selected after the 12th gene:

↓

V_1 = [0001001111000010101]

V_2 = [1011000111010100011]

The resulting offspring by exchanging the right parts of their parents would be as follows:

V'_1 = [000100111100 0100011]

V'_2 = [101100011101 0010101]

The probability of crossover is set as p_c = 0.5, so we expect that, on average, 50% of chromosomes undergo crossover.

3.5 Mutation

Mutation alters one or more genes with a probability equal to the mutation rate. Assume that the 11th gene of the chromosome V_1 is selected for a mutation. Since the gene is 0, it would be flipped into 1. Thus the chromosome after mutation would be

$$V_1 = [000100111100010001\,1]$$

$$V_2 = [0001001111100100011]$$

The probability of mutation is set as $p_m = 0.1$ so we expect that, on average, 10% of total of population would undergo mutation. Thus we random select the chromosomes and random select gene to mutate.

After the chromosomes have finished in crossover and mutation procedures and turn back to evaluation procedure then continue GA operations until the experiment will be end.