

中華民國第四十六屆中小學科學展覽會
作品說明書

高職組 電子、電機及資訊科

091008

節能式太陽能自然公園

學校名稱： 高雄市私立大榮高級中學

作者：	指導老師：
職二 蔡易輯	林萬成
職二 陳甲均	張簡助立
職二 顏宏致	
職二 黃國鵬	

關 鍵 詞：節能式、太陽能電池模組、自然公園

壹、 摘 要

目前日常生活中所使用的自動化產品皆需要使用電力，因此電對人類來說是非常重要的能源，然而目前台灣所有的電力大部是由水力、火力、核能發電產生，核能發電雖較為簡捷、快速、但卻對我們的環境有造成傷害之虞，核廢料也成了破壞地球自然生態的重要因素之一。全世界大部分國家都簽署廢除核能發電，因此必須加速開發再生能源的發電裝置。試想，若我們能以太陽能發電逐漸來取代其他污染性發電，則省去了使用污染性電廠的頻繁，對我們的環境、生態來說應是一大福音。

太陽能的優點為在地球的任何地方均可取得。現代的太陽能系統，在每天日照時間相當短的國家，也可以經濟有效地提供大量電能，太陽能的科技應用甚廣，例如太陽能的計算機、手錶，在市面上很普遍。另外，利用太陽能來驅動的熱水器和太陽屋，在外國亦屢見到不鮮。

而太陽能的交通工具，在一些科技較先進的國家亦有研究發展，例如美國、日本。這些交通工具包括飛機、汽車等。現今人類最關注的能源問題，太陽能發電亦能貢獻一臂之力。

本文即以此為出發點，規劃一節能式太陽能自然公園供電系統，並予以測試及安全保護，驗證其可行性並實現之。

貳、 研究動機

在能源短缺的時代，用無污染的大自然能源，提供自然公園所需之電力，以達節約能源之效果，以最具經濟效益、最多、最方便的能源，達到再生能源物盡其用的最高原則。在這個太陽能自然公園模型供電系統中，它的電力是由太陽能發電並儲存於蓄電池，以提供照明系統以及步道燈用電；而灑水系統以及人工瀑布所使用的電力是 AC110V，在經由 DC to AC 轉換器，以提供灑水系統以及人工瀑布幫浦用電。

照明系統使用白光 LED，每把路燈使用 5 顆白光 LED 串聯，以 LED 取代目前照明系統來達到省電的目的。至於步道燈系統則使用紅光、綠光、藍光三種 LED，在判斷是否該開啓照明系統則使用光敏電阻來判斷，且以分時多工方式經由單晶片 P89C51RD+控制電路，以達到目。

参、 研究目的

- 一、 拋磚引玉，使研究風氣更活潑化、實際化及應用化，以推展應用知識的空間，以便服務人群、社會，期許大眾能為改善人類生活的繼起意義而活。
- 二、 維護大自然環境，並兼顧科技發展，提高人類生活品質而盡力。
- 三、 用太陽能，將沒有可動部份之吵雜，是為安靜的能源轉換。
- 四、 即使用擴散光源亦可發電。
- 五、 可利用原已被放棄之能源來促使光發電。
- 六、 提高最佳的轉換效率。
- 七、 使自然能源應用更能邁向至真至善至美的境界。

肆、 研究設備及器材

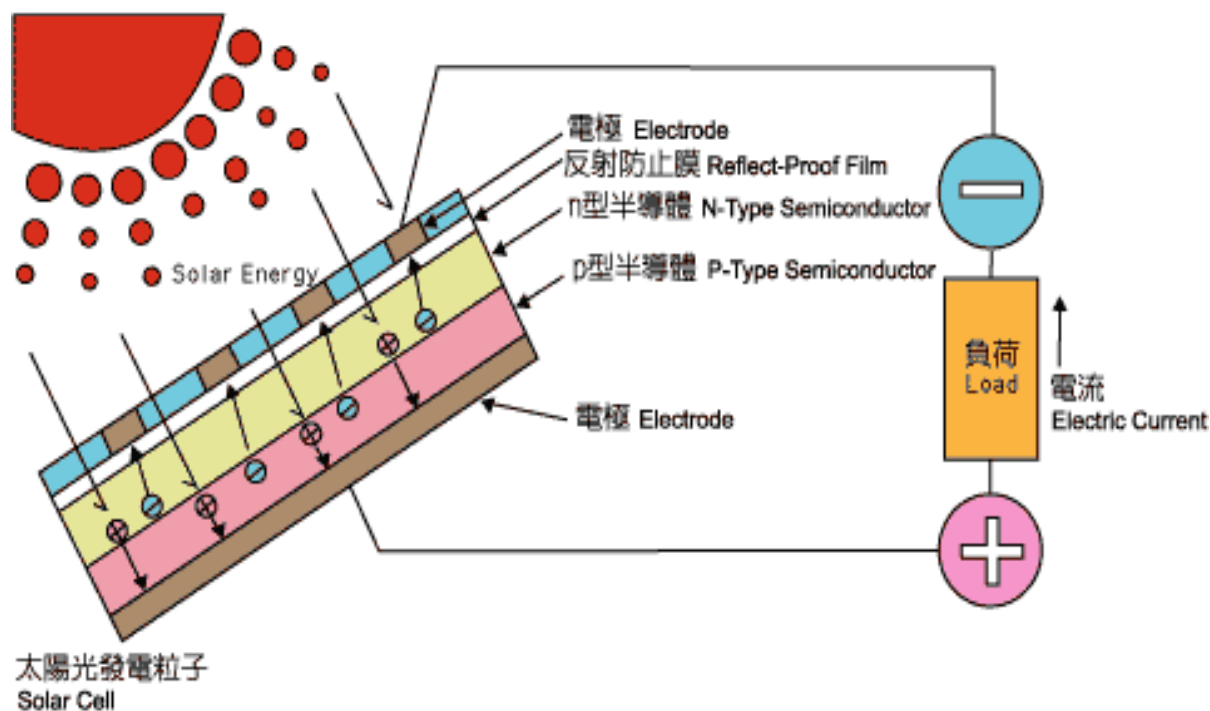
- 一、 太陽能光電板 x 1 只
- 二、 DC/AC Converter x 1 台
- 三、 充電器 x 1 只
- 四、 89051 控制組 x 1 台
- 五、 抽水馬達 x 1 只
- 六、 壓縮馬達 x 1 只
- 七、 高量路 LED x 1 只
- 八、 電壓錶 x 1 台
- 九、 電流錶 x 1 台
- 十、 開關 x 5 只
- 十一、 燒錄器 x 1 台
- 十二、 電腦 x 1 台
- 十三、 示波器 x 1 台
- 十四、 三用電錶 x 1 台
- 十五、 排線 x 1 只
- 十六、 庭園造景 x 1 批

伍、 研究過程與方法

一、 太陽能原理

太陽能供電系統具備獨立性、高移動性與可擴充性。由於太陽能電池的輸出特性隨環境照度與溫度變化，為獲得任意條件下的最大功率輸出，太陽能電池的主要成分是「矽」〈純度要很高〉，一個矽原子有 4 個電子，但太陽能電池的矽必須要達到 8 個才能達到穩定狀態。在結晶狀態下，利用矽原子互補，就可以達穩定狀態。而有一部分可以參雜「砷」做出 N 形半導體，另一部份混入「硼」，做出 P 形半導體。太陽能電池就是將 P 形和 N 形半導體結合，此時交界區的電子電洞會中和，而在電中性地區形成一到屏障，電子與電洞無法跨越介面，電路呈現斷路。當陽光照射近來時，光子提供額外動能，激發電子穿越屏障行成通路，將太陽能電池兩側接上電路，即可使機械運轉。

另地球上所有綠色植物，都是通過光合作用來直接利用太陽能的。於是有人提出利用葉綠素再造太陽能電池的設想，根據計算，利用葉綠素製造的太陽能電池，效果相當理想。科學家研究其他半導體和染料，以找出模擬光合作用的最佳搭配，實現通過光合作用利用太陽直接發電，這對解決人類對電力的需求具有特別意義。利用光電效應使太陽能轉化成電能以太陽的輻射能力使太陽能直接轉化成 電能的製品。其轉換結構圖如下：



二、 太陽能發電科技發展

太陽能 (Solar energy) 是再生能源 (Clean and renewable energy) 之一。由於太陽能是一項取之不盡，用之不竭的能源，再生能源包括太陽能、風力 (Wind power)、地熱 (Geothermal)、生物大量廢棄物 (Biomass)、清靜煤炭 (Clean coal)、與小型水力發電 (Small scale hydroelectricity) 及燃料電池 (Fuel cell) 利用氫氣 (Hydrogen) 做為能源來源等。而太陽能電力科技開發使用相當早，因此已有相當長的歷史，太空中之人造衛星即利用太陽能板 (Solar panels) 吸收太陽日照光能轉化為電能供應人造衛星電源，除人造衛星外，偏遠地區高科技設施機艙、通信高塔等一般電力線路無法連接者，均利用太陽能板發電提供電力，然因傳統上以矽為材料做為太陽能電池 (Solar cells)，其矽材料與製造電腦晶片的矽材料相同，相對上相當昂貴，用於製造昂貴電腦晶片價格可以接受，但用以製造太陽能電池 (Solar cells) 發電，相對上發電成本要比一般傳統上用化石燃料做為能源之發電成本高出許多，因此太陽能電力在發電成本上難與傳統化石燃料能源電力相競爭。

此外，由於太陽能日照時間變化，用以吸收日照的太陽能板 (Solar panels) 面積佔用空間廣大，要有相當規模的發電量必須要有廣大的面積可以利用並且要有充足日照太陽光能，加以太陽能電池將日照光能轉化成電能的效率仍然偏低、大約只維持在 15% 至 25% 之間，太陽能發電尚不能普及，以上這些都是原因。目前全球太陽能發電尚不及全球發電總量的 0.4%，全球太陽能發電以日本領先，其次為德國、美國、西班牙等。西班牙目前使用的太陽能源板數量約為 58 萬 1 千平方公尺，到 2010 年將增加為當前的 10 倍。在歐洲國家中，德國當前使用太陽能源板數的總量領先各國，約為 540 萬平方公尺，西班牙雖然落後甚多，但它卻是全球太陽能源板最大生產國。

近年來，很多製造太陽能面板與太陽能電池廠商捨棄昂貴的矽晶材料，尋求改用較為便宜的非結晶形矽材料 (Cheap amorphous silicon) 或甚至於用半導體合金之材料 (Semiconductor alloys) 製成很薄的薄膜 (Thin film)，展開成為薄膜層，可以超薄到幾千毫米之厚度 (a few thousandths of a millimeter)，此種超薄膜較諸用以做為傳統太陽能電池之材料薄一百倍左右，因為超薄而可以使用較少的矽材料而且也易於大量生產，且相對顯著便宜許多，此種超薄半導體材料較矽晶材料缺乏電子穿梭效率，但可以彌補的是其可以較矽晶材料接收較多的光量子。導電塑膠材料可以導電發光，已被運用於製造發光平面顯示板 (Flat-panel displays) 及照相膠捲等，此項新材料如可以被反向應用，在使用過程上用於吸收光產生電能，若可以成功，那麼導電塑膠材料將是不貴的太陽能電池 (inexpensive solar cells) 材料。在未來太陽能發電科技發展應用上，將有深遠巨大的影響。

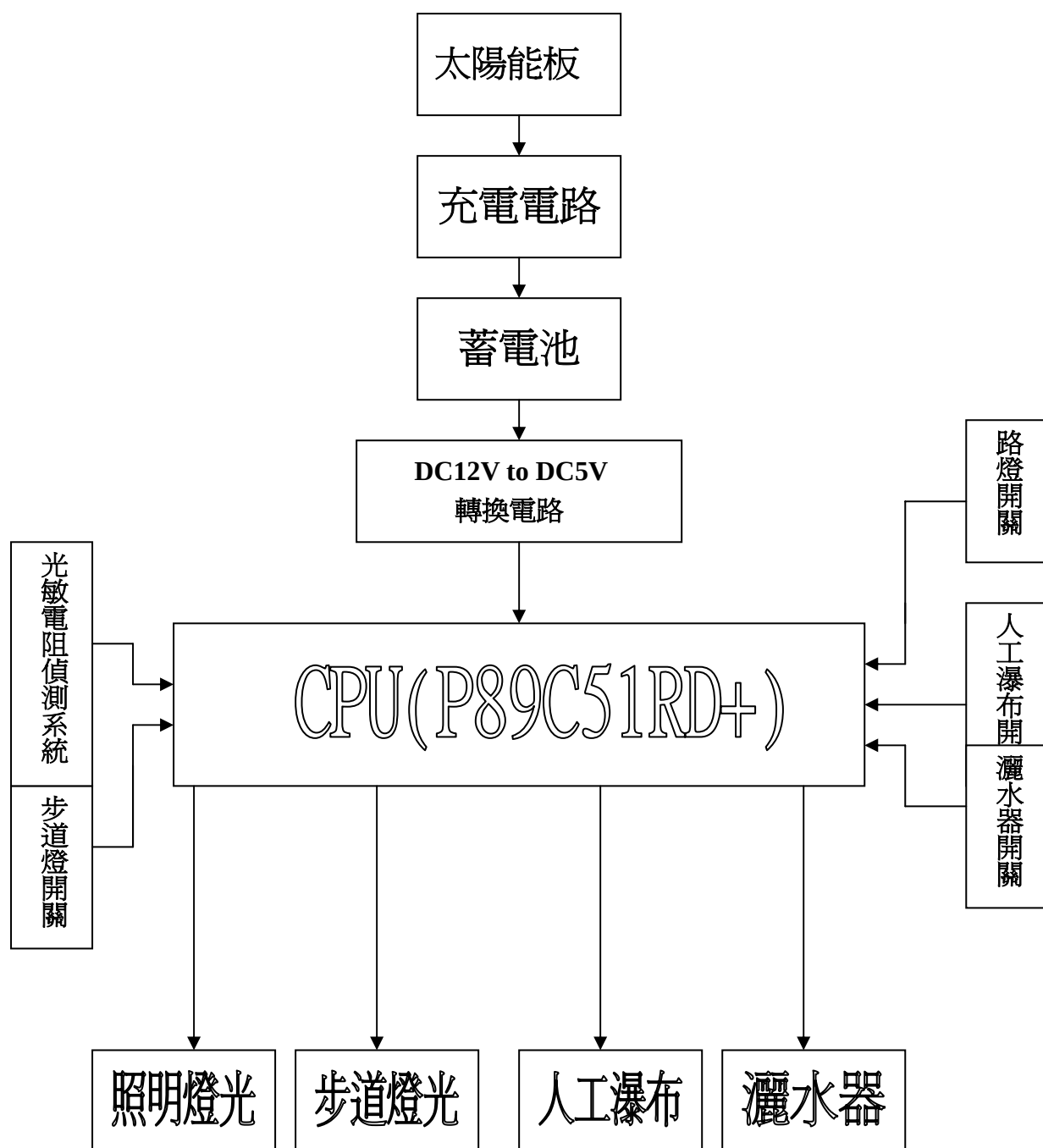
三、 太陽能系統應用

人類在 18 世紀初啟動了工業革命的按鈕，開始了以耗用大量石化能源的方式來支撐工業經濟的持續發展，導致濫用各種資源，使生態環境受到嚴重的破壞。全球普遍意識到，人類必須在兼顧經濟成長(Economic Growth)與環境保護(Environmental Protection)及取得充足穩定的能源(Energy Security)的 3E 策略下相互調和，才能有永續發展的可能。在聯合國氣候綱要公約(UNFCCC)及在 1997 年的京都議定書(Kyoto Protocol)中，所表達的共識訊息，已明確勾勒人類所該嚴肅面對的生存課題。

人類照明能源的消耗，佔所有能源總消耗約 20~30%的比例。目前在各國所推行的照明省能政策中，不外乎是以獎勵措施、補貼條款、節稅措施等方式來鼓勵，並採用下列五種方式來減少照明之耗能支出，分別是利用(1)太陽能(2)照明設備的改善(3)光源效率的提昇(4)配合控光系統(5)替代新能源。目前各國在開發新能源的領域，探討最多且技術最成熟，及最具有市場規模者，首推太陽光電能源的開發運用。

在石化等能源危機及環保意識的高漲下，尋求一種無污染、無公害、無所不在且取之不盡用之不竭的新能源在科技上是具有歷史意義的，而其中當以太陽能光電轉換最合時宜，目前太陽能之使用並非發達，且都具有向上發展之空間，若使用太陽能發電之研究停滯，實為一大遺憾。

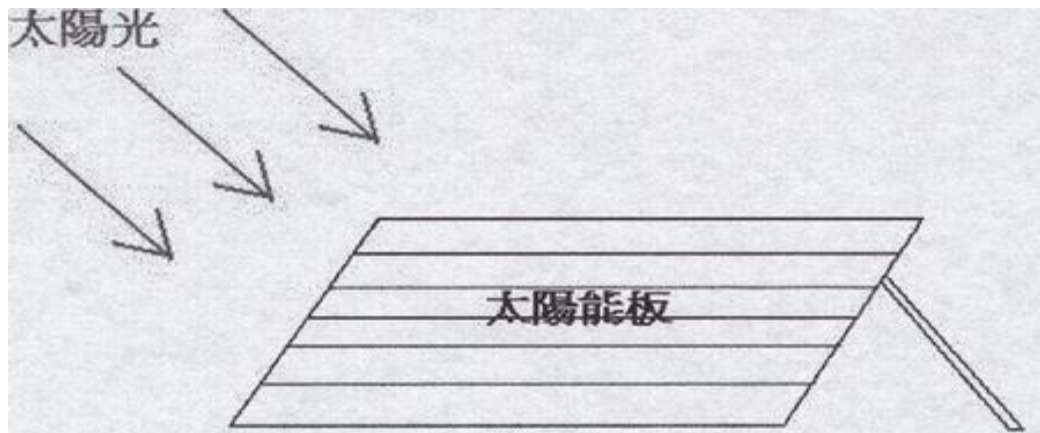
四、 太陽能自然公園系統方塊圖



五、方塊圖說明

在這自然公園供電系統中，它的電力是以太陽能發電在經由充電電路儲存於蓄電池，以供給系統裝置用電。敘述如下：

1· **太陽光發電**：利用太陽能集光板，收集太陽光產生電力。



以 23.5 度角斜放並面對太陽，使太陽光均勻照射在太陽能集光板上，達到較高的光能吸收，使其電力轉換較佳。

我們尚可利用費奈爾凸透鏡均勻集光，使其集光面積更寬廣並均勻聚集於太陽能板上，達成高電力轉換之效能，如圖 1-1 所示

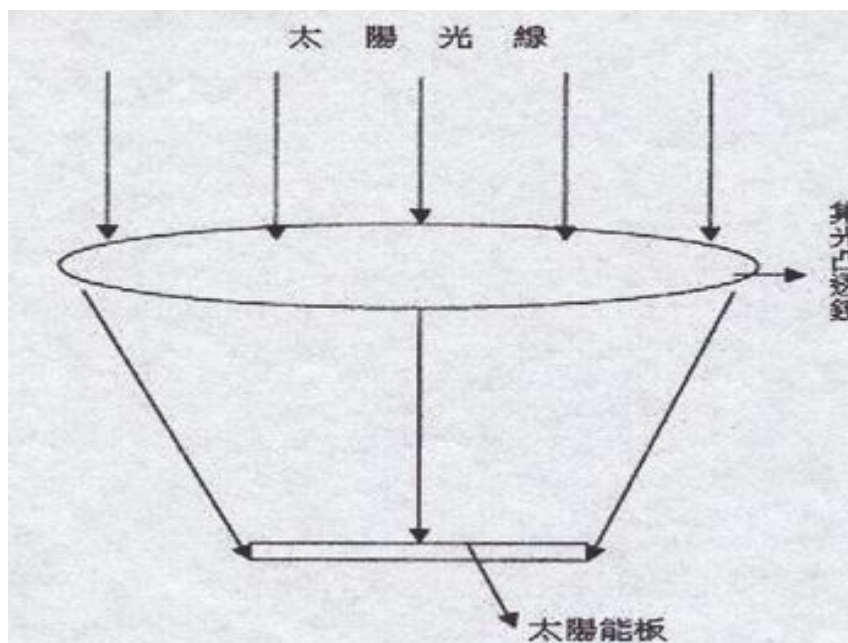


圖 1-1 凸透鏡均勻集光於太陽能板

將其構想應用於太陽能集光板上，是將數片費奈爾凸透鏡採蜂巢式星點狀設置，製成擬蜂巢式玻璃遮罩集光板，如圖 1-2 所示

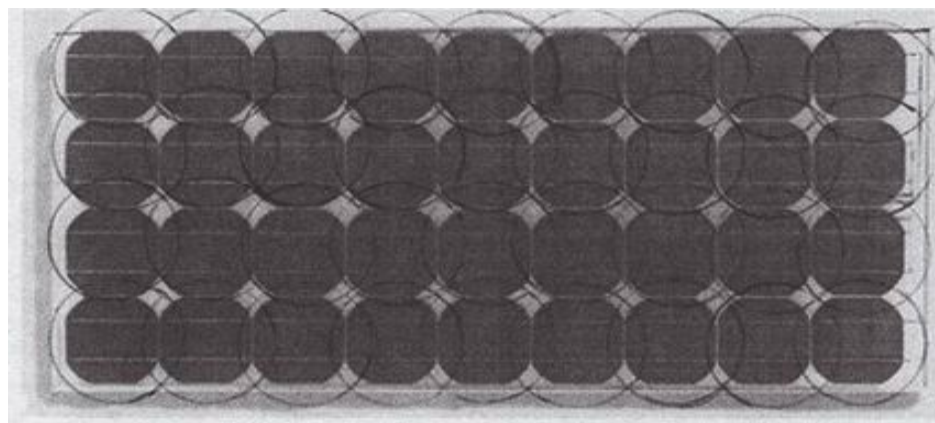


圖 1-2 擬蜂巢式玻璃遮罩集光板

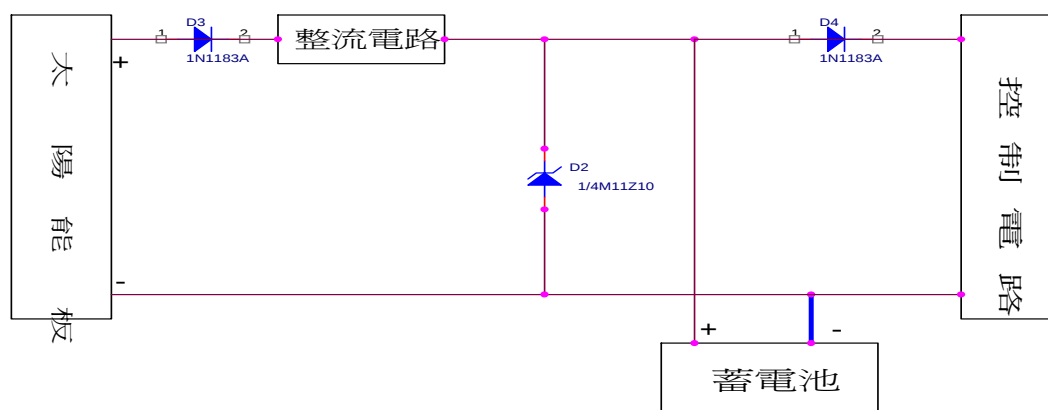
將擬蜂巢式集光型遮罩與太陽能集光板、元件溫降致冷器結合一體之架構，即為擬蜂巢式太陽能模組，其中擬蜂巢式集光型遮罩與太陽能集光板之間距，以太陽光能均勻集光於太陽能集光板上為依據而做適度調整。

本模組最大優點可因地制宜，且擴充性佳，同時致冷器之集熱效應亦可與太陽能熱水器並合，造成電熱共同集收，提高太陽能之轉換效率。

上述為『擬蜂巢式太陽能模組』，使用此模組確實可大幅增加光能吸收，產生更高的能量轉換，但因光能吸收量大，造成模組溫度上升以致轉換效率降低，所以模組下方需裝置元件溫降致冷器，元件溫降致冷器以白鐵或鋁製材料製造形成，其中以流動冷卻水降溫。

2. 充電電路：

我們藉由太陽能板將光能轉化成電能，再透過整流電路使直流穩定，用稽納二極體穩壓。其充電電路設計圖如下：

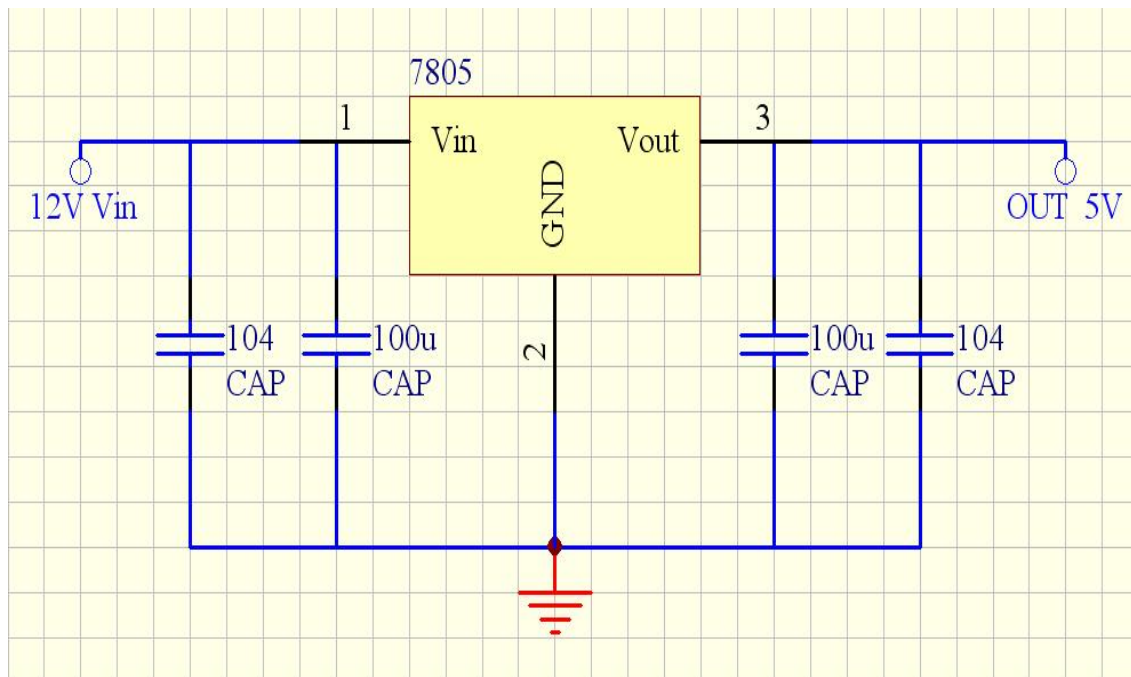


3・蓄電池：

可利用充電而重複使用的二次電池即為蓄電池或充電電池，論及二次電池則提到種類、壽命、充放電安全性、保存性、高效率充放電對應性、耐漏液性、安全性、信賴性、容量等許多的指標，這樣的檢討項目與限制，困擾著電路設計者，然在這些特性中去發現特點，進而給予設計者極大空間與自由性，而成為創新發明得以繼續連結的理由。

4・DC12V to DC5V 轉換電路：

利用此電路將蓄電池直流 12V 轉換成直流 5V，以提供公園控制電路所需之電源。

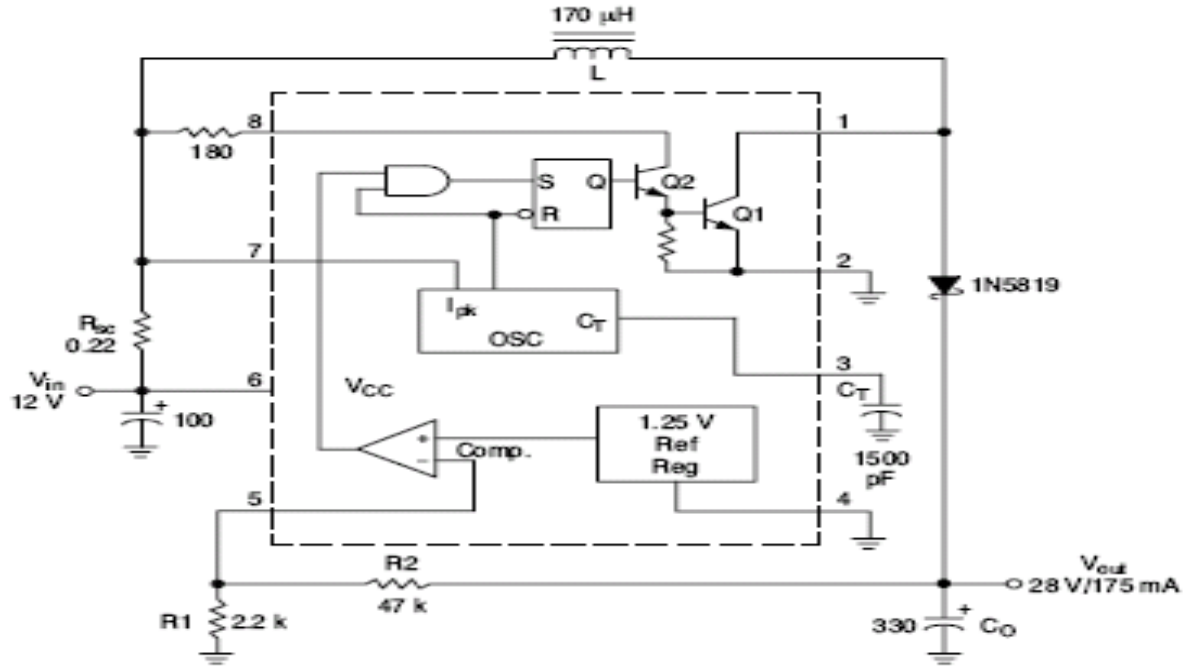


5・CPU(P89C51RD+)：

此顆 8051 用來接收 P0.0~P0.4 開關輸入及光敏電阻偵測系統輸入，並控制人工瀑布、灑水器、照明燈、步道燈 ON 或者 OFF。

6・照 明 燈 光：

這是由六把 5 顆白光 LED 串聯組成，所需電壓為 DC16V，其提升電壓電路如下：



Calculation	Step-Up	Step-Down	Voltage-Inverting
t_{on}/t_{off}	$\frac{V_{out} + V_F - V_{in(min)}}{V_{in(min)} - V_{sat}}$	$\frac{V_{out} + V_F}{V_{in(min)} - V_{sat} - V_{out}}$	$\frac{ V_{out} + V_F}{V_{in} - V_{sat}}$
$(t_{on} + t_{off})$	$\frac{1}{f}$	$\frac{1}{f}$	$\frac{1}{f}$
t_{off}	$\frac{t_{on} + t_{off}}{\frac{t_{on}}{t_{off}} + 1}$	$\frac{t_{on} + t_{off}}{\frac{t_{on}}{t_{off}} + 1}$	$\frac{t_{on} + t_{off}}{\frac{t_{on}}{t_{off}} + 1}$
t_{on}	$(t_{on} + t_{off}) - t_{off}$	$(t_{on} + t_{off}) - t_{off}$	$(t_{on} + t_{off}) - t_{off}$
C_T	$4.0 \times 10^{-5} t_{on}$	$4.0 \times 10^{-5} t_{on}$	$4.0 \times 10^{-5} t_{on}$
$I_{pk(switch)}$	$2I_{out(max)} \left(\frac{t_{on}}{t_{off}} + 1 \right)$	$2I_{out(max)}$	$2I_{out(max)} \left(\frac{t_{on}}{t_{off}} + 1 \right)$
R_{sc}	$0.3/I_{pk(switch)}$	$0.3/I_{pk(switch)}$	$0.3/I_{pk(switch)}$
$L_{(min)}$	$\left(\frac{(V_{in(min)} - V_{sat})}{I_{pk(switch)}} \right) t_{on(max)}$	$\left(\frac{(V_{in(min)} - V_{sat} - V_{out})}{I_{pk(switch)}} \right) t_{on(max)}$	$\left(\frac{(V_{in(min)} - V_{sat})}{I_{pk(switch)}} \right) t_{on(max)}$
C_O	$9 \frac{I_{out} t_{on}}{V_{ripple(pp)}}$	$\frac{I_{pk(switch)}(t_{on} + t_{off})}{8V_{ripple(pp)}}$	$9 \frac{I_{out} t_{on}}{V_{ripple(pp)}}$

V_{sat} = Saturation voltage of the output switch.

V_F = Forward voltage drop of the output rectifier.

The following power supply characteristics must be chosen:

V_{in} - Nominal input voltage.

V_{out} - Desired output voltage, $|V_{out}| = 1.25 \left(1 + \frac{R2}{R1} \right)$

I_{out} - Desired output current.

f_{min} - Minimum desired output switching frequency at the selected values of V_{in} and I_O .

$V_{ripple(pp)}$ - Desired peak-to-peak output ripple voltage. In practice, the calculated capacitor value will need to be increased due to its equivalent series resistance and board layout. The ripple voltage should be kept to a low value since it will directly affect the line and load regulation.

NOTE: For further information refer to Application Note AN920A/D and AN954/D.

因為路燈是用 5 顆 LED 串聯，所以需要 16V 提供電源。依照公式 $V_{out} = 1.25 \left(1 + \frac{R_2}{R_1} \right)$

求得 $\frac{R_2}{R_1} = 11.8$ ， R_1 使用 $1K\Omega$ 、 R_2 使用 $12K\Omega$

7・步道燈光：

這是由三種顏色 LED 所組成，依照紅色、綠色、藍色依序閃爍，再電路中我們共有 63 組步道燈，每組以三顆 LED 採串接方式在由程式控制閃爍順序。

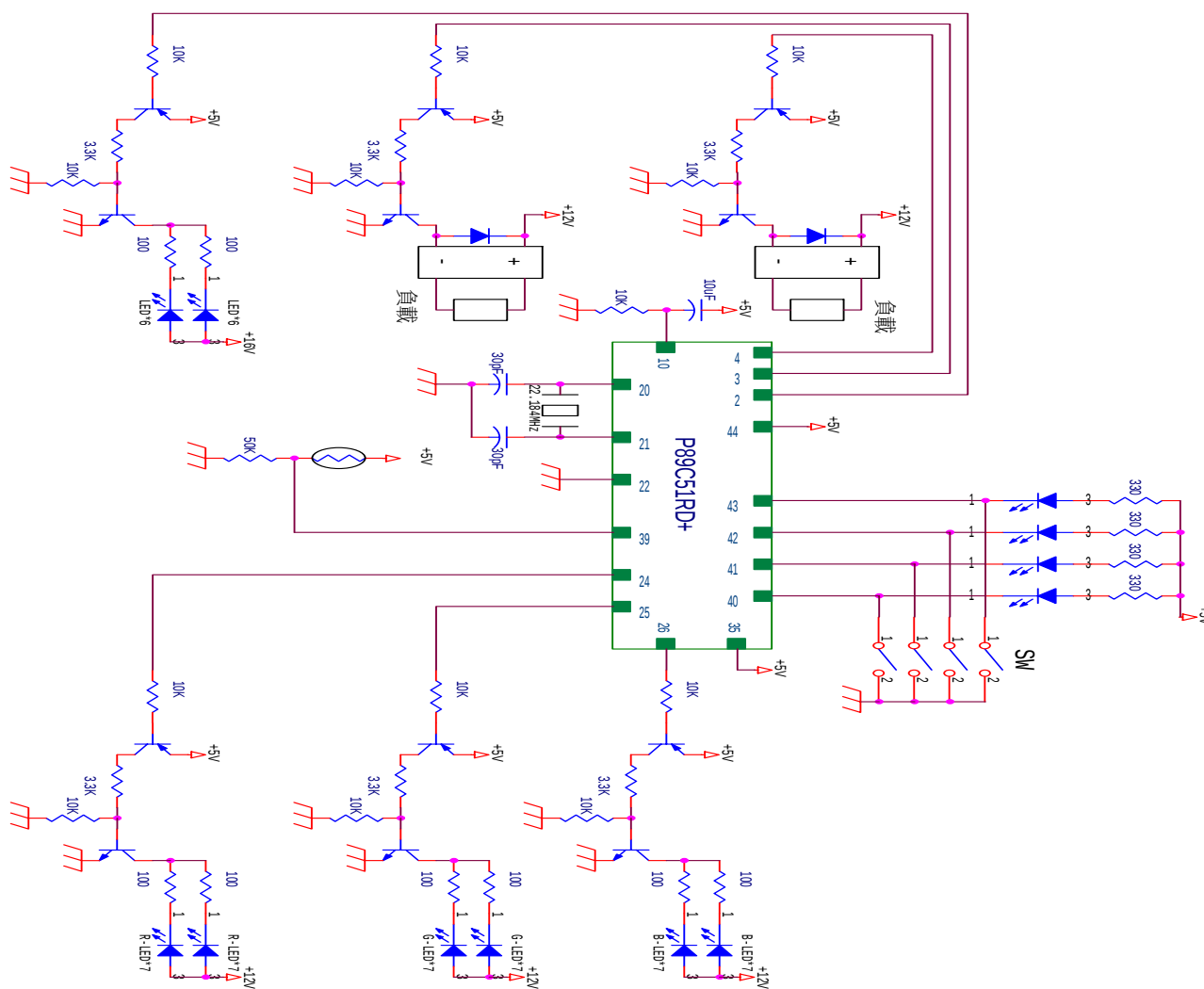
8・人工瀑布：

由單晶片 P89C51RD+控制 RELAY 啟動幫浦動作，電力則由蓄電池經過 DC to AC 轉換器提供，透過幫浦將水送到假山流出在流向位於假山下方的水池，再流回蓄水桶中如此得以循環，以達到節省水資源。

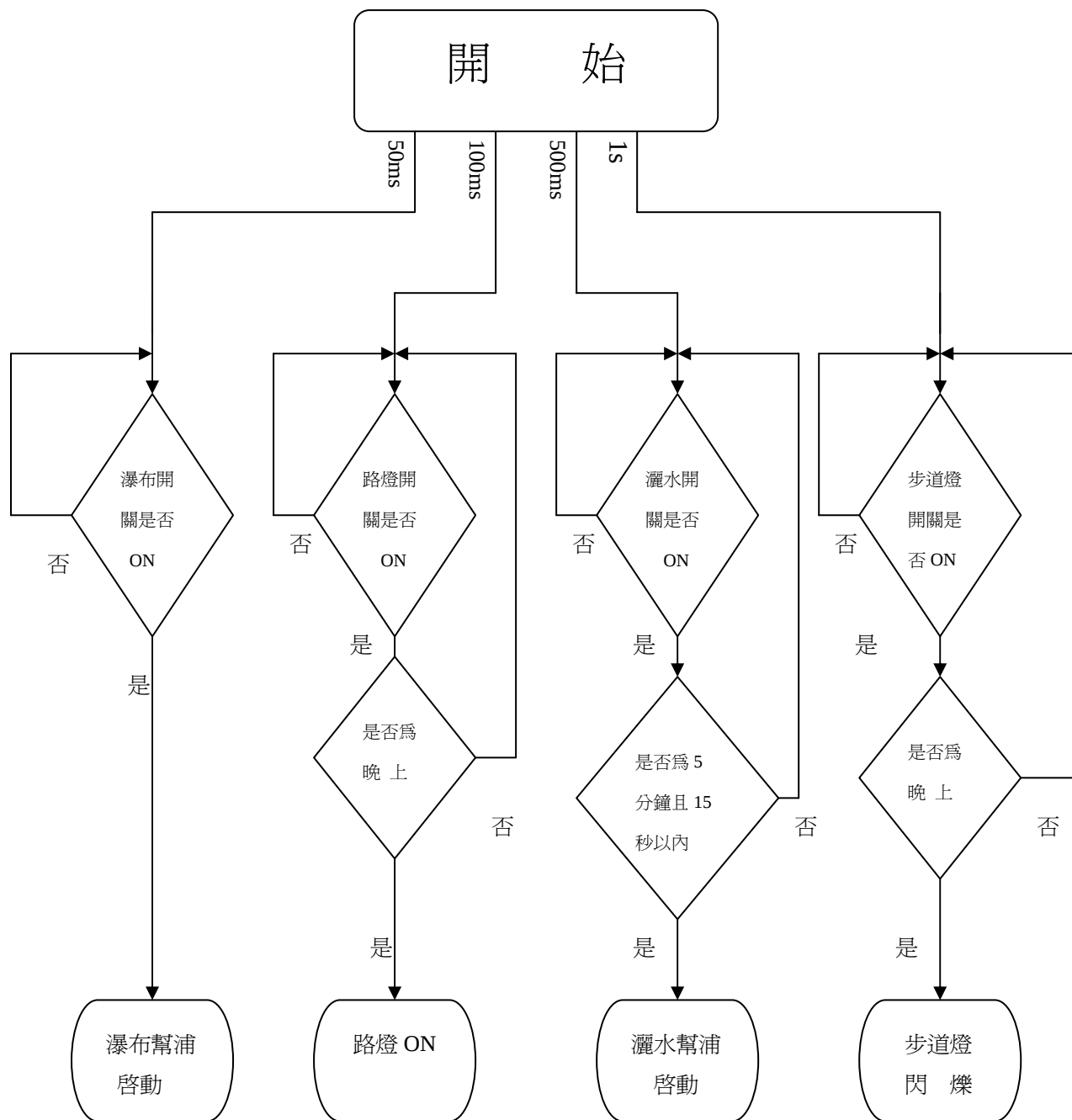
9・灑水器：

由單晶片 P89C51RD+控制 RELAY 啟動幫浦動作，電力則由蓄電池經過 DC to AC 轉換器提供，透過幫浦將水送到灑水器灑出，由程式控制每 5 分鐘灑水一次每次持續 15 秒。

10.電路圖：



11.程式流程圖



12.程式

(a)main.c

```
#include "P89C51RX.h"
#include "timer.h"
#include "task.h"
sfrWDT    =    0xC1;

unsigned char ct;
unsigned char led[]={0xfe,0xfd,0xfb,0xf8,0xff};
unsigned short ct1;
unsigned short ct2;
unsigned short ct3;
unsigned char i;
//=====
//[    void SFR_Init(void)                                ]
//[    System Register Initial                            ]
//=====
void      Sfr_Init(void)
{
    WDT &= 0xbf;
    PCON = 0x00;
    IE = 0xB2;                // Interrupt Setting
    IP = 0x02;
    TCON = 0;
    TMOD = 0x21;
    SCON = 0x0D0;
    IT0=1;
    IT1=1;

    TR0=0;
    TR1=1;                    // RS232 Interrupt  SW
    TR2=0;

    PS=1;
    ES = 1;

    Timer_0_Init();          // Sound Interrupt
```

```

    Timer_2_Init();                // Task Interrupt
    ct=0;
    ct1=0;
    ct2=0;
    ct3=0;
    i=0;
}

//=====
//[    void main(void)                ]
//[    Entry Pointer                  ]
//=====
void main(void)
{
    P0=0xff;
    P1=0xff;
    P2=0xff;
    Sfr_Init();
    while(1)
    {
        TaskProcess();
    }
}

```

主程式說明:

無窮回圈來執行分時多工系統所分割的時間間隔在 Task..c 所有的動作，每當所分割的時間間隔，時間一到便去執行一次。

(b)Timer.c

```

#include "P89C51RX.h"
#include <Reg52.h>
#include "timer.h"
#include "task.h"

unsigned char Timer_Second_Divider;
unsigned char TaskFlag;
unsigned char Timer_Hour,Timer_Minute,Timer_Second;

```

```

//=====
//[      void Timer_2_Init(void)                                ]
//[      Init the Timer2 Register & Init Timer2 to be 10mS Interrupt  ]
//=====
void    Timer_2_Init(void)
{
    RCAP2H = THDiv10mS;
    RCAP2L = TLDiv10mS;
    T2CON = 4;
    TR2 = 1;
    Timer_Second_Divider=0;
    TaskFlag=0;
}

//=====
//[      void Timer_0_Init(void)                                ]
//[      Init the Timer0 Register & Init Timer0 to be 10mS Interrupt  ]
//[      Disable Timer0 Interrupt                                ]
//=====
void    Timer_0_Init(void)
{
    TH0 = THDiv10mS;
    TL0 = TLDiv10mS;
    TR0 = 0;
}

//=====
//[      void Timer_2_Int(void)                                ]
//[      Timer2 Interrupt Subroutine                                ]
//[      TaskTimeBase Timer : 10mS Interrupt                    ]
//=====
void    Timer_2_Int(void) interrupt 5
{
    if ((Timer_Second_Divider%5==0))
    {
        Task50mSFlag=1;
    }
}

```

```

}

if ((Timer_Second_Divider%10==0))
{
    Task100mSFlag=1;
}

if ((Timer_Second_Divider%20==0))
{
    Task200mSFlag=1;
}
if ((Timer_Second_Divider%50==0))
{
    Task500mSFlag=1;
}

Timer_Second_Divider++;
if (Timer_Second_Divider==100)
{
    Timer_Second_Divider=0;
    Task1SecFlag=1;
    Timer_Second++;
    if (Timer_Second>59)
    {
        TaskMinFlag=1;
        Timer_Second=0;
        Timer_Minute++;

        if (Timer_Minute>59)
        {
            TaskHourFlag=1;
            Timer_Minute=0;
            Timer_Hour++;
            if (Timer_Hour>23)
            {
                Timer_Hour=0;
            }
        }
    }
}

```

```

        }
    }
}
TF2 = 0;
}

```

Timer.c 程式說明:

首先設定一個時間為 10ms，來分割我們所需的分時多工時間間隔，
例如將 10ms 除 5 取餘數，當餘數等於 0 時，就分割好 50ms 的時間，
其他由此類推。

(c)Task.c

```

#include "P89C51RX.h"
#include "task.h"
#include "timer.h"

bit Task50mSFlag;
bit Task100mSFlag;
bit Task200mSFlag;
bit Task500mSFlag;
bit Task1SecFlag;
bit Task2SecFlag;
bit TaskMinFlag;
bit TaskHourFlag;
extern unsigned char ct;
extern unsigned short ct1;
extern unsigned short ct2;
extern unsigned short ct3;
extern unsigned char i;
extern unsigned char led[];

//=====
//[      void Task50mS()                               ]
//[      50mS Tasking Function:                         ]
//[      1. Keyboard_ReadCode()                         ]
//=====
void Task50mS(void)
{

```

```

    P0_0=1;
    if (P0_0==0)
    P1_1=0;
    else
    P1_1=1;
}

```

```

//=====================================================
//[      void Task100mS()                                     ]
//[      100mS Tasking Function:                             ]
//[      None                                                  ]
//=====================================================
void Task100mS(void)
{
    P0_2=1;
    if (P0_2==1)
    {
        P1_0=1;
        ct1=0;
    }
    if (P0_2==0)
    {
        if(ct1==599)
        P1_0=0;
        P0_4=1;
        if(P0_4==0&&ct1<599)
        ct1++;
        if(P0_4==1)
        {
            P1_0=1;
            ct1=0;
        }
    }
}
}

```

```

//=====
//[      void Task500mS()                                ]
//[      500mS Tasking Function:                          ]
//[      None                                              ]
//=====
void    Task500mS(void)
{
    P0_1=1;
    if(P0_1==1)
    {
        P1_2=1;
        ct2=0;
        ct3=0;
    }
    if(P0_1==0)
    {
        if(ct2<599)
        {
            if(ct3<30)
            P1_2=0;
            else
            P1_2=1;
            ct2++;
            ct3++;
            if(ct2==599)
            {
                ct2=0;
                ct3=0;
            }
        }
    }
}

```

```

//=====
//[      void Task1Sec()                                ]
//[      1 Sec Tasking Function:                        ]
//[      1. Clock_Display                               ]
//=====
void    Task1Sec(void)
{
    P0_3=1;
    if (P0_3==1)
    {
        P2=0xff;
        ct=0;
        i=0;
    }
    if (P0_3==0)
    {
        if(ct==59)
        {
            P2=led[i];
            i++;
            if(i>4)
            i=0;
        }
        P0_4=1;
        if(P0_4==0&&ct<59)
            ct++;
        if(P0_4==1)
        {
            P2=0xff;
            ct=0;
        }
    }
}

```

```

//=====
//[    void TaskProcess(void)                                ]
//[    Check Tasking Flag , if TaskFlag was set              ]
//[    then excute Task program                               ]
//[    Task : ForeverTask                                    ]
//=====
void    TaskProcess(void)
{
    if ((Task50mSFlag)==1)
    {
        Task50mSFlag=0;
        Task50mS();
    }
    if ((Task100mSFlag)==1)
    {
        Task100mSFlag=0;
        Task100mS();
    }
    if ((Task200mSFlag)==1)
    {
        Task200mSFlag=0;
        Task200mS();
    }
    if ((Task500mSFlag)==1)
    {
        Task500mSFlag=0;
        Task500mS();
    }
    if ((Task1SecFlag)==1)
    {
        Task1SecFlag=0;
        Task1Sec();
    }
}

```

Task.c 程式解說:

1. Task50Ms 為人工瀑布之控制，每 50ms 執行一次，判別人工瀑布開關 ON 或 OFF，開關 ON 啟動幫浦。
2. Task100Ms 為路燈之控制，每 100ms 執行一次，判別路燈開關 ON 並且偵測是否為晚上持續一分鐘，此兩個條件符合則路燈 ON。
3. Task500Ms 為灑水系統之控制，每 500ms 執行一次，當灑水系統開關 ON，5 分鐘灑水一次，一次持續 15 秒鐘。
4. Task1Sec 為步道燈系統之控制，每 1Sec 執行一次，當步道燈開關 ON 並且偵測是否為晚上一分鐘持續，此兩個條件符合，步道燈依照紅色、綠色、藍色、全亮、全滅五種動作閃爍之。

陸、研究結果



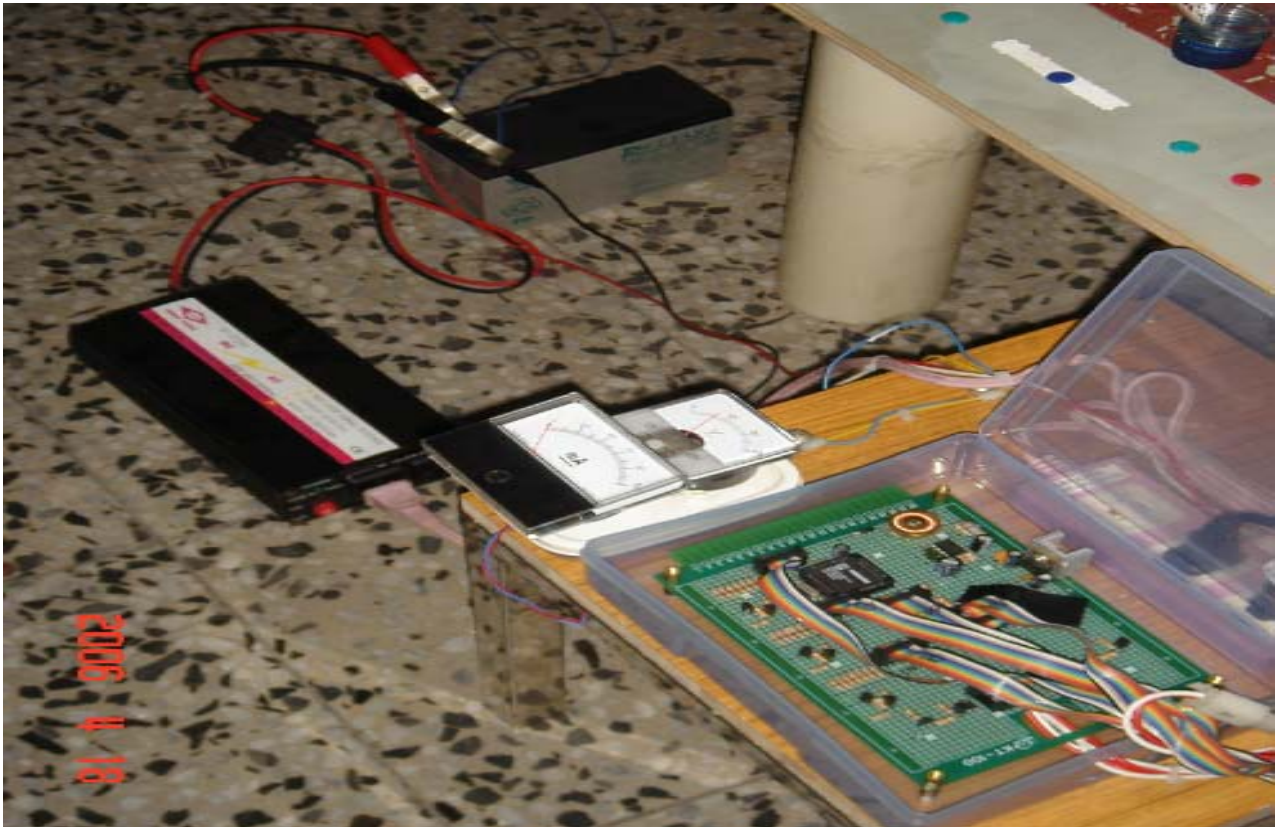
自然公園模型圖(夜)



自然公園模型圖(日)



控制電路實體圖



蓄電池及 DC/AC 電路實體圖

依上圖系統所示在測試單一種動作皆為正常，但是假如幫浦有動作卻發現單晶片有 (RESET) 現象。並且幫浦停止時，會造成路燈及步道燈動作不正常，有時是重新 (RESET)，有時卻是整個系統動作完全不正確。這是馬達及幫浦這種負載產生的突波問題，於是在馬達及幫浦加上突波吸收器才能解決這些問題，同時保護這自然公園模型系統。

綜合以上研究及其結果，吾人可得此節能式太陽能自然公園之可行性及實用性且本系統為一獨立型發電及供電系統，其擴充性廣泛、製作簡易、節能、無污染，對再生能源應用而言極具參考與發展價值，在此僅提供一己之力，期許在能源應用科技方面有所貢獻。

柒、 討論

用 keilC 來寫程式，有分時多工系統功能，可以依照自己所需來分割時間間隔即一種動作使用一種時間，時間一到單晶片就執行一次 Task，而且以飛利浦廠商提供的單晶片以及燒錄器使用之，只要在 Task.C 所分割好的時間內寫上動作就能完成，同時達到節能效果。

而照明路燈一隻使用 5 顆白光 LED 並聯，結果發現 LED 亮度不一，後來改成 5 顆串聯這問題就解決，不過所使用的電壓必須提昇至 16V，因此使用 MC34063 來達到提升電壓至 16V。而且當初電路只設計用一顆 npn 電晶體當作開關，發現只要一送電，所有燈光都會先亮一下，於是在輸出埠又加了一顆 pnp 電晶體解決這問題。

本分時多工系統功能亦可嘗試以 DSP 晶片來實現之並擴充更多功能與規劃及設計。

捌、 結論

本太陽能自然公園供電系統中，蓄電池電量用來提供以 LED 當作照明及步道燈所需電力已經綽綽有餘。但是基於目前以 LED 取代照明系統技術尚未成熟，白光 LED 亮度還有待加強，目前光電產業也正在研發製造高亮度 LED 來達到照明，業界預計在西元 2030 年前使用 LED 照明取代目前照明系統。

灑水器及人工瀑布部份，因為要驅動幫浦，因此所消耗的電力比較大，而使用太陽能發電充滿蓄電池，大約只能提供整個自然公園系統六小時之用電，尚嫌不足。在成本、經濟、設備衡量下，製作此一自然能源應用似乎有些局限與束縛，實際上在自然能源應用上尚有無限寬廣的擴充性及可行性只要資源足夠(含設備、腦力、物資、經費、時間、…等)，相信發展前景一片光明。

本研究即以此概念為基礎繁衍，藉此拋磚引玉，期望能有更多更好的技術研究出現，使其產生最佳的轉換效率，以利太陽能發電系統能順利推展應用。

玖、 參考資料及其他

- 一、林萬成。『擬蜂巢式太陽能電池模組化技術之研究』。國立成功大學碩士論文。
- 二、林萬成。『太陽能夢幻水族系統』。第四十四屆高雄市高中職組科學展。
- 三、賴麒文。2002。『C 與 8051 單晶片韌體設計－實務篇』。文魁資訊股份有限公司。
- 四、林邦傑。2001。『C/C++程式設計』。知城數位科技股份有限公司。
- 五、林銘波。2000。『數位系統設計』。全華科技圖書股份有限公司。
- 六、李克宇、黃新賢、陳瑞錡。2002。『8051/52 單晶片微電腦原理與應用』。知城數位科技股份有限公司。
- 七、蔡朝洋。2000。『單晶片微電腦 8051/8951 原理與應用』。全華科技圖書股份有限公司。

評語

091008 節能式太陽能自然公園

石油資源的大量使用，未來石油能源的匱乏是可見的，正因如此，其他能源的尋找與研發，更是重要。太陽能是重點之一。所以研究太陽能的方向是正確的，盼能繼續努力。

作品的重點，有點偏離。應朝太陽能板結構及安裝位置探討。或朝太陽能量取得後。思考如何以自創電路作更有效率的電力轉換研究。