UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA
Faculty of Engineering, Built Environment and Information Technology

# ENR 420

## RESEARCH PROJECT

### PRACTICAL 2 & 3: EXPERIMENTAL DESIGN AND RESULTS ANALYSIS

| Name and Surname | Student Number | Signature |
|---|---|---|
| Brighton Chikomo | 16071558 | |

By signing this assignment I confirm that I have read and are aware of the University of Pretoria's policy on academic dishonesty and plagiarism and I declare that the work submitted in this assignment is my own as delimited by the mentioned policies. I explicitly declare that no parts of this assignment have been copied from current or previous students' work or any other sources (including the internet), whether copyrighted or not. I understand that I will be subjected to disciplinary actions should it be found that the work I submit here does not comply with the said policies.

December 24, 2025

# ENR 420 practical Topic 1 Design and Analysis

Brighton Chikomo (16071558)

## I. Introduction

Photovoltaic-Diesel-Battery (PVDB) hybrid system is widely used to minimise the cost of energy. The PVDB hybrid system has a big advantage of reducing air pollution and global warming since there is a significant reduction in fossil fuel used [1]–[6]. If the PVDB system is optimized, a significant amount of energy will be conserved.

The optimisation of PVDB hybrid system was done in a journal article for practical one using the "quadprog" MATLAB function. The aim of this paper is to design the hardware to extract the practical results that will be compared to simulation results obtained in practical one. The obtained practical results will be simulated using the quadprog MATLAB function i.e. in the same manner as practical one. This will form practical three. Lastly, the obtained practical results will be compared to practical results to fully analysis whether the proposed solution to minimize the cost solution of PVDB hybrid system to people living in rural areas is feasible.

## II. Physical Implementation

To optimize the operation cost of the Photovoltaic-Diesel-Battery(PVDB) hybrid power system PV module, the battery module and the diesel generator where considered. The practical system designed makes use of a dsPIC30F4011 to control the ON times of the on solar and the battery.

$$P_1(t) - \text{control variable of energy flow from DG to the load at a given hour} \tag{1}$$

$$P_2(t) - \text{control variable of energy flow from PV to the load at a given hour} \tag{2}$$

$$P_3(t) - \text{control variable of energy flow from PV to battery at a given hour} \tag{3}$$

$$P_4(t) - \text{control variable of energy flow from battery to the load at a given hour} \tag{4}$$

$$P_L(t) - \text{load power measured after an hour} \tag{5}$$

$$P_{pv}(t) - \text{Photo-voltaic total power output.} \tag{6}$$

The following diagram shows the physical implementation of the designed system.

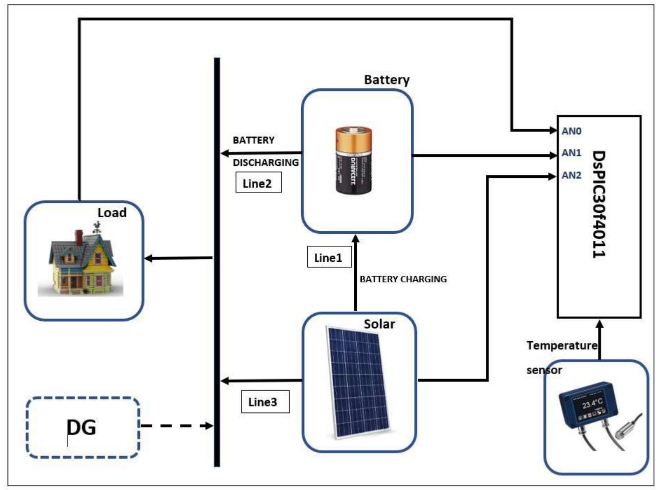



Fig. 1. Physical hardware implementation

### A. Detailed design

Fig. 2 - 4 below shows the complete connection of the overall designed hardware drawn using eagle auto-desk PCB design software.

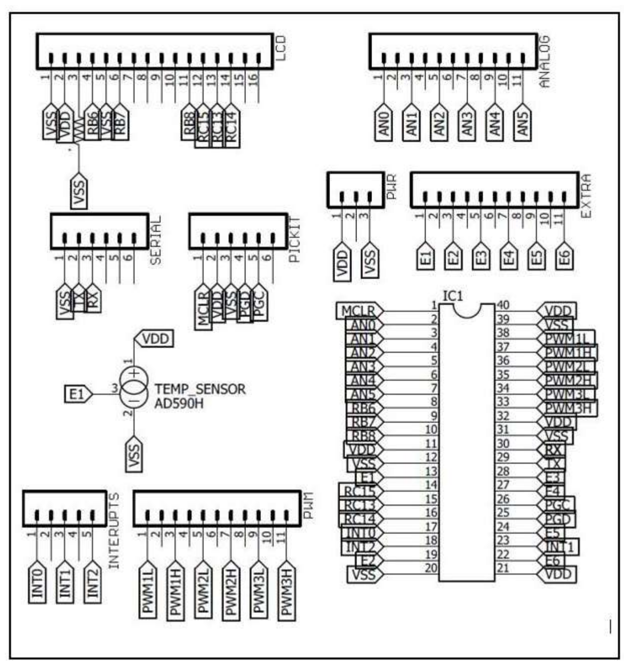



Fig. 2. Overal complete schematic of the micro-controller

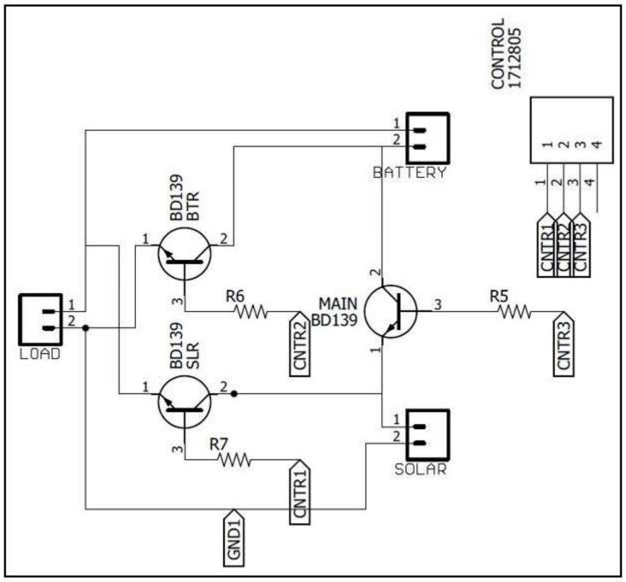



Fig. 3. Overal complete schematic of the control circuit

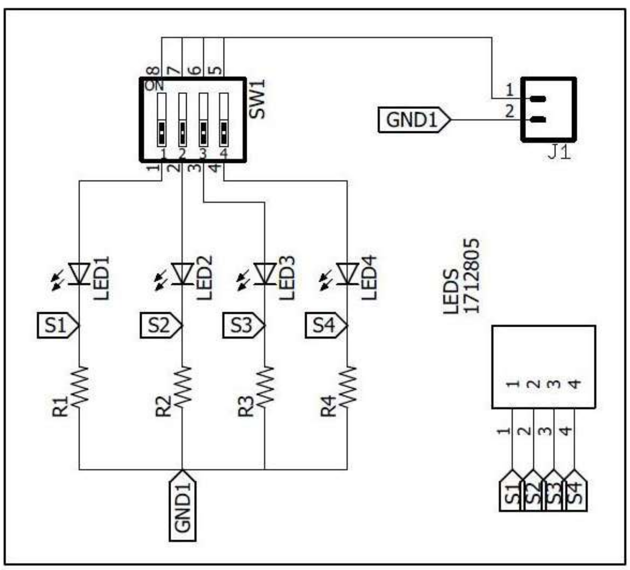



Fig. 4. Overal complete schematic of the load circuit

The PCBs for the circuit diagrams shown in fig.2 - 4 where etched in acid to form the traces to connect various components together. The etched PCBs are shown in appendix B.

## B. Components used and description

$dsPIC30F4011$ – to control the ON and OFF times of the PV and the battery load supply.
– to measure $P_2(t), P_3(t)$ and $P_4(t)$.

$Transistors$ – are electronic switches to switch the PV and the battery load supply.

$DIP\ switches$ – to vary the load hourly through switching different LED combinations

$LEDs$ – as load to be varied hourly using DIP switches

$Thermostat$ – as temperature sensor to obtain $T_A$ values

Fig. 5 shows the final circuit implementation

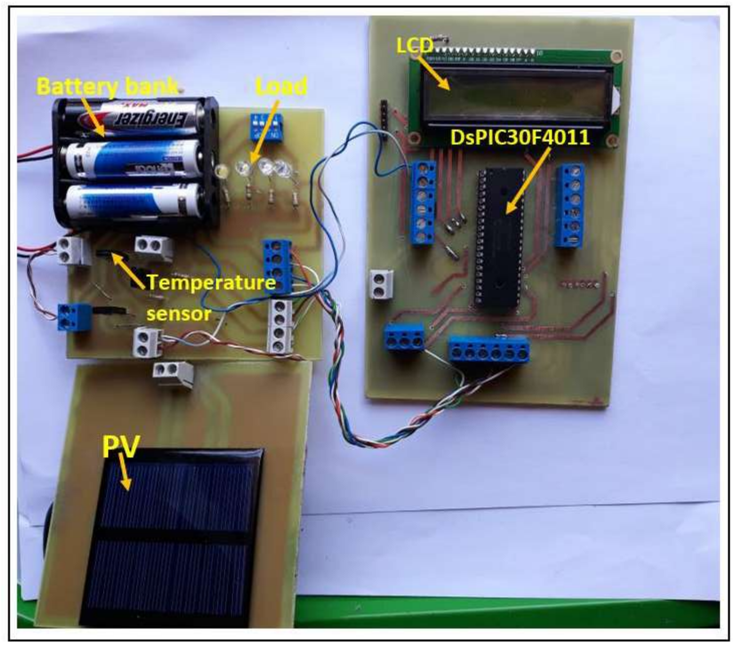



Fig. 5. Hardware of designed circuit

# III. Detailed design principles and design variable choice

Control variables $P_2(t)$, $P_3(t)$, $P_4(t)$, $P_L(t)$ and $P_{pv}(t)$ were physical implement in the designed system. These variables were chosen because they have a great impact on the optimisation of the PVDB hybrid system. Due to limited resources, the Diesel Generator (DG) power flow to the load (variable $P_1(t)$) shown in dashed lines in figure 1 was not designed. The values of the power flow from the DG to load were calculated from the equation below. The equation states that the power flow from the DG to the load is the shortfall power to meet the load demand.

$$P_1(t) + P_2(t) + P_4(t) = P_L(t) \quad (7)$$

## A. Design working principles

Analog channel AN0 on fig.1 measures the total load demand and it compares it with the power being supplied by the PV panel connected to analogue channel AN1. If:

1) AN0(total load power) $\leqslant$ AN2(PV power) the micro-controller turns ON the switch to charge the battery(line 1) and the switch for the solar to supply power to the load(line 3) as shown in fig.1. Upon writing the described working principle mathematically the following constraint was obtained

$$P_2(t) + P_3(t) \leqslant P_{pv}(t), \tag{8}$$

2) AN0(total load power) > AN1(PV power) the micro-controller turns ON the switch to discharge the battery(line 2), turns ON the switch for the solar to supply load and turns OFF the switch to charge the battery(line 1). Furthermore, if the PV and the battery can not meet the load demand, the DG(not implemented) turns ON. Assuming the DG power is the power required to compensate the demand when the PV and the battery are not able to meet the demand, the following constraint was obtained:

$$P_1(t) + P_2(t) + P_4(t) = P_L(t) \tag{9}$$

### B. Design Variable choice

To address the problem at hand, the designed hardware focus mainly on obtaining the values of $P_L(t)$ in constraint 8 and $P_{pv}$ in constraint 9.

### C. Determining $P_L(t)$ practically

The load was implemented with 4 LEDs of different colours and known current and voltage ratings. Dip switches were used to vary the load by switching ON and OFF different combination of the lead array as shown in fig.4. By varying the LED combination, the values of hourly load power,$P_L(t)$ for summer weekend days and weekdays were found and recorded in table II

Analog channel AN0 measures the load demand voltage. Since the load demand is known, the current flowing in the circuit(load current) is also known. The power values were calculated using equation 10 below

$$P_L(t) = I_L \times V_L \tag{10}$$

The LCD was used to display the measured hourly load power, $P_L(t)$ and PV total power output $P_{pv}(t)$. The total PV output power and hourly load power were recorded in table I and table II respectively.

### D. Determining $P_{pv}(t)$ practically

To find the power produced by the PV in equation 8, the following equations and constants in table III were used.

$$P_{pv} = \eta_{pv} A_c I_{pv} \tag{11}$$

$$\begin{aligned} where: \eta_R &= 15\% \\ A_C &= \text{PV surface area}(10cm \times 9.8cm) \\ &= 98cm^2 \\ &= 0.0098m^2 \end{aligned}$$

$$\eta_{pv} = \eta_R \left[ 1 - 0.9\beta \left( \frac{I_{pv}}{I_{pv,NT}} \right) (T_{C,NT} - T_{A,NT}) - \beta (T_A - T_R) \right] \tag{12}$$

To measure the efficiency of the PV, the hourly ambient temperature for the day and place where the practical was conducted was considered. The practical was held on the 3 October 2019 at Hillcrest, Pretoria. The ambient temperature($T_A$) in table I was measured using the temperature sensor connected to micro-controller for automatic recording of the hourly ambient temperature. The LCD display was used as the interface between the observer and the micro-controller. The LCD displayed the ambient temperature after every hour. The values of $T_A$ and calculated $p_{pv}(t)$ values are shown in table I.

TABLE I
MEASURED $P_{pv}(t)$ VALUES

| Time | $I_{pv}(Wh/m^2)$ | $T_A(C)$ | $\eta_{pv}$ | $P_{pv}(W)$ |
|---|---|---|---|---|
| 00:30 | 0.00 | 19.87 | 0.153078 | 0.000000 |
| 01:30 | 0.00 | 19.18 | 0.153492 | 0.000000 |
| 02:30 | 0.00 | 18.46 | 0.153924 | 0.000000 |
| 03:30 | 0.00 | 17.59 | 0.154460 | 0.000000 |
| 04:30 | 0.00 | 16.87 | 0.154878 | 0.000000 |
| 05:30 | 16.35 | 16.51 | 0.154818 | 0.099226 |
| 06:30 | 164.72 | 17.78 | 0.151552 | 0.97858 |
| 07:30 | 255.99 | 18.33 | 0.149682 | 1.502032 |
| 08:30 | 352.40 | 18.58 | 0.147905 | 2.043175 |
| 09:30 | 644.89 | 20.74 | 0.141673 | 3.581461 |
| 10:30 | 624.82 | 22.73 | 0.140818 | 3.449051 |
| 11:30 | 832.56 | 24.66 | 0.136155 | 4.443588 |
| 12:30 | 798.66 | 26.23 | 0.135785 | 4.251072 |
| 13:30 | 711.53 | 26.23 | 0.137255 | 3.828311 |
| 14:30 | 752.34 | 27.79 | 0.135630 | 3.999971 |
| 15:30 | 416.75 | 27.79 | 0.141293 | 2.308253 |
| 16:30 | 165.09 | 26.62 | 0.146242 | 0.946410 |
| 17:30 | 47.74 | 18.54 | 0.153070 | 0.286457 |
| 18:30 | 10.90 | 18.78 | 0.153548 | 0.065608 |
| 19:30 | 0.11 | 18.99 | 0.153604 | 0.000662 |
| 20:30 | 0.00 | 19.62 | 0.153228 | 0.000000 |
| 21:30 | 0.00 | 19.25 | 0.153450 | 0.000000 |
| 22:30 | 0.03 | 18.41 | 0.153953 | 0.000181 |
| 23:30 | 0.00 | 16.86 | 0.154884 | 0.000000 |

TableII show the results of the the recorded hourly load power

TABLE II
RECORDED WEEKDAY AND WEEKEND DEMAND PROFILES FOR SUMMER.

| Time | Summer load (W) | |
|---|---|---|
| | Weekend | Weekday |
| 00:30 | 0.9 | 0.9 |
| 01:30 | 0.9 | 0.9 |
| 02:30 | 1.11 | 1.11 |
| 03:30 | 1.17 | 1.17 |
| 04:30 | 1.11 | 1.11 |
| 05:30 | 0.90 | 0.9 |
| 06:30 | 0.99 | 0.89 |
| 07:30 | 0.99 | 0.95 |
| 08:30 | 1.02 | 1.01 |
| 09:30 | 1.05 | 1.02 |
| 10:30 | 1.05 | 1.01 |
| 11:30 | 1.05 | 1.00 |
| 12:30 | 0.75 | 0.75 |
| 13:30 | 0.79 | 0.79 |
| 14:30 | 0.81 | 0.81 |
| 15:30 | 0.81 | 0.81 |
| 16:30 | 0.87 | 0.87 |
| 17:30 | 1.26 | 1.29 |
| 18:30 | 1.44 | 1.39 |
| 19:30 | 2.28 | 1.95 |
| 20:30 | 2.28 | 1.95 |
| 21:30 | 1.20 | 1.20 |
| 22:30 | 1.17 | 1.17 |
| 23:30 | 0.99 | 0.99 |

The following tables show the practical values of the battery constants and other variables for the designed hardware. These values were obtained from the datasheets of the components used to build the hardware.

TABLE III
USEFUL CONSTANTS TO SOLVE THE OPTIMIZATION PROBLEM

| Parameter | Value |
|---|---|
| Nominal battery capacity, $B_C^{max}$ | $4.45Wh$ |
| Battery charging efficiency, $\eta_C$ | 80% |
| Battery discharging efficiency, $\eta_D$ | 50% |
| Battery allowable depth of discharge, DOD | 50% |
| Diesel generator capacity, S | $0.5VA$ |
| PV array capacity | $3.5W$ |
| a | $US\$0.246/h$ |
| b | $US\$0.1/Wh$ |
| Fuel Cost, $C_f$ | $US\$1.2/l$ |
| PV array area, $A_c$ | $0.0098m^2$ |
| PV generator efficiency, $\eta_R$ | 15% |

## IV. SIMULATIONS FOR PRACTICAL 3

### A. Analysis of the data collected in practical 2

In this section, the approach used in practical 1 simulations is going to be used to form all the required matrix to simulate the practical 2 results. The model to be optimized consists of objective function, constraints and design variable. .The objective function is to minimise [2]:

$$min\ C_f \sum_{t=1}^{N} \left(aP_1^2(t) + bP_1(t)\right) \tag{13}$$

subject to the constraints

$$P_2(t) + P_3(t) \leqslant P_{pv}(t), \tag{14}$$

$$P_1(t) + P_2(t) + P_4(t) = P_L(t), \tag{15}$$

$$P_1(t) \geqslant 0,\ P_2(t) \geqslant 0,\ P_3(t) \geqslant 0,\ P_4(t) \geqslant 0, \tag{16}$$

$$P_i^{min} \leqslant P_i(t) \leqslant P_i^{max}, \tag{17}$$

$$B_c^{min} \leqslant B_C(0) + \eta_C \sum_{\tau=1}^{t} P_3(\tau) - \eta_D \sum_{\tau=1}^{t} P_4(\tau) \leqslant B_C^{max}, \tag{18}$$

The design variables of this optimization problem are $P_1(t)$, $P_2(t)$, $P_3(t)$, and $P_4(t)$, respectively.

## V. "QUADPROG" MATLAB FUNCTION

The objective function to be minimized fall under non-linear or quadratic programming problems. Therefore the "quadprog" MATLAB function is to be used as a tool to solve or simulate the optimization problem formulated from practical 2 results [2]. The MATLAB quadprog function is given by equation 19 and 20 below.

$$min \frac{1}{2} x^T H x + f^T x \tag{19}$$

subject to constraints:

$$\begin{aligned} Ax &\leqslant b \\ A_{eq} &= b_{eq} \\ lb \leqslant x &\leqslant ub \end{aligned} \tag{20}$$

$$\begin{aligned} where:\ H &- \text{Hessian matrix} \\ x &- \text{control variables} \\ f &- \text{vector matrix} \\ A,\ b &- \text{Inequality constraints matrix} \\ A_{eq},\ b_{eq} &- \text{equality constraints matrix} \end{aligned}$$

To solve the optimization problem the suggested objective function and the constraints are transformed into the MATLAB quadprg form shown in equation 19 and equation 20 respectively. The following subsections describe the transformation of the suggested energy optimization problem into a MATLAB quadprog form.

### A. Transformation of the objective function to MATLAB quadprop form

Equation 13 is going to be transformed into equation 19 through the formulation of matrix H and matrix f.

Matrix H is a $96 \times 96$ matrix with:

1) diagonal entries of $2 \times C_f \times a$ from $H_{1,1}$ to $H_{24,24}$
2) All other entries are zeros

The framework of the H matrix is shown in matrix array 21

$$H = \begin{bmatrix} H_{1,1} & H_{1,2} & H_{1,3} & .. & .. & .. & H_{1,96} \\ H_{2,1} & H_{2,2} & H_{2,3} & .. & .. & .. & .. \\ H_{3,1} & H_{3,2} & H_{3,3} & .. & .. & .. & .. \\ .. & .. & .. & .. & .. & .. & .. \\ .. & .. & .. & .. & .. & .. & .. \\ .. & .. & .. & .. & .. & .. & .. \\ H_{96,1} & .. & .. & .. & .. & .. & .. \end{bmatrix} \tag{21}$$

Matrix f is a $96 \times 1$ matrix with:

1) First 24 row entries of $C_f \times b$
2) All other entries are zeros

The framework of matrix f is shown in matrix array 22

$$f = \begin{bmatrix} c_f \times b \\ c_f \times b \\ c_f \times b \\ .. \\ .. \\ .. \\ 0 \end{bmatrix} \tag{22}$$

*B. Transformation of inequality constraints into MATLAB quadprog form*

The inequality constraints in equation 14 and equation 18 are going to be transformed into first equation of equation array 20 to form the matrix A and matrix b as defined in equation 20. Matrix A is given by the coefficients of inequality constraint control variables ($P_2(t)$, $P_3(t)$ from equation 14 and equation 18. To fomulate matrix A three matrix namely $A_a$, $A_b$ and $A_c$ are going to be considered as sub matrix of A.

Matrix $A_a$ is found from equation 14 as a $24 \times 96$ matrix with:

1) Two diagonal entries with ones from $A_{a1,25}$ to $A_{a24,48}$ and $A_{a1,49}$ to $A_{a24,72}$
2) All other entries are zeros

Matrix $A_b$ and $A_c$ are found from equation 18 as $1 \times 96$ matrices with: Equation 18 is separated into two parts to form $A_b$ and $A_c$

$$\begin{aligned} A_b &: \eta_C \sum_{\tau=1}^{t} P_3(\tau) - \eta_D \sum_{\tau=1}^{t} P_4(\tau) \leqslant B_C(0) - B_c^{max} \\ A_c &: -\eta_C \sum_{\tau=1}^{t} P_3(\tau) + \eta_D \sum_{\tau=1}^{t} P_4(\tau) \leqslant B_C(0) - B_c^{min} \end{aligned} \tag{23}$$

$$where: \quad A_b = -A_c$$

$$\eta_C,\ \eta_D,\ B_C^{max},\ DOB \text{ are given in table III}$$

$$B_C^{min} = (1 - DOB) B_C^{max} \tag{24}$$

$$B_C^{min} \leqslant B_c(t) \leqslant B_C^{max} \tag{25}$$

Substituting values of DOB and $B_C^{max}$ in equation 25, $B_C^{min}$ was found to be 2.225Wh. After substituting constants from table III into equation 23 the following equations were obtained.

$$\begin{aligned} A_b &: 0.80 \sum_{\tau=1}^{t} P_3(\tau) - \sum_{\tau=1}^{t} P_4(\tau) \leqslant 0.255 \\ A_c &: -0.80 \sum_{\tau=1}^{t} P_3(\tau) + \sum_{\tau=1}^{t} P_4(\tau) \leqslant 0 \end{aligned} \tag{26}$$

From equation 26 matrix $A_b$ can be found to be a $1 \times 96$ matrix with:

1) entries of 0.80 from entry $A_{a49,72}$ to $A_{a73,96}$
2) All other entries are zeros

To find $A_c$ use equation 27 below

$$A_b = -A_c \tag{27}$$

matrix A is a $26 \times 96$ matrix with:

1) $24 \times 96$ top matrix as $A_a$
2) $1 \times 96$ matrix at row 25 as $A_b$
3) $1 \times 96$ matrix at row 26 as $A_c$

The frame work of the matrix A is shown in matrix array 28

$$A = \begin{bmatrix} A_a(24 \times 96) \\ A_b(1 \times 96) \\ A_c(1 \times 96) \end{bmatrix} \tag{28}$$

Matrix b is obtained from the coefficients of $P_{pv}(t)$ and the values $(B_c 0 - B_C^{min})$ and $(B_c^{max} - B_C(0))$ shown in equation 23. Using equation 26 $b_b$ is 2.225 and $b_c$ is 0. Matrix $b_a$ are coefficients of $P_{pv}(t)$. The values of $P_{pv}$ were calculated in practical 2 above to as shown in table I.

*C. Transformation of equality constraints into MATLAB quadprog form*

The equality constraints in equation 9 is going to be transformed into second equation of equation array 20 to form matrix $A_{eq}$ and matrix $b_{eq}$ as defined in equation 20. Matrix $A_{eq}$ is given by the coefficients of equality constraint control variables $(P_1(t), P_2(t) and P_4(t))$ from equation 9. Matrix $b_{eq}$ is given by coefficients of $P_L(t)$ in equation 15.

Matrix $A_{eq}$ is a $24 \times 96$ matrix with:

1) Three diagonal entries with ones from $A_{eq_{1,1}}$ to $A_{eq_{24,24}}$, $A_{eq_{1,25}}$ to $A_{eq_{24,48}}$ and $A_{eq_{1,73}}$ to $A_{eq_{24,96}}$
2) All other entries are zeros

Matrix $b_{eq}$ is a $24 \times 1$ matrix with entries $P_L(t)$ as obtained in practical 2. $P_L(t)$ are shown in table II.

*D. Boundaries*

The upper boundaries of the practical 3 simulations are less than the boundaries from the practical 1 since a small PV and battery was used to build the hardware. The PV and the battery capacity used in the design are as shown in table III. The boundaries for this optimization problem are as follows: The UB(Upper boundary) is a $96 \times 1$ matrix which is given by the capacities of the DG, PV and battery respectively.

UB matrix is a $96 \times 1$ matrix with:

1) entries 1 to 24 as DG capacity
2) entries 25 to 72 as PV capacity
3) entries 73 to 96 as battery capacity

The frame work of matrix UB is shown in matrix 30

$$UB = \begin{bmatrix} \text{DG capacity (0.5VA)} \\ \text{PV capacity (3.5W)} \\ \text{Battery capacity (4.45Wh)} \end{bmatrix} \tag{29}$$

LB(Lower boundary) matrix is given by equation **??** and **??**.

LB is a is a $96 \times 1$ matrix with all entries as zeros.

The frame work of matrix UB is shown in matrix 30

$$LB = \begin{bmatrix} 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{bmatrix} \tag{30}$$

### E. Optimization problem solution

The quadprog function of MATLAB was used to solve the energy optimization problem described by the matrix above. Only data for summer weekends and weekdays(Table **??**) was analysed and plotted as shown in the results section.

## VI. Results

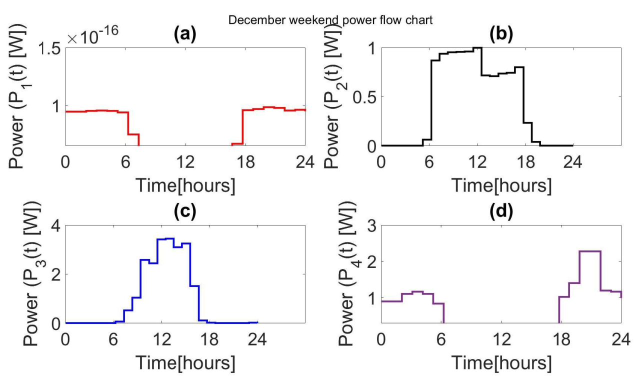



Fig. 6. Summer weekend power flow.

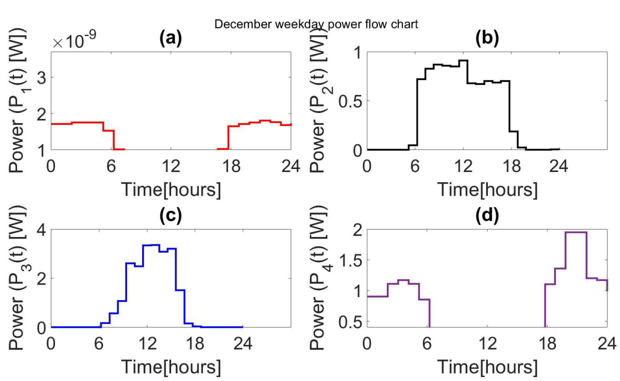



Fig. 7. Summer weekday power flow

## VII. Results Analysis

### A. Comparisons of Practical one results and practical 3 results

For practical 3 the power flow from the battery to the load is much smaller than the value obtained in practical 1. This is because the battery bank which was used to design the hardware for practical 3 has a very small capacity as compared to the one which was used to do practical 1 simulations

Although the wave-forms for the power flow from the PV to the battery bank are similar, the amount of power following from the PV to the battery bank is much much smaller for practical 3 as compared to the one for practical 1. This is because a PV which was used to implement the hardware for practical 3 is very small as compared to the PV capacity which was used in practical 1.

The power flow from the PV to the load for practical 3 is smaller than that of practical 1. This is because the surface area and the power ratings of the PV used for practical 3 is smaller as compared to the one which was used for practical 1

The PV to load power flow wave-forms for Practical 3 are slightly different from those in practical one. This is because the ambient temperatures obtained in practical 3 were slightly less than the ambient temperatures for practical 1 simulations.

Lastly, the power flow from the DG to the load for practical 3 is small than that of practical 1. This is because the assumed DG capacity for practical 3 is smaller than the one which was used in practical 1 simulations.

In summary, the wave-forms for practical 1 summer power flows are identical to practical 3 wave-forms. The only different of that the waveforms obtained in practical 3 have has power values measured in watts while the practical 1 wave-forms have power values recorded in kilo-watts. This is because the designed hardware has a small solar panel and small battery bank capacities as shown in table III.

### B. General comments to practical 3 wave-forms obtained

Fig. 6 and 7 show the flow of energy for 24hrs during the summer weekdays and weekends respectively. From time 0:00 to approximately 08:00 and from 18:30 to 0:00 the load demand is met by the DG and/or the battery bank. At these times, the DG switches ON only when the PV and/or the battery is not able to meet the load demand

Fig.6(c) and (d) and Fig.7(c) and (d) shows the charging and discharging processes of the battery bank respectively. The battery bank charges during the day and discharge (supply energy to the load) during the night when the PV can not provide enough energy to the load.

PV, battery bank and DG provides energy to the load during the early hours of the day. The DG switches ON only when the PV and/or the battery bank is not able to provide enough energy to the load. Whenever the PV and/or the battery bank is able to meet demand, the DG switches OFF to reduce the fuel costs and hence operation cost of the PVDB system. The ON time of the DG depends on the battery's state of charge(SOC) and the energy the PV array can supply(size of the PV array). The DG is ON for more hours if the PV and/or the battery bank provides less energy.

From the two sets of graphs(weekend and weekdays energy flow plots) in Fig.6 and Fig.7 it can be deduced that the energy demand is higher during the weekend than during the weekdays. This is because during the week people living in remote areas will be busy in the fields and with other different outside home activities but during the weekend they will be at home using various electrical appliances.

From plots in Fig.6 and Fig.7, the fuel cost saving can be deduced and the total cost to the proposed PVDB system can be obtained. If winter graphs where plotted, the seasonal variation in PV energy supply to the load is deduced. In general, the PV supplies more energy in summer than in winter hence the DG runs for a short time in summer than in winter.

In this optimization model, the battery is only charged by the PV and the DG only supplies energy to the load when it is switched ON. This model configuration ensures low operation cost since the DG ON time is limited. For this optimization energy problem, the optimization solution is obtained by computing the difference between the DG operating cost(fuel cost) for a stand-alone DG system and the PVDB hybrid system. The practical simulated results show that PVDB system achieves more saving than the stand-alone DG. This indicates the proposed optimization model has the capacity to reduce the operation cost for the PVDB system.

## VIII. Conclusion

The physical hardware was designed and the results of $P_L(t)$ and $T_A$ where obtained. The obtained results were simulated using the quadprog MATLAB function and plotted as shown in Fig.6 and Fig.7. Only the results for December weekends and weekdays were considered. Practical 1 results were then compared to practical 3 results to see whether the proposed solution can optimise the operation cost of the PVDB hybrid system. The proposed solution to minimum cost solution to PVDB hybrid can be deemed feasible since practical 1 simulations and the physical designed system results have similar optimised solution to the problem. The proposed optimization model has the capacity to reduce the operation cost for the PVDB system since it reduces the ON time of the DG.

## References


[1] K. Kusakana, "Operation cost minimization of photovoltaic-diesel-battery hybrid systems," *Energy*, vol. 85, pp. 645–653, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.energy.2015.04.002

[2] H. Tazvinga, X. Xia, and J. Zhang, "Author ' s personal copy ScienceDirect Minimum cost solution of photovoltaic â" diesel â" battery hybrid power systems for remote consumers Author ' s personal copy," vol. 96, pp. 292–299, 2013.

[3] A. E.-S. A. Nafeh, "Fuzzy Logic Operation Control for PV-Diesel-Battery Hybrid Energy System," *The Open Renewable Energy Journal*, vol. 2, no. 1, pp. 70–78, 2009.

[4] S. B. Jeyaprabha and A. I. Selvakumar, "Optimal sizing of photovoltaic/battery/diesel based hybrid system and optimal tilting of solar array using the artificial intelligence for remote houses in India," *Energy and Buildings*, vol. 96, pp. 40–52, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.enbuild.2015.03.012

[5] F. Fodhil, A. Hamidat, and O. Nadjemi, "Potential, optimization and sensitivity analysis of photovoltaic-diesel-battery hybrid energy system for rural electrification in Algeria," *Energy*, vol. 169, pp. 613–624, 2019. [Online]. Available: https://doi.org/10.1016/j.energy.2018.12.049

[6] K. Karakoulidis, K. Mavridis, D. V. Bandekas, P. Adoniadis, C. Potolias, and N. Vordos, "Techno-economic analysis of a stand-alone hybrid photovoltaic-diesel-battery-fuel cell power system," *Renewable Energy*, vol. 36, no. 8, pp. 2238–2244, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.renene.2010.12.003

# IX. Appendix A: ELO 5 - Engineering methods, skills and tools

## A. Application of appropriate engineering methods

To minimize the operation cost of the PVBD hybrid system, the energy optimization problem was formulated. The constraints of the optimization problem were formulated from the functionality of the PVDB hybrid system using the power flows from the PV, battery bank and/or the DG. The objective function of the proposed solution was formulated using the cost of the fuel. This is because to minimize the operation cost of the system one has to minimize the fuel cost of running the DG.

Since the obtained energy optimization problem is complex quadratic which is very difficult to solve manually, quadprog MATLAB function was used as a tool to find the optimal solution to the problem.

## B. Using appropriate engineering skills and tools.

To be able to use the MATLAB quadprog function, the energy optimization problem was transformed into a quadprog function form simply by forming different matrix equations which correspond to the quadprog from.

The MATLAB quadprog function was used because it has the following advantages:

1) The optimization problem is a complex quadratic problem which can be best solved using MATLAB quadprog function
2) The quadprog MATLAB function is very easy to use since it allows applying the optimization problem at hand directly into MATLAB terminal and solves it in one click.
3) MATLAB codes are very easy to debug for errors as it indicates the line code with errors and gives a hint of the error type.

## C. Assessment of outcome from engineering methods, skills and tools.

To be able to assess and analyse the obtained results, The power flow from the PV, battery bank and DG to the load and the power flow from the PV to the battery vs time graphs were plotted in matlab.

One of the main reasons why MATLAB was chosen to use as a tool to solve the optimization problem at hand is that it makes the analysis of the results very easy. This is because it allows plotting the results as graphs. Graphs are easy to analyse the validity and the meaning of the solution as it offers a visual solution which can be analyzed in a short time.

The results of this problem are based on the shape and amplitude of the MATLAB plots obtained. Practical 1 and practical 3 plots are very similar in shape. The only difference in the results is that for practical one the designed hardware implemented the real practical problem with smaller power values. To sum it up all, it can be said that practical 3 results are a scaled version of practical 1 results.

## X. Appendix A: Etched PCBs

This section shows pictures of the actually PCBs that were used to implement the designed hardware.

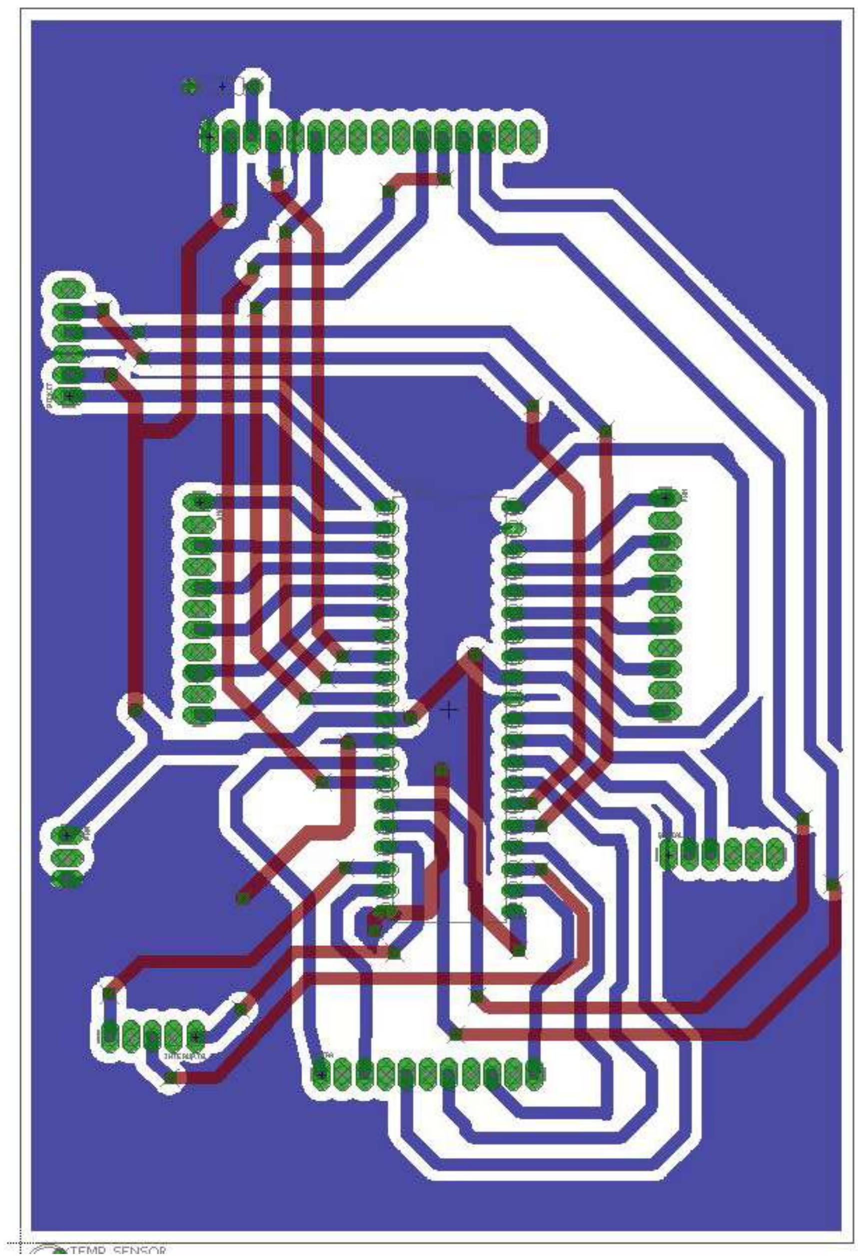



Fig. 8. Micro-controller and LCD PCB board

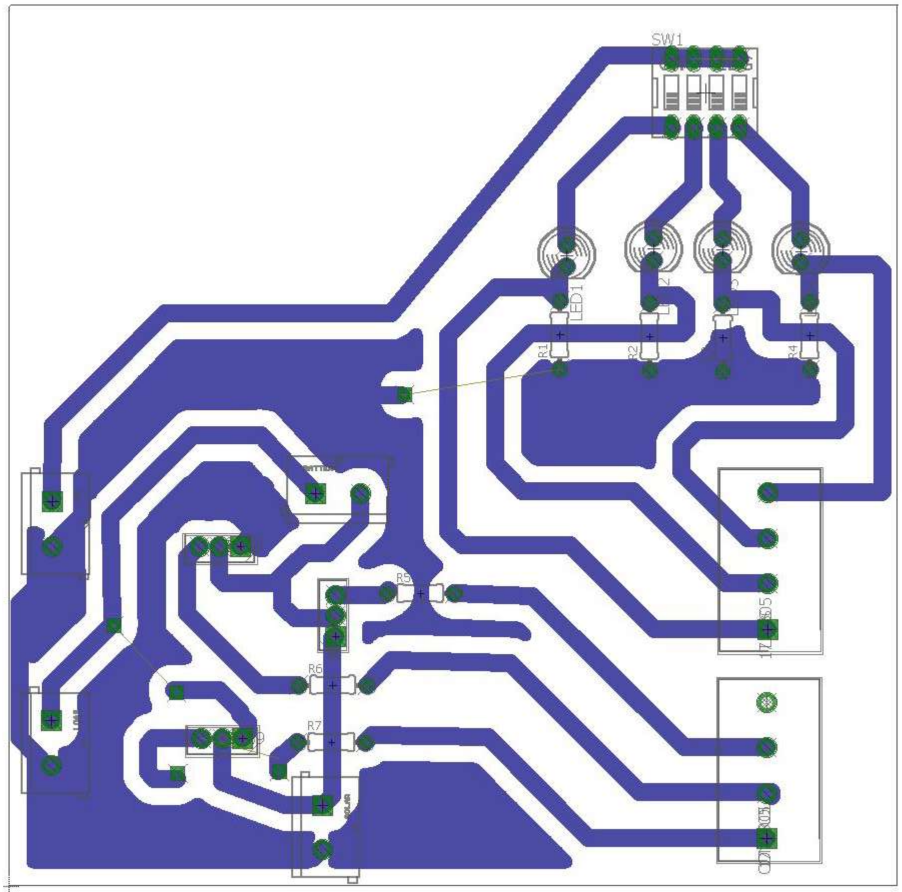



Fig. 9. Load, temperature sensor and battery bank PCB board

# XI. Appendix B:C Code and MATLAB Code

The following code was used to optimize the energy problem in MATLAB.

```
%-----------------------------------------------------------------------
%BRIGHTON S CHIKOMO
%ENR PRACTICAL PRACTICAL 2 FOR TOPIC 1
%MINIMUM COST SOLUTION
%LAST EDITED: 08/10/2019     TIME:0133hrs
%-----------------------------------------------------------------------

%variables
cf = 1.2;
a = 0.246;
b = 0.1;

%--------- THE HESSIAN MATRIX FORMULATION------------

%Big matrix 96*96
big_H = zeros(96);

%creating small matrix to fit in a big matrix
sm1_H = 2*cf*a*ones(1,24);
small_H = diag(sm1_H);


% Specify upper left row, column of where
% we'd like to paste the small matrix.
row1_H = 1;
column1_H = 1;
% Determine lower right location.
row2_H = 24;
column2_H = 24;


% It will fit, so paste it.
big_H(row1_H:row2_H, column1_H:column2_H) = small_H;
H = big_H;
%----------------END OF HESSIAM MATRIX FORMULATION----------------

%--------- THE f MATRIX FORMULATION------------


big_f = zeros(96,1);
%creating small matrix to fit in a big matrix
small_f = b*cf*ones(24,1);


% Specify upper left row, column of where
% we'd like to paste the small matrix.
row1_f = 1;
column1_f = 1;

% Determine lower right location.
row2_f = 24;
column2_f = 1;

% It will fit, so paste it.
```

```
big_f(row1_f:row2_f, column1_f:column2_f) = small_f;
f = big_f;

%----------------END OF f MATRIX FORMULATION----------------


%--------- THE Aeq MATRIX FORMULATION------------

big_Aeq = zeros(24,96);
%creating small matrix to fit in a big matrix
sm_Aeq = ones(1,24);
small_Aeq = diag(sm_Aeq);

% Specify upper left row, column of where
% we'd like to paste the small matrix.
row1_Aeq1 = 1;
column1_Aeq1 = 1;
% Determine lower right location.
row2_Aeq1 = 24
column2_Aeq1 = 24

% Specify upper left row, column of where
% we'd like to paste the small matrix.
row1_Aeq2 = 1;
column1_Aeq2 = 25;
% Determine lower right location.
row2_Aeq2 = 24
column2_Aeq2 = 48

% Specify upper left row, column of where
% we'd like to paste the small matrix.
row1_Aeq3 = 1;
column1_Aeq3 = 73;
% Determine lower right location.
row2_Aeq3 = 24
column2_Aeq3 = 96

% It will fit, so paste it.
big_Aeq(row1_Aeq1:row2_Aeq1, column1_Aeq1:column2_Aeq1) = small_Aeq;
big_Aeq(row1_Aeq2:row2_Aeq2, column1_Aeq2:column2_Aeq2) = small_Aeq;
big_Aeq(row1_Aeq3:row2_Aeq3, column1_Aeq3:column2_Aeq3) = small_Aeq;
Aeq = big_Aeq;
%----------------END OF f MATRIX FORMULATION----------------

%--------- THE A MATRIX FORMULATION------------

big_A = zeros(26,96);
%creating small matrix to fit in a big matrix
sm_A1 = ones(1,24);
small_A1 = diag(sm_A1);

sm_A2 = -1*sm_A1;
sm_A3 = 0.80*ones(1,24);
sm_A4 = -0.80*ones(1,24);



% Specify upper left row, column of where
```

```
% we'd like to paste the small matrix.
row1_A2 = 1;
column1_A2 = 25;
% Determine lower right location.
row2_A2 = 24;
column2_A2 = 48;

% Specify upper left row, column of where
% we'd like to paste the small matrix.
row1_A3 = 1;
column1_A3 = 49;
% Determine lower right location.
row2_A3 = 24;
column2_A3 = 72;

%matrix fitting
row1_A4 = 25;
column1_A4 = 49;
row2_A4 = 25;
column2_A4 = 72;

%matrix fitting
row1_A5 = 25;
column1_A5 = 73;
row2_A5 = 25;
column2_A5 = 96;

%matrix fitting
row1_A6 = 26;

% It will fit, so paste it.
big_A(row1_A2:row2_A2, column1_A2:column2_A2) = small_A1;
big_A(row1_A3:row2_A3, column1_A3:column2_A3) = small_A1;
big_A(row1_A4:row2_A4, column1_A4:column2_A4) = sm_A3;
big_A(row1_A5:row2_A5, column1_A5:column2_A5) = sm_A2;
big_A(row1_A6:row1_A6, column1_A4:column2_A4) = sm_A4;
big_A(row1_A6:row1_A6, column1_A5:column2_A5) = sm_A1;
A = big_A;
%----------------END OF A MATRIX FORMULATION----------------


%--------- THE b_eq MATRIX FORMULATION------------

%winter weekend b_eq

beq1 = [0.9; 0.9; 1.11; 1.17; 1.11; 0.9; 0.99; 0.99; 1.02;
      1.05; 1.05; 1.05; 0.75; 0.792; 0.81; 0.81; 0.87; 1.26;
      1.44; 2.28; 2.28; 1.2; 1.17;0.99]; %summer weekend

beq = [0.9; 0.9; 1.11; 1.17; 1.11; 0.9; 0.89; 0.95; 1.01;
        1.02; 1.01; 1.00; 0.75; 0.792; 0.81; 0.81;
        0.87; 1.29; 1.386; 1.95; 1.95; 1.2; 1.17; 0.99]; %summer weekday




%----------------END OF b_eq MATRIX FORMULATION--------------
```

```
%---------- B MATRIX FORMULATION-------------


B= [0; 0; 0; 0; 0; 0.099226013; 0.978577161; 1.502031824;
    2.043174956; 3.581461404; 3.449051368; 4.44358782; 4.251072954;
    3.828311248; 3.99997081; 2.30825284; 0.946409885; 0.286457148;
    0.065608016; 0.000662341; 0; 0; 0.000181049; 0; 2.225; 0];





%----------------END OF B MATRIX FORMULATION-----------------

%---------- BOUNDARY MATRIX FORMULATION-------------

%Bounderies
%1) lower boundary
lb = zeros(96,1);

%2) upper boundary
UB_big = zeros(96,1);
UB_sm1 = 0.5*ones(24,1);
UB_sm2 = 3.5*ones(48,1);
UB_sm3 = 4.45*ones(24,1);

row1_ub1 = 1;
col_ub = 1;
row1_ub2 = 24;

row2_ub1 = 25;
row2_ub2 = 72;

row3_ub1 = 73;
row3_ub2 = 96;

UB_big( row1_ub1:row1_ub2, col_ub:col_ub)= UB_sm1;
UB_big( row2_ub1:row2_ub2, col_ub:col_ub)= UB_sm2;
UB_big( row3_ub1:row3_ub2, col_ub:col_ub)= UB_sm3;
ub = UB_big;
%----------END OF BOUNDARY MATRIX FORMULATION-------------





%------------------MAIN CODE TO PLOT---------------------------
[x,fval] = quadprog(H,f,A,B,Aeq,beq,lb,ub);

t = linspace(0,24,24);
%t = hours(1):hours(1):hours(24);

p1= x(1:24,1);
p2= x(25:48,1);
p3= x(49:72,1);
p4= x(73:96,1);

```

```
p_1 = transpose(p1);
p_2 = transpose(p2);
p_3 = transpose(p3);
p_4 = transpose(p4);



h = suptitle({'December weekday power flow chart',''})
%h = suptitle({'December weekend power flow chart',''})
set(h,'FontSize',25);


s(1)=subplot(2,2,1);
h1 = stairs(s(1),t,p_1);
h1.Color = 'red';
h1.LineWidth = 3.5;
title('(a)', 'FontSize', 15);
hy1 = ylabel('Power (P_1(t) [W])');
hx1 = xlabel('Time[hours]');
hy1.FontSize = 20;
hx1.FontSize = 20;
ax = gca;
ax.FontSize = 20;


s(2)=subplot(2,2,2);
h2 = stairs(s(2),t,p_2);
h2.Color = 'black';
h2.LineWidth = 3.5;
title('(b)', 'FontSize', 15);
hy2 = ylabel('Power (P_2(t) [W])');
hx2 = xlabel('Time[hours]');
hy2.FontSize = 20;
hx2.FontSize = 20;
ax = gca;
ax.FontSize = 20;


s(3)=subplot(2,2,3);
h3 = stairs(s(3),t,p_3);
h3.Color = 'blue';
h3.LineWidth = 3.5;
title('(c)', 'FontSize', 15);
hy3 = ylabel('Power (P_3(t) [W])');
hx3 = xlabel('Time[hours]');
hy3.FontSize = 20;
hx3.FontSize = 20;
ax = gca;
ax.FontSize = 20;


s(4)=subplot(2,2,4);
h4 = stairs(s(4),t,p_4);
h4.Color = [0.4940, 0.1840, 0.5560];% deep purple colour
h4.LineWidth = 3.5;
title('(d)', 'FontSize', 15);
hy4 = ylabel('Power (P_4(t) [W])');
hx4 = xlabel('Time[hours]');
```

```
hy4.FontSize = 20;
hx4.FontSize = 20;
ax = gca;
ax.FontSize = 20;

%axis settings for summer weekdays
axis([s(1)],[0 24 1e-9 3.7e-9]);
axis([s(4)],[0 24 0.4 2]);


%axis settings for summer weekends
%axis([s(1)],[0 24 0.653e-16 1.5e-16]);
%axis([s(4)],[0 24 0.3 3]);



%axis([s(2) s(3) s(4)],[0 24 0 4]);
xticks([s(1) s(2) s(3) s(4)],[0 6 12 18 24]);
%yticks([s(2) s(3) s(4)],[0 2 4]);
ax = gca;
ax.FontSize = 20;

%---------------THE END-----------------------------
```

The following code was used to get the ambient temperature values from the senor

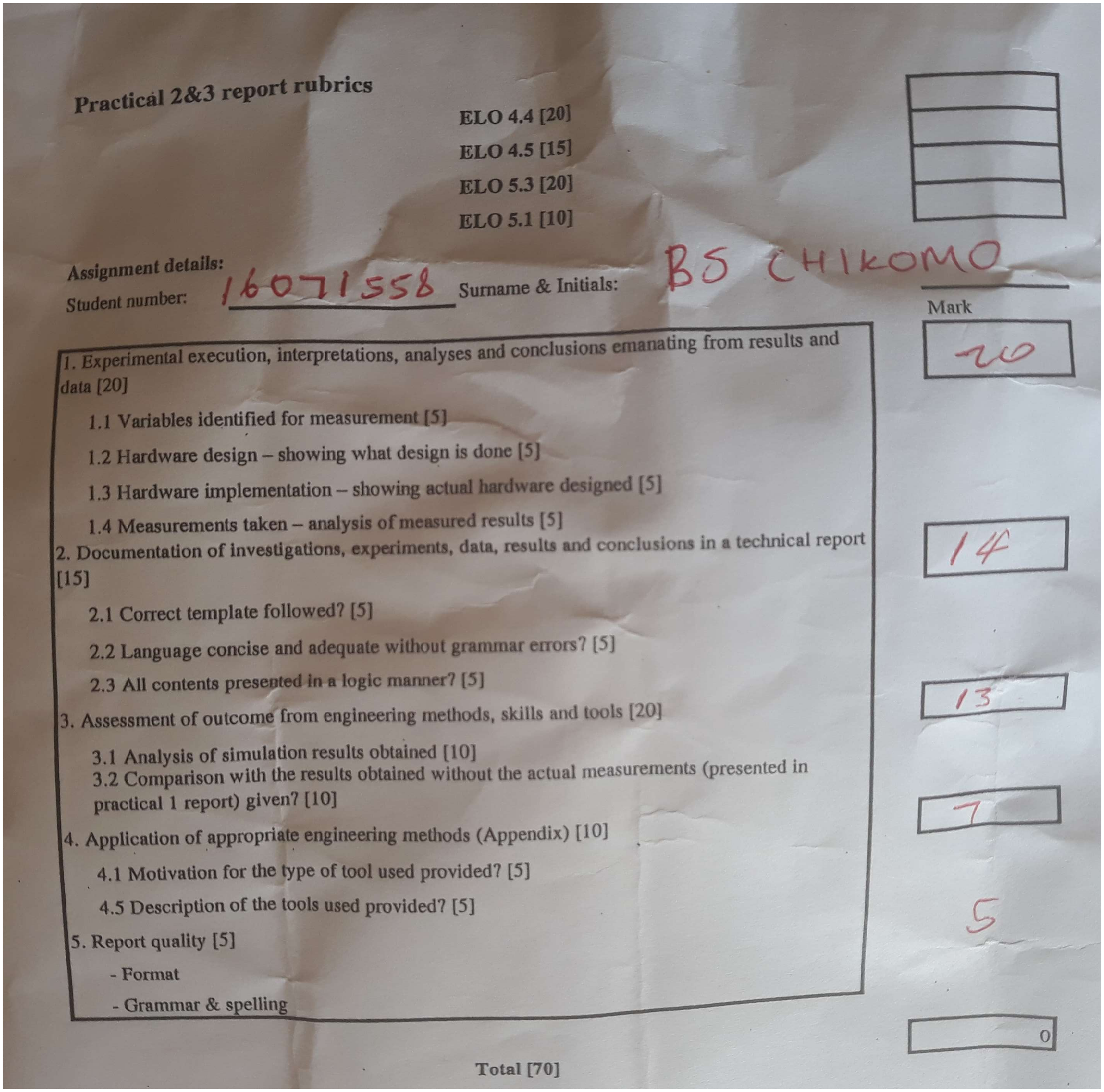
```c
#include "xc.h"
#include "header.h"
#include <stdio.h>
#include <stdlib.h>
#include <libpic30.h>
#include "p30F30F4011.h"
int adc2(void);
int adc3(void);
void delay(void);
int main(void) {
PORTB= 0b000001100;
TRISB= 0b000001100;
TRISC= 0b1111111111111001;
LATB= 0b000000000;


ADCON2= 0b0001100011100000;

ADCON2= 0b0000000000000000;
ADCON4= 0b0000110010111111;
ADCHS= 0b0001101100000011;
ADPCFG= 0b1110011111110000;
ADCSSL= 0b0000000000000000;

float ADC_sensor;
float CAL;
float temp1;
int 12temp;
int temp_VALUE; 15307672 12

DisTilt=0;
DisRot=0;
while(1)
{
ADC_sensor=adc2();
CAL=adc3();
temp1=(ADC_sensor/360)*CAL;
if (temp1>25)
temp_VALUE=temp1-25;
else if (temp1>15)
temp_VALUE=temp1-15;
else if (temp1>12)
temp_VALUE=temp1-12;
12temp=temp1/12;
PORTB=(16*12temp)+temp_VALUE;
}
return 0;
}
int adc2(void){
ADCON1bits.ADON=1;
ADCHSbits.CH0SA=0b0000;
ADCON1bits.SAMP=1;
delay();
ADCON1bits.SAMP=0;
while(!ADCON1bits.DONE){
}
ADCON1bits.ADON=0;
return ADCBUF0;
}
int adc2(void){
ADCON1bits.ADON=1;
ADCHSbits.CH0SA=0b0001;
ADCON1bits.SAMP=1;
delay();
ADCON1bits.SAMP=0;
while(!ADCON1bits.DONE){
}
ADCON1bits.ADON=0;
return ADCBUF0;
}
void delay(){
int CNTR;
for(CNTR=0;CNTR<10;CNTR++);
}
```



Practical 2&3 report rubrics

ELO 4.4 [20]
ELO 4.5 [15]
ELO 5.3 [20]
ELO 5.1 [10]

Assignment details:
Student number: 16071558 Surname & Initials: BS CHIKOMO

Mark

1. Experimental execution, interpretations, analyses and conclusions emanating from results and data [20] — 20
   1.1 Variables identified for measurement [5]
   1.2 Hardware design – showing what design is done [5]
   1.3 Hardware implementation – showing actual hardware designed [5]
   1.4 Measurements taken – analysis of measured results [5]
2. Documentation of investigations, experiments, data, results and conclusions in a technical report [15] — 14
   2.1 Correct template followed? [5]
   2.2 Language concise and adequate without grammar errors? [5]
   2.3 All contents presented in a logic manner? [5]
3. Assessment of outcome from engineering methods, skills and tools [20] — 13
   3.1 Analysis of simulation results obtained [10]
   3.2 Comparison with the results obtained without the actual measurements (presented in practical 1 report) given? [10]
4. Application of appropriate engineering methods (Appendix) [10] — 7
   4.1 Motivation for the type of tool used provided? [5]
   4.5 Description of the tools used provided? [5]
5. Report quality [5] — 5
   - Format
   - Grammar & spelling

Total [70] — 0

Fig. 10. Marking attach for this paper"ALL THE BEST!!"