

Treball de Fi de Grau

Grau en Enginyeria en Tecnologies Industrials

Implementació d'un banc de proves d'inversors multinivell per aplicacions de vehicles elèctrics

ANNEXOS

Autor: Adrià Luque Acera
Directors: Sergio Busquets Monge, Àlber Filbà Martínez
Convocatòria: Juny 2018



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Taula de continguts

A. Imatges dels components del sistema	3
B. Codis dels programes	13
B.1. Programa de captura de dades de l'oscil·loscopi.....	13
B.2. Programa de processat de dades.....	19
B.3. Programa per comparar resultats	24
C. Resultats gràfics de les proves	27
C.1. Cas 1	27
C.1.1. Cas 1a	27
C.1.2. Cas 1b	29
C.2. Cas 2	32
C.2.1. Cas 2a	32
C.2.2. Cas 2b	34
C.3. Cas 3	37
C.3.1. Cas 3a	37
C.3.2. Cas 3b	39

A. Imatges dels components del sistema

En aquest primer annex s'adjunten les fotografies de tots els elements importants del banc de proves que s'han usat durant el treball, així com algunes fotografies generals del sistema amb les seves connexions, que majoritàriament no s'han inclòs a la memòria per qüestions d'espai. El seu funcionament i les seves connexions sí estan descrits, així doncs aquí només tindran una breu descripció al peu.

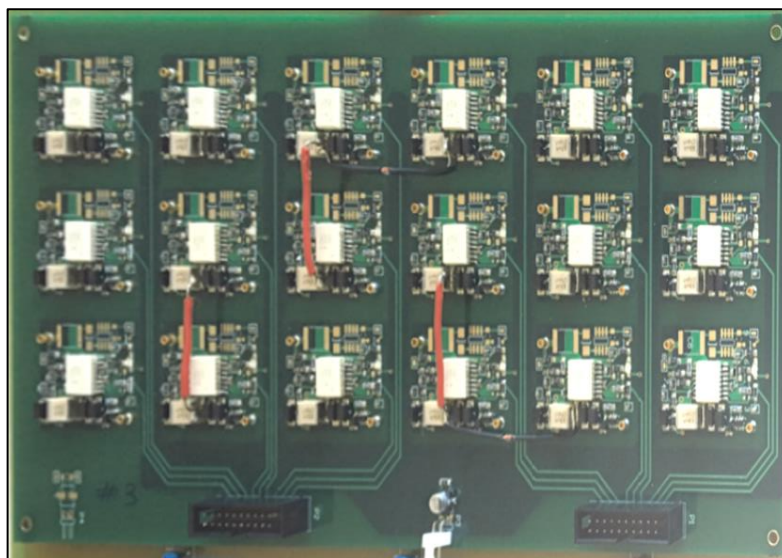


Figura A.1 – Placa del convertidor

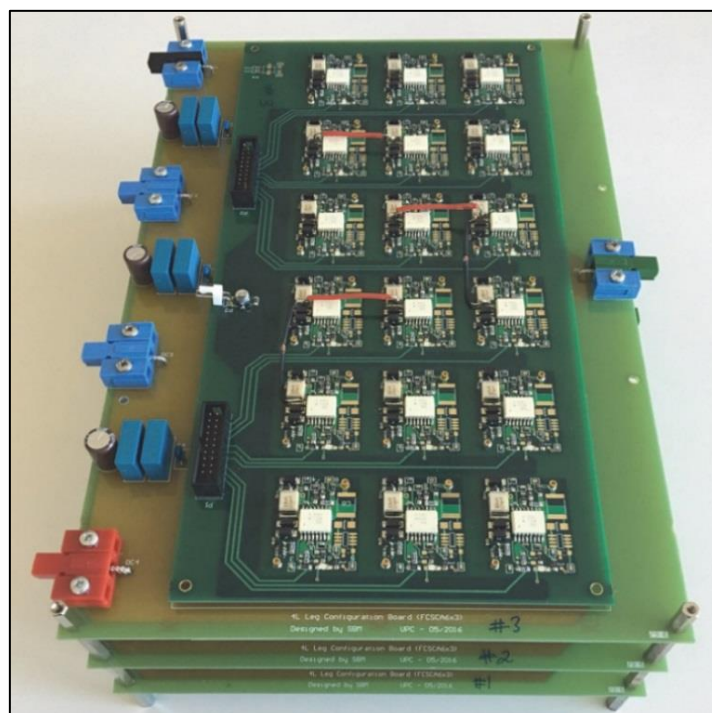


Figura A.2 – Prototip del convertidor EMAC trifàsic de 4 nivells



Figura A.3 – Caixa amb les bateries a dins



Figura A.4 – Càmera tèrmica Keysight U5855A: detall (esquerra) i col·locació al trípede (dreta)



Figura A.5 – Bobines de la càrrega trifàsica



Figura A.6 – Resistències (estufes) de la càrrega trifàsica



Figura A.7 – Fonts de CC usades per alimentar el convertidor i els sensors



Figura A.8 – Càrrega electrònica i multímetres usats per calibrar, i descarregar a corrent constant



Figura A.9 – Ordinador i dSpace

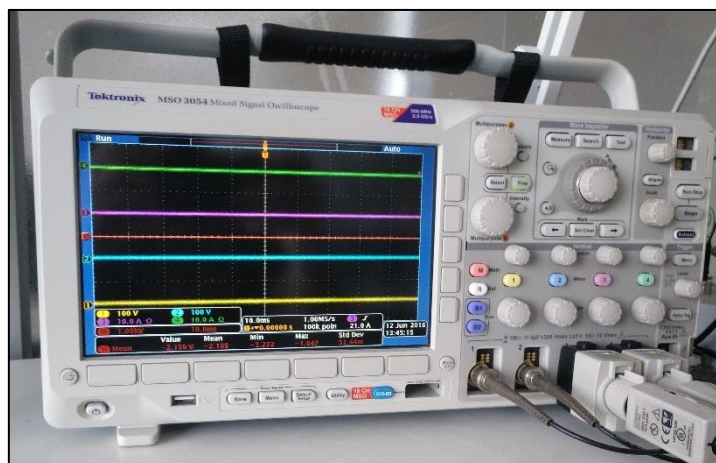


Figura A.10 – Oscil·loscopi MSO3054



Figura A.11 – Sistema d'Alimentació Ininterrompuda (frontal i posterior)



Figura A.12 – Fonts de CC usades com a sistema de contínua per les primeres proves

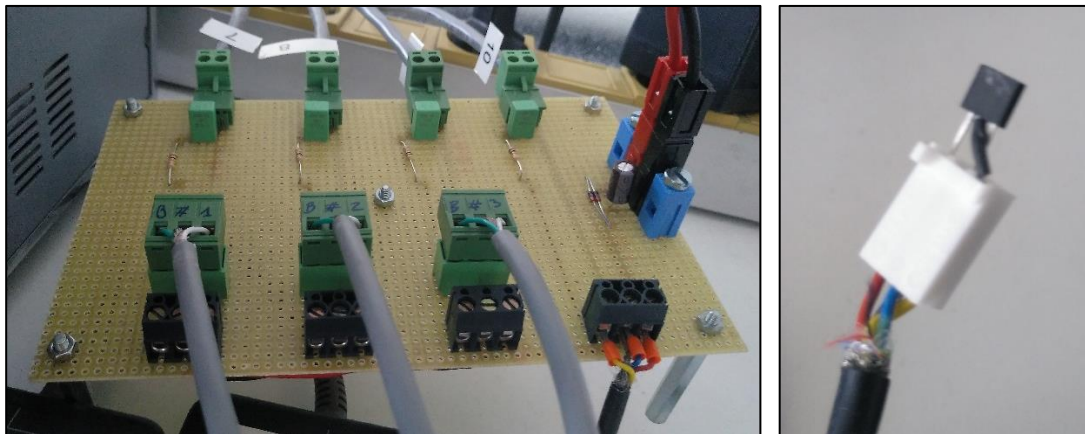


Figura A.13 – Placa on es connecten els sensors de temperatura (esquerra) i sensor de temperatura ambient (dreta)

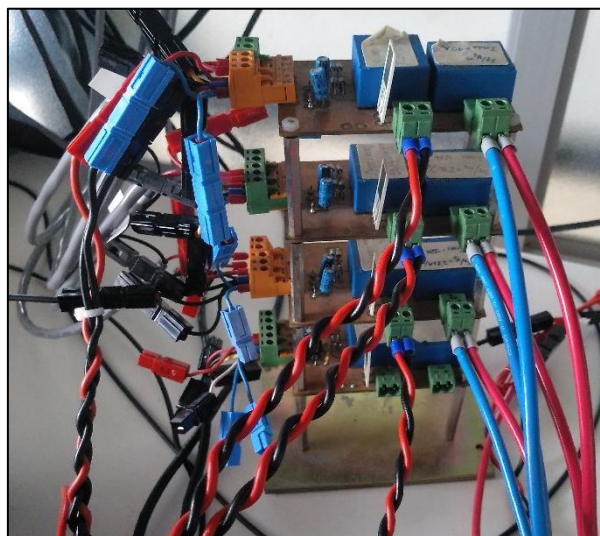


Figura A.14 – Torre de sensors de tensió i corrent del costat de contínua

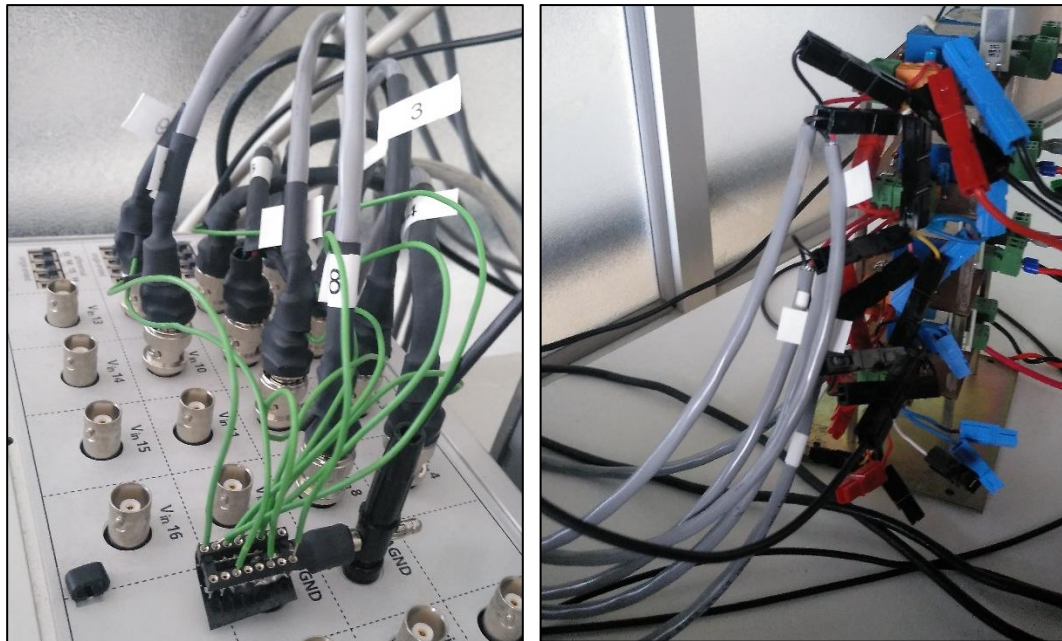


Figura A.15 – Placa de connexions dels sensors de la dSpace (esquerra) per connectar els sensors (dreta)

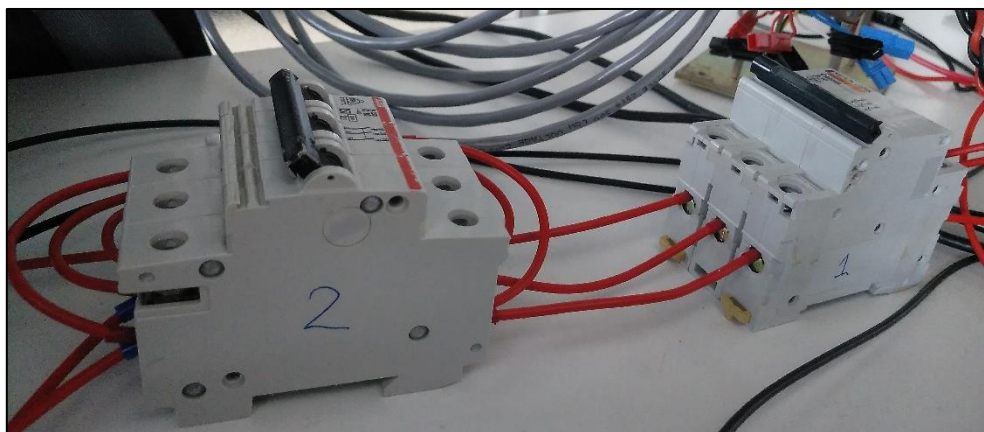


Figura A.16 – Interruptors per connectar les bateries al convertidor

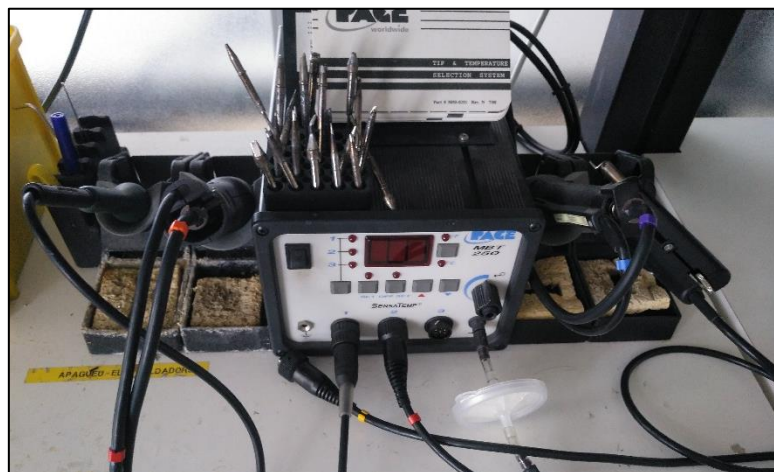


Figura A.17 – Soldador del laboratori

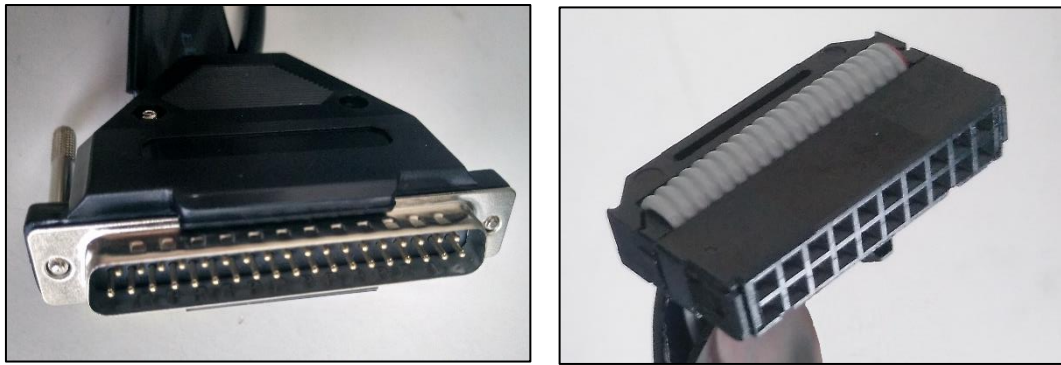


Figura A.18 – Connectors DB37 (esquerra) i IDC20 (dreta) per connectar convertidor i dSpace



Figura A.19 – Numeració de les bateries i els sensors per connectar-los correctament

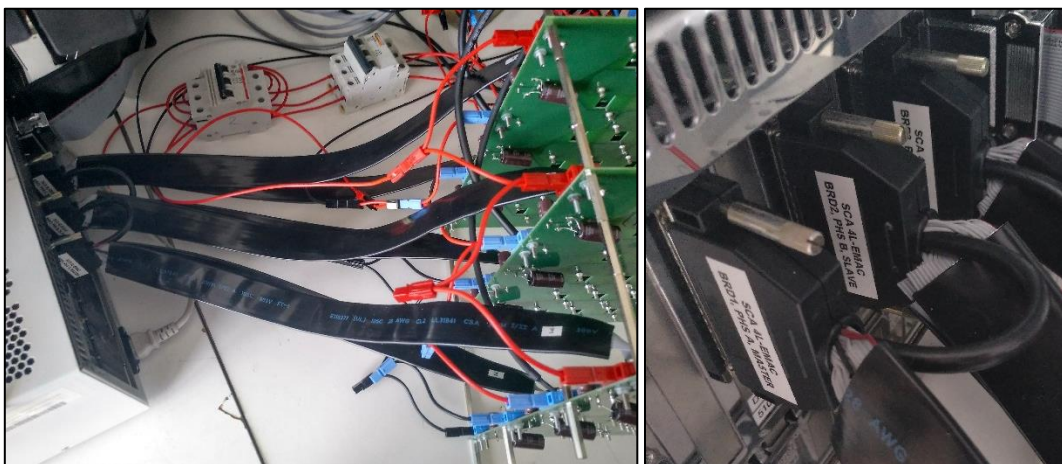


Figura A.20 – Connexió del convertidor a la dSpace i etiquetes dels connectors DB37



Figura A.21 – Connexió Aron al costat d'alterna del convertidor



Figura A.22 – Fotografies del sistema durant una prova de calibratge (esquerra) i del convertidor (dreta) amb els objectes mòbils en primer pla, la torre de fonts i la càmera en el trípede respectivament

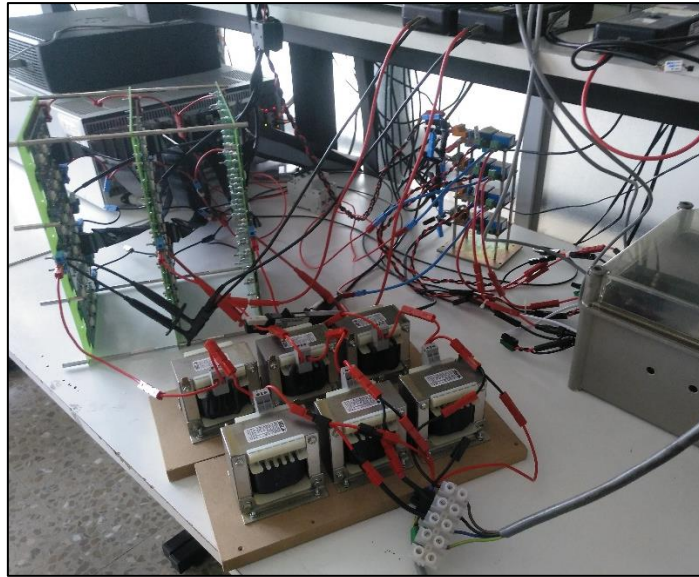


Figura A.23 – Detall del sistema: bateries, sensors, convertidor, bobines i dSpace al fons

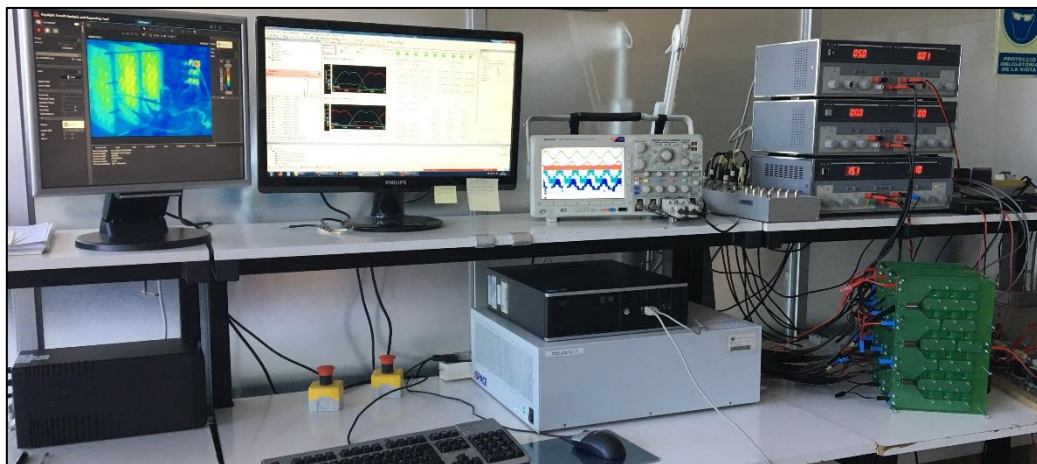


Figura A.24 – Vista general del sistema durant una prova, just quan l'oscil·loscopi fa una captura de pantalla



Figura A.25 – Vista general del banc de proves amb tots els elements

B. Codis dels programes

En aquest annex s'adjunten els codis complets dels programes realitzats durant el treball, començant pel de captura de dades de l'oscil·loscopi, el funcionament del qual s'ha descrit al capítol 3 de la memòria, seguit del programa que genera i desa els gràfics d'interès a partir dels fitxers de dades de les proves, descrit al capítol 5 de la memòria, i finalment el programa fet servir per facilitar la creació dels gràfics de comparació de dues proves que s'ha creat i usat durant la redacció de la memòria.

B.1. Programa de captura de dades de l'oscil·loscopi

Aquest apartat consta, de fet, de tres fitxers diferents, degut al fet que a les versions antigues de *Matlab* no es pot fer un *script* en un fitxer *.m* i afegir-hi funcions al final, sinó que s'han de separar en arxius diferents. El codi està comentat i acolorit per facilitar la seva comprensió.

Programa *get_data_MSO3054.m*:

```

%% Programa Principal Captura Dades %%
% El programa crea una GUI per editar paràmetres i controlar la
captura

% Panell de control per definir parametres i iniciar/aturar captura
bfig = figure('Name', 'Control Panel', 'NumberTitle','off');
set(bfig, 'Position', [100,100,240,300], 'Color', [0.9 0.9 0.9]);

% Text i camp per introduir el temps total de la prova
tttotaltx = uicontrol('Style','text', 'String', 'Temps total:', ...
    'Position', [20 268 80 20], 'BackgroundColor', [0.9 0.9 0.9]);
tttotalui = uicontrol('Style','edit', 'String', '4800', ...
    'Position', [100 270 50 20], 'Tag', 'tttotalui', ...
    'BackgroundColor', 'w');
tttotaltx_s = uicontrol('Style','text', 'String', ' s', ...
    'Position', [150 268 15 20], 'BackgroundColor', [0.9 0.9 0.9]);

% Text i camp per introduir el retard entre preses de dades
retardtx = uicontrol('Style','text', 'String', 'Retard:', ...
    'Position', [20 238 80 20], 'BackgroundColor', [0.9 0.9 0.9]);
retardui = uicontrol('Style','edit', 'String', '30', ...
    'Position', [100 240 50 20], 'Tag', 'retardui', ...
    'BackgroundColor', 'w');
retardtx_s = uicontrol('Style','text', 'String', ' s', ...
    'Position', [150 238 15 20], 'BackgroundColor', [0.9 0.9 0.9]);

% Control de captures de pantalla a l'oscil·loscopi
ferfotoui = uicontrol('Style','checkbox', 'String', 'Fer foto', ...
    'Position', [100 210 100 20], 'Callback', @togglefoto, ...
    'Tag', 'ferfotoui', 'BackgroundColor', [0.9 0.9 0.9]);
fotocadatx = uicontrol('Style','text', 'String', 'Foto cada:', ...

```

```

    'Position', [20 178 80 20], 'BackgroundColor', [0.9 0.9 0.9]);
fotocadaui = uicontrol('Style','edit', 'String', '10', ...
    'Position', [100 180 50 20], 'Enable', 'off', 'Tag',
'fotocadaui');
fotocadatx_s = uicontrol('Style','text', 'String', 'dades', ...
    'Position', [150 178 40 20], 'BackgroundColor', [0.9 0.9 0.9]);

% Botons per escollir tipus de VISA
grupbv = uibuttongroup('Title', 'Tipus VISA',
'Position',[.075 .4 .85 .15], ...
    'BackgroundColor', [0.9 0.9 0.9], 'Tag', 'grupbv');
visatcpipui = uicontrol(grupbv, 'Style','radiobutton', 'String',
'TCPIP', ...
    'Position', [40 7.5 80 20], 'BackgroundColor', [0.9 0.9
0.9], ...
    'Value', 0, 'Tag', 'visatcpipui');
visausbui = uicontrol(grupbv, 'Style','radiobutton', 'String',
'USB', ...
    'Position', [120 7.5 80 20], 'BackgroundColor', [0.9 0.9
0.9], ...
    'Value', 1, 'Tag', 'visausbui');

% Esperar a detectar un nou arxiu o iniciar immediatament
scanui = uicontrol('Style','checkbox', 'String', 'Inicia
immediatament', ...
    'Position', [60 90 120 20], ...
    'Tag', 'scanui', 'BackgroundColor', [0.9 0.9 0.9]);

% Botó START - Crida la funció startttest (ha d'estar al mateix dir)
startbutton = uicontrol('Style', 'PushButton', 'String', 'INICIA LA
CAPTURA', ...
    'Callback', @startttest, ...
    'Position', [20 60 200 30], ...
    'BackgroundColor', [0 1 0], 'Enable', 'on', 'Tag',
'startbutton');

% Botó STOP - Atura la captura i tanca la finestra
% A la funció startttest es comprova que s'ha polsat el botó
% S'atura la captura i es desa el fitxer
stopbutton = uicontrol('Style', 'PushButton', 'String', 'ATURA LA
CAPTURA', ...
    'Callback', 'delete(gcf)', 'Position', [20 20 200 30], ...
    'BackgroundColor', [1 0 0], 'Enable', 'off', 'Tag',
'stopbutton');

```

Funció *togglephoto.m*:

```

function togglefoto(ferfotoui, eventdata)
    hfotocadaui = findobj('Tag','fotocadaui');
    if get(ferfotoui,'Value') == 1
        set(hfotocadaui,'Enable','on');
    else
        set(hfotocadaui,'Enable','off');
    end
end

```


Funció *starttest.m*:

```

function starttest(startbutton, ~)

    % Path on es creen els arxius temporals del ControlDesk
    pathcd = 'D:\04
SBM\03_Baterias_4L\Project_CD\Experiment_001\Measurement Data';

    % Trobem tots els objectes del panell de control
    hretard = findobj('Tag','retardui');
    htttotal = findobj('Tag','tttotalui');
    hferfoto = findobj('Tag','ferfotoui');
    hfotocadax = findobj('Tag','fotocadaui');
    hstopbutton = findobj('Tag','stopbutton');
    hvisatcpip = findobj('Tag','visatcpipui');
    hvisausb = findobj('Tag','visausbui');
    hscan = findobj('Tag','scanui');

    % Desactivem els canvis al panell
    set(startbutton,'Enable','off');
    set(hretard,'Enable','off');
    set(htttotal,'Enable','off');
    set(hferfoto,'Enable','off');
    set(hfotocadax,'Enable','off');
    set(hvisatcpip,'Enable','off');
    set(hvisausb,'Enable','off');
    set(hscan,'Enable','off');

    % Esperem a que es creï un nou arxiu al directori indicat
    if get(hscan,'Value') == 0
        set(startbutton,'String','ESCAJANT...');
        set(startbutton,'BackgroundColor',[1 0.8 0]);
        dir_content = dir(pathcd);
        filenames = {dir_content.name};
        current_files = filenames;
        waitmax = 300; %Temps d'espera
        for twait = 1:waitmax
            dir_content = dir(pathcd);
            filenames = {dir_content.name};
            new_files = setdiff(filenames, current_files);
            if ~isempty(new_files)
                disp('Fitxer detectat. Iniciant captura');
                break
            end
            pause(0.100)
        end

    % Si s'arriba al temps d'espera màxim s'atura el programa
    if twait == waitmax
        disp('Cap fitxer nou detectat. Captura aturada');
        set(startbutton,'String','INICIA LA CAPTURA');
        set(startbutton,'BackgroundColor',[0 1 0]);
        set(startbutton,'Enable','on');
        set(hretard,'Enable','on');
        set(htttotal,'Enable','on');
        set(hferfoto,'Enable','on');
    end
end

```

```

        set(hferfoto, 'Value', 0);
        set(hvisatcpip, 'Enable', 'on');
        set(hvisausb, 'Enable', 'on');
        set(hscan, 'Enable', 'on');
        return
    end

else
    disp('Captura immediata iniciada');
end

% Si s'ha creat un arxiu dins del temps esperat
% O s'ha activat l'inici immediat, s'inicia la captura
set(startbutton, 'String', 'CAPTURANT');
set(startbutton, 'BackgroundColor', [0 1 1]);
set(hstopbutton, 'Enable', 'on');

% Agafem el valor de les variables del panell
retard = str2double(get(hretard, 'String'));
tttotal = str2double(get(httotal, 'String'));
ferfoto = get(hferfoto, 'Value');
fotocadax = str2double(get(hfotocadax, 'String'));

nmesures = ceil(tttotal/retard)+1;

% Obtenim els temps
inici = clock;
inici = inici(4)*3600 + inici(5)*60 + inici(6);
final = inici + tttotal;

ara = clock;
ara = ara(4)*3600 + ara(5)*60 + ara(6);

% Trobem el número de test segons els fitxers de la carpeta
numtest = 0;
fitxers = ls('*test*');
numf = size(fitxers);
for f = 1:numf(1)
    %dades_testN_yyyy-m-dd_hh-mm.mat
    possnum = fitxers(f,:);
    talli = strfind(possnum, 'Test');
    if ~isempty(talli)
        possnum = possnum((talli(1)+4):end);
        possnum = str2double(possnum);

        if and(~isempty(possnum), possnum>numtest)
            numtest = possnum;
        end
    end
end

numtest = numtest + 1;
numtest = num2str(numtest);

% Creem un directori pel test
ndir = ['Test', numtest];
mkdir(ndir)

```

```

% Creem l'objecte VISA per a comunicar-nos amb l'oscil·loscopi.
if get(hvisatcpip, 'Value') == 1
    g1 = visa('tek', 'TCPIP0::147.83.138.70::INSTR');
elseif get(hvisausb, 'Value') == 1
    g1 = visa('tek', 'USB::0x0699::0x0425::C010328::INSTR');
end

% IMG: Augmentem el tamany del buffer per poder rebre les dades
set(g1, 'InputBufferSize', 480*800*3);

% Obrim el port de comunicacions.
fopen(g1);

% Iteració principal per prendre dades cada cert temps
breaking = 0;
i = 1;
while i <= nmesures

    % Demanem el valor de la mesura i
    fprintf(g1, 'measurement:meas1:value?');
    dadai = fscanf(g1);
    if isempty(str2double(dadai))
        dadai = str2double(dadai(26:end));
    else
        dadai = str2double(dadai);
    end

    % Afegim el temps i el valor mesurat a la llista
    dades(i,:) = [(i-1)*retard, dadai];

    % Condició per capturar imatge:
    %     que la mesura sigui múltiple del que hem dit
    %     (tenint en compte que comença a t=0s -> i=1), i
    %     que ferfoto estigui activat (1)

    if and(rem(i-1, fotocadax) == 0, ferfoto)

        % IMG: Fixem el format com a BMP de color
        fprintf(g1, 'save:image:fileformat BMP');
        % IMG: Configurem el hardcopy per imatge horitzontal
        fprintf(g1, 'HARDCopy:layout Landscape');
        % IMG: Netegem el buffer de sortida de l'oscil·loscopi.
        fprintf(g1, 'CLEARSpool');
        % IMG: Realitzem la captura i la guardem a un vector
        fprintf(g1, 'HARDCopy start');
        A = fread(g1, 800*480*3, 'uint8');
        % IMG: Ordenem les dimensions i corregint l'offset
        img = (permute(uint8(reshape(A, 3, 800, 480)), [3 2 1]));
        img = img(end:-1:1, [20:end, 1:19], :);

        % Guardem la matriu de la imatge a un arxiu BMP.
        % El nom de l'arxiu conté la data i hora actuals.
        dataihora = clock;
        if dataihora(5) < 10
            minuts = ['0', num2str(dataihora(5))];
        else
            minuts = num2str(dataihora(5));
        end
    end
end

```

```

        titol = ['captura_test', numtest, '_', ...
                num2str(dataihora(1)), '-', num2str(dataihora(2)), ...
                '-', num2str(dataihora(3)), '_', ...
                num2str(dataihora(4)), '-', minuts, '-', ...
                num2str(round(dataihora(6)))];
        imwrite(img, colormap, ['./', ndir, '/', titol, '.bmp'], ...
                'BMP');

    end

    % Si no hem arribat al final, esperem un retard
    while and(ara < (inici + i*retard), ara <= final)
        ara = clock;
        ara = ara(4)*3600 + ara(5)*60 + ara(6);
        pause(0.1)

        % Parem si s'ha polsat el botó STOP
        if ~ishandle(startbutton)
            disp('Captura aturada!');
            breaking = 1;
            break
        end
    end
    if breaking
        break
    end

    % Actualitzem els camps del while
    i = i+1;
    ara = clock;
    ara = ara(4)*3600 + ara(5)*60 + ara(6);
end

% Fi del bucle principal de captures iteratives

% Tanquem el port de comunicacions i l'eliminem
fclose(g1);
clear g1;

if ishandle(startbutton)
    set(startbutton, 'String', 'CAPTURA ACABADA');
    set(startbutton, 'BackgroundColor', [0 1 0]);
    set(hstopbutton, 'String', 'TANCA');
end

% Obtenim la data i hora actuals (afegint 0 als minuts si cal)
ara = clock;
if ara(5) < 10
    minuts = ['0', num2str(ara(5))];
else
    minuts = num2str(ara(5));
end

% Posem nom al fitxer
titol = ['dades_test', numtest, '_', num2str(ara(1)), '-', ...
        num2str(ara(2)), '-', num2str(ara(3)), '_', ...
        num2str(ara(4)), '-', minuts];

```

```

% Desem l'arxiu al directori del test
save(['./',ndir,'/',titol,'.mat'], 'dades');

% Grafic de les dades vs. temps
tp = figure();
plot(dades(:,1),dades(:,2));
saveas(tp, ['./',ndir,'/plot_test',numtest,'.bmp']);

end

```

B.2. Programa de processat de dades

En aquest apartat s'inclou el programa que, donat un directori on hi ha un fitxer de dades capturades des de l'oscil·loscopi fent servir `get_data_MSO3054.m` i un de dades capturades pel ControlDesk amb les variables correctes, representa 7 gràfics d'interès i els desa. Els gràfics estan numerats i sempre són:

1. Tensions de les bateries (vB)
2. Corrents de les bateries (iBav)
3. Estats de càrrega de les bateries (SOC)
4. Potències de les bateries (pB)
5. Temperatures de les bateries i ambient (Ta)
6. Potència d'entrada i de sortida (CC i CA) i rendiment del convertidor
7. Senyals o accions de control (k)

Aquest programa consta d'un sol fitxer amb una funció, `plot_data.m`:

```

function plot_data()

dirprova = input('Carpeta amb els arxius de la prova: ', 's');
plotspos = [600,100,800,600]; %[-1200,300,800,600] al lab;
scopeplot = 0; imformat = 'fig'; %png

% Per tenir els mateixos colors que als gràfics del lab (Pre-R2014b)
colors = [[0 0 1]; [0 0.5 0]; [1 0 0]; [0 0.75 0.75]; ...
          [0.75 0 0.75]; [0.75 0.75 0]; [0.25 0.25 0.25]];
set(groot, 'defaultAxesColorOrder', colors);

filein = ls(['./', dirprova, '/*exp*']);
filein = filein(1:(length(filein)-4));
fileout = ls(['./', dirprova, '/*dades*']);
fileout = fileout(1:(length(fileout)-4));

file = load(['./', dirprova, '/', filein]);
file2 = load(['./', dirprova, '/', fileout]);

% Les dades de X són el temps
t = file.(filein).X.Data;
t2 = file2.dades(:,1)';

```

```

% Troba els índexos de tensions, corrents, SOC i T dins Y
Jvbat = find(strncmp({file.(filein).Y.Name}, 'vB',2));
Jibat = find(strncmp({file.(filein).Y.Name}, 'iBav',4));
JSOCbatk = find(strncmp({file.(filein).Y.Name}, 'SOC',3));
JTa = find(strncmp({file.(filein).Y.Name}, 'Ta',2));
Jk = find(strncmp({file.(filein).Y.Name}, 'k',1));

Nbats = length(Jvbat);

for k = 1:Nbats
    % Crea tres matrius, una per cada magnitud
    % Posa les dades de cada bateria en una fila diferent
    vbatk(k,:) = file.(filein).Y(Jvbat(k)).Data;
    ibatk(k,:) = file.(filein).Y(Jibat(k)).Data;
    SOCbatk(k,:) = file.(filein).Y(JSOCbatk(k)).Data;
end

for j = 1:length(JTa)
    % Crea una matriu per T (length(JTa) != Nbats)
    Ta(j,:) = file.(filein).Y(JTa(j)).Data;
end

for j = 1:length(Jk)
    % Crea una matriu per k (length(Jk) != Nbats)
    ki(j,:) = file.(filein).Y(Jk(j)).Data;
end

% Calcula potència, càrrega i energia
Pbatk = vbatk.*ibatk;
Qk = trapz(t,ibatk,2)'/3600;
Ek = trapz(t,Pbatk,2)'/3600;

% Creem vectors de magnituds per calcular-les totes de cop
database = [vbatk; ibatk; SOCbatk; Pbatk];
dbnames = {'1- Tensions','2- Corrents','3- SOCs', ...
           '4- Potències de les bateries'};
dbvariables = {'Tensió [V]','Corrent [A]','SOC','Potència [W]'};
magnitudes = length(database(:,1))/Nbats;

% Trobem el número de test segons els fitxers de la carpeta
numtest = 0;
fitxers = ls(['./', dirprova, '/*plots*']);
numf = size(fitxers);
for f = 1:numf(1)
    %TestN
    possnum = fitxers(f,:);
    talli = strfind(possnum, 'Plots');
    if ~isempty(talli)
        possnum = possnum((talli(1)+5):end);
        possnum = str2double(possnum);
        if and(~isempty(possnum), possnum>numtest)
            numtest = possnum;
        end
    end
end
numtest = numtest + 1;
numtest = num2str(numtest);

```

```

% Creem un directori pels gràfics
ndir = ['./', dirprova, '/Plots', numtest];
mkdir(ndir);

##### PLOT #####

% ALLbatk vs. temps
for g = 1:magnitudes
    h = figure(g);
    set(h, 'Color', 'w');
    set(h, 'Position', plotspos);
    ini = 1 + (g-1)*Nbats;
    fin = g*Nbats;
    plot(t, database(ini:fin,:));
    legend('Bat 1','Bat 2','Bat 3');
    xlim(gca,[0 max(t)])
    titol = dbnames{g};
    title(titol)
    xlabel('Temps [s]');
    ylabel(dbvariables(g));
    saveas(gca,['./',ndir,'/',titol], imformat)
end

% Temperatura vs. temps
h = figure(g+1);
set(h, 'Color', 'w');
set(h, 'Position', plotspos);
plot(t,Ta);
xlim(gca,[0 max(t)])
ylim(gca,[20 60])
xlabel('Temps [s]')
ylabel('Temperatura [°C]')
title('5- Temperatures Bateries i Ambient')
legend('Bat 1', 'Bat 2', 'Bat 3', 'Ambient');
saveas(gca,['./',ndir,'/5- Temperatures Bateries-Ambient'], imformat)

% Potència total entrada i sortida, i rendiment en funció del temps
% Si s'ha definit scopeplot=1 pinta les captures de l'oscil·loscopi
% com punts taronjes sobre la interpolació. Si no, només pinta la
% interpolació com una línia verda
h = figure(g+2);
set(h, 'Color', 'w');
set(h, 'Position', plotspos);
if length(Pbatk(:,1)) == 1
    P_in = Pbatk;
else
    P_in = sum(Pbatk);
End

P_out = file2.dades(:,2)';
P_oip = interp1(t2, P_out, t);
rend = P_oip ./ P_in;
trend = t;
xmax = max([t, t2]);

```

```

for ri=length(rend):-1:1
    if (rend(ri) < 0) || (rend(ri) > 1) || ...
        (P_oip(ri) <= 400) || (ri <= 120)
        rend(ri) = [];
        trend(ri) = [];
    end
end

E_out = trapz(t2,P_out)'/3600;
[hax, ~, ~] = plotyy(t,[P_in;P_oip],trend,rend);
xlim(hax(1),[0 xmax])
xlim(hax(2),[0 xmax])
ylim(hax(1), [0 800])
ylim(hax(2), [0.6 1.4])
set(hax(2), 'xTick', 0:2500:xmax)
set(hax(2), 'xTick', 0:2500:xmax)
set(hax(1), 'yTick', linspace(0, 800, 9))
set(hax(2), 'yTick', linspace(0, 1, 21))
xlabel('Temps [s]')
ylabel(hax(1), 'Potència [W]')
ylabel(hax(2), 'Rendiment')
grid on

if (scopeplot)
    hold on
    plot(t2, P_out, '.', 'Color', [1 0.6 0])
    hold off
    legend('P_{in}', 'P_{out (ip.)}', 'P_{out (osc.)}', 'Rend');
else
    legend('P_{in}', 'P_{out}', 'Rend');
end
title('6- Potència entrada-sortida i rendiment')
saveas(gca,['./',ndir,'/6- Potència in-out i rendiment'], imformat)

% Variables k de control en funció del temps
h = figure(g+3);
set(h, 'Color', 'w');
set(h, 'Position', plotspos);
plot(t,ki);
xlim(gca,[0 max(t)])
ylim(gca,[-0.151 0.151])
xlabel('Temps [s]')
ylabel('Acció k')
title('7- Variables de control')
legend('k_{2}', 'k_{3}');
saveas(gca,['./',ndir,'/7- Variables de control'], imformat)

%%%%% DISPLAY %%%%%
% Nom, Data, Durada
% Primera bateria que s'apaga
% Q de cada bateria i total
% E de cada bateria, total i AC
% Rendiment

```



```

nomprova = dirprova;
dataprova = file.(filein).Description.General.DateTime;
duradaprova = file.(filein).Description.Measurement.Length;

% Primera bateria que es desconnecta
primeraoff = 0;
tempsoff = inf;
for i = 1:Nbats
    salt_vbat = find(vbatk(i,:) < 10, 1);
    if ~isempty(salt_vbat)
        if salt_vbat < tempsoff
            primeraoff = i;
        end
    end
end

% Mostra les dades importants per pantalla
disp(['Nom de la prova:      ', nomprova])
disp(['Data de la prova:    ', dataprova])
disp(['Durada de la prova:   ', num2str(duradaprova), ' s'])
disp(['la Bateria apagada:  ', num2str(primeraoff)])
disp(['Càrrega extreta:     ', 'B1: ', num2str(Qk(1)), ' Ah.      ', ...
      'B2: ', num2str(Qk(2)), ' Ah.      ', ...
      'B3: ', num2str(Qk(3)), ' Ah'])
disp(['Energia extreta:      ', 'B1: ', num2str(Ek(1)), ' Wh.      ', ...
      'B2: ', num2str(Ek(2)), ' Wh.      ', ...
      'B3: ', num2str(Ek(3)), ' Wh'])
disp(['Energia total DC:    ', num2str(sum(Ek)), ' Wh']);
disp(['Energia total AC:    ', num2str(E_out), ' Wh']);
disp(['Rendiment (AC/DC):   ', num2str(E_out/sum(Ek)*100), ' %']);

% Comprova si hi ha un Diari de Proves al directori
% Si no existeix, crea la capçalera
hihaxl = ls('*diariproves*');
if ~isempty(hihaxl)
    [~, ~, raw] = xlsread('DiariProves.xlsx');
    xlfila = num2str(length(raw(:,1)) + 1);
else
    initdata = {'Nom', 'Data i hora', 'Durada (s)', '1a Bat OFF', ...
               'QB1ext (Ah)', 'QB2ext (Ah)', 'QB3ext (Ah)', 'QText (Ah)', ...
               'EB1ext (Wh)', 'EB2ext (Wh)', 'EB3ext (Wh)', 'EText (Wh)', ...
               'E_AC (Wh)', 'Rendiment'};
    xlswrite('DiariProves.xlsx', initdata, 'A1:N1');
    xlfila = '2';
end

% Escriu les dades a la fila corresponent
xlrang = ['A', xlfila, ':N', xlfila];
xlldata = {nomprova, dataprova, duradaprova, primeraoff, ...
           Qk(1), Qk(2), Qk(3), sum(Qk), ...
           Ek(1), Ek(2), Ek(3), sum(Ek), E_out, E_out/sum(Ek)};
xlswrite('DiariProves.xlsx', xlldata, xlrang);

end

```

B.3. Programa per comparar resultats

Per últim, s'inclou un programa escrit per facilitar la creació dels gràfics de comparació entre proves usats a la memòria. Aquest programa s'ha elaborat després de realitzar les proves, i tot i formar part del processat de les dades, no era el que s'havia plantejat als objectius i per tant no s'ha detallat com els altres a la memòria. Com els altres dos, està comentat i acolorit per facilitar la seva lectura.

En essència, el programa agafa com a entrades dos fitxers de dades del *ControlDesk* (on s'assumeix que un correspon a una prova amb control i l'altre a una prova sense control, però es pot usar per altres comparacions si es vol) i una variable registrada, i fa el gràfic corresponent amb línies sòlides pel fitxer proporcionat al input "amb control" i amb línies puntejades pel fitxer "sense control". Simplement agilitza el procés de creació dels gràfics i permet editar el que genera.

Consta d'un sol fitxer, *CompareResults.m*:

```

%%% COMPARA PROVES %%%

% Entrades: dos directoris i una variable
dir_nc = input('Directori amb dades sense control: ', 's');
dir_wc = input('Directori amb dades amb control: ', 's');
variable = input('Variable a comparar: ', 's');

% Trobem els fitxers
nfile_nc = ls(['./', dir_nc, '/*exp*']);
nfile_nc = nfile_nc(1:(length(nfile_nc)-4));
nfile_wc = ls(['./', dir_wc, '/*exp*']);
nfile_wc = nfile_wc(1:(length(nfile_wc)-4));

% Carreguem els fitxers
filenc = load(['./', dir_nc, '/', nfile_nc]);
filewc = load(['./', dir_wc, '/', nfile_wc]);

% Les X són el temps
t_nc = filenc.(nfile_nc).X.Data;
t_wc = filewc.(nfile_wc).X.Data;

% Trobem els índexs de la variable dins els Structs
Jvar_nc = find(strncmp({filenc.(nfile_nc).Y.Name}, variable,
length(variable)));
Jvar_wc = find(strncmp({filewc.(nfile_wc).Y.Name}, variable,
length(variable)));
Nbats = length(Jvar_nc);

% Creem les matrius per les bateries
for k = 1:Nbats
    var_nc(k,:) = filenc.(nfile_nc).Y(Jvar_nc(k)).Data;
    var_wc(k,:) = filewc.(nfile_wc).Y(Jvar_wc(k)).Data;
end

```

```
% Configuració general de la figura
% Es fan servir propietats de Matlab recents
h = figure(1);
h.Position = [700,150,800,600];
h.Color = [1 1 1];

% Dibuixem els gràfics de la variable per ambdós casos
% Les propietats es poden editar segons convingui
hold on
p11 = plot(t_wc, var_wc(1,:), '-r');
p11.LineWidth = 0.5;
p21 = plot(t_nc, var_nc(1,:), ':r');
p21.LineWidth = 0.5;

p12 = plot(t_wc, var_wc(2,:), '-', 'Color', [0 0.6 0]);
p12.LineWidth = 0.5;
p22 = plot(t_nc, var_nc(2,:), ':', 'Color', [0 0.6 0]);
p22.LineWidth = 0.5;

% Si la variable és k, només hi ha dues línies a dibuixar
if ~strcmp(variable, 'k')
    p13 = plot(t_wc, var_wc(3,:), '-b');
    p13.LineWidth = 0.5;
    p23 = plot(t_nc, var_nc(3,:), ':b');
    p23.LineWidth = 0.5;
end

% Si la variable és Ta, llavors hi ha quatre línies
if strcmp(variable, 'Ta')
    p14 = plot(t_wc, var_wc(4,:), '-', 'Color', [0 0.7 0.7]);
    p14.LineWidth = 0.5;
    p24 = plot(t_nc, var_nc(4,:), ':', 'Color', [0 0.7 0.7]);
    p24.LineWidth = 0.5;
end
hold off

% Configura els límits del gràfic
box on
xlim([0 max(3600, max(t_wc))])
xlabel('Temps [s]')
ymax = max(max(max(var_nc)), max(max(var_wc)));

if strcmp(variable, 'SOC')
    ylim([-0.05 1])
end

% Configura les unitats
switch variable
    case 'vB'
        unitat = '[V]';
    case 'iBav'
        unitat = '[A]';
    case 'SOC'
        unitat = '';
    case 'Ta'
        unitat = '[°C]';
    case 'k'
        unitat = '';
end
```

```
% Inclou els retols de cada línia del color corresponent
tx1 = text(-300,ymax/2-ymax/10,[variable, '_1'],'Color','r');
set(tx1, 'Rotation',90);
tx2 = text(-300,ymax/2,[variable, '_2'],'Color',[0 0.6 0]);
set(tx2, 'Rotation',90);
tx3 = text(-300,ymax/2+ymax/10,[variable, '_3'],'Color','b');
set(tx3, 'Rotation',90);
txa = text(-325,ymax/2+ymax/5,unitat);
set(txa, 'Rotation',90);
```

C. Resultats gràfics de les proves

A continuació s'adjunta la col·lecció sencera dels gràfics generats pel programa *plot_data.m* dels resultats per cada cas diferent, però només per la segona repetició de les proves, i amb alguns canvis de dimensions per ocupar menys espai. És necessari recordar que al Cas 1 la bateria central comença al 80% de càrrega, al Cas 2 és la primera la que comença més descarregada, i al Cas 3 totes tres bateries comencen amb la càrrega plena. En tots tres casos, la prova a és sense activar el control i la b és amb el control per SOC activat. El programa genera sempre 7 gràfics, que s'adjunten amb la numeració corresponent.

C.1. Cas 1

C.1.1. Cas 1a

Gràfics corresponents a les proves realitzades el 27/4/2018.

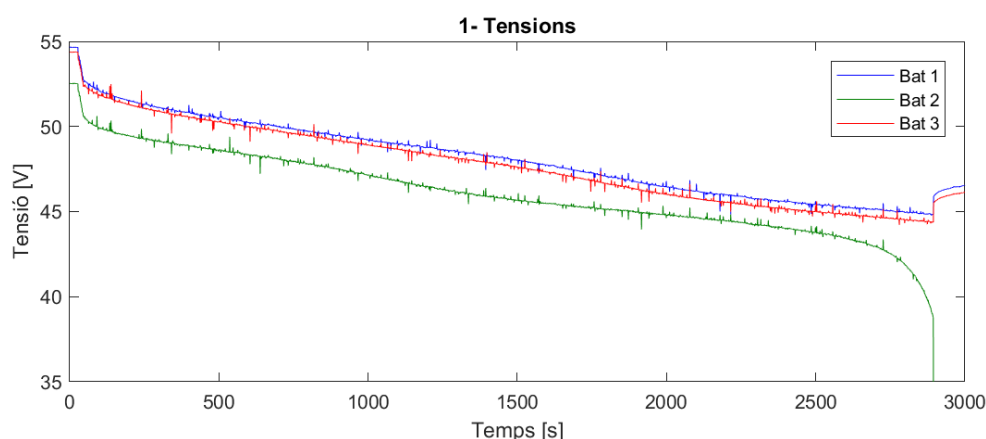


Figura C.1 – Gràfic de tensions pel Cas 1a

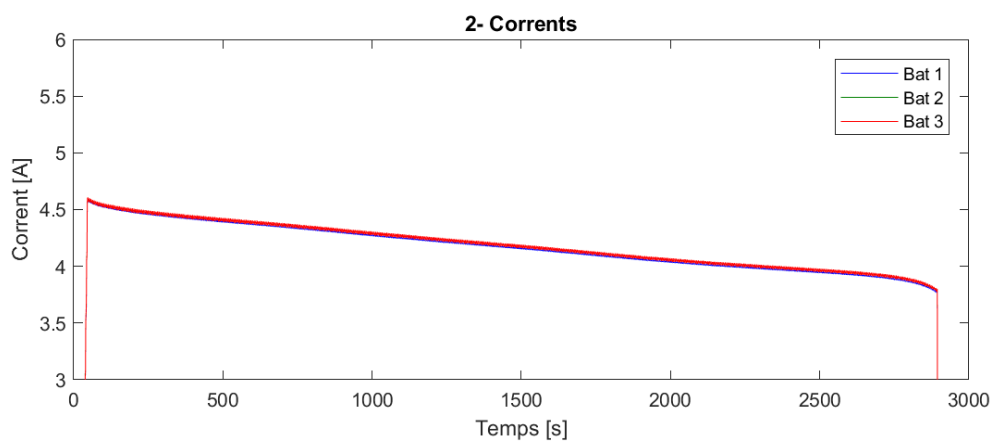


Figura C.2 – Gràfic de corrents pel Cas 1a

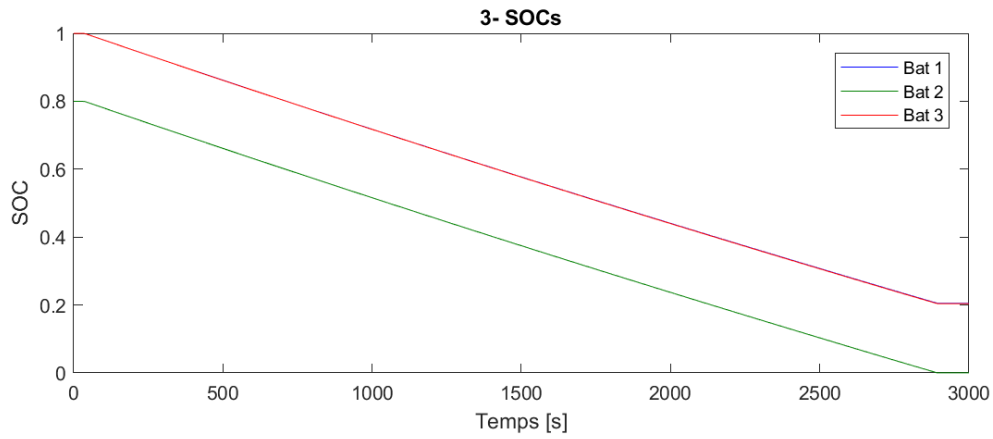


Figura C.3 – Gràfic d'estats de càrrega pel Cas 1a

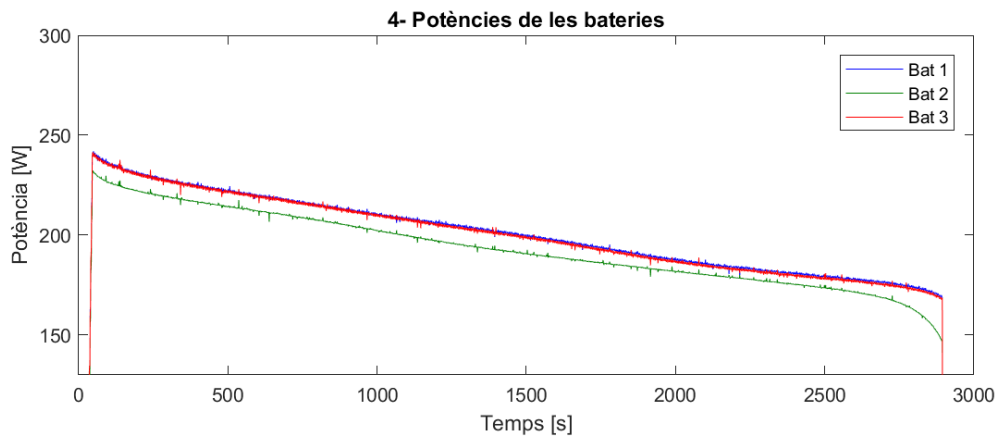


Figura C.4 – Gràfic de potències de les bateries pel Cas 1a

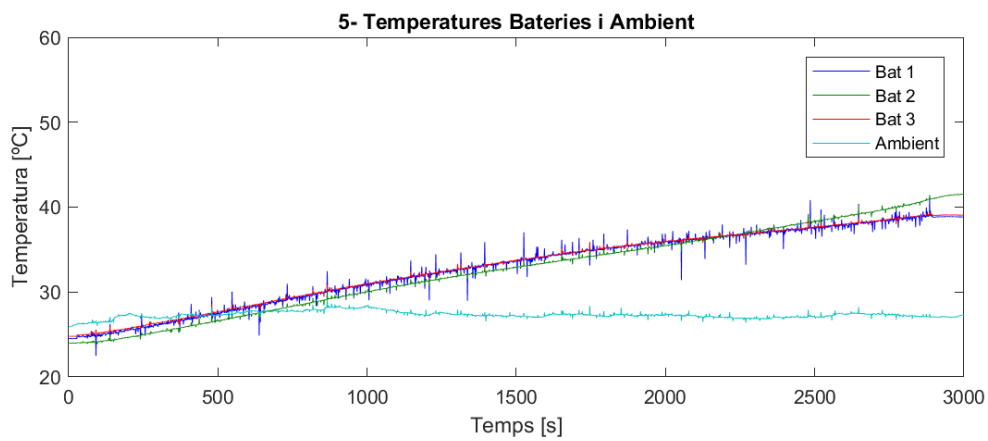


Figura C.5 – Gràfic de temperatures pel Cas 1a

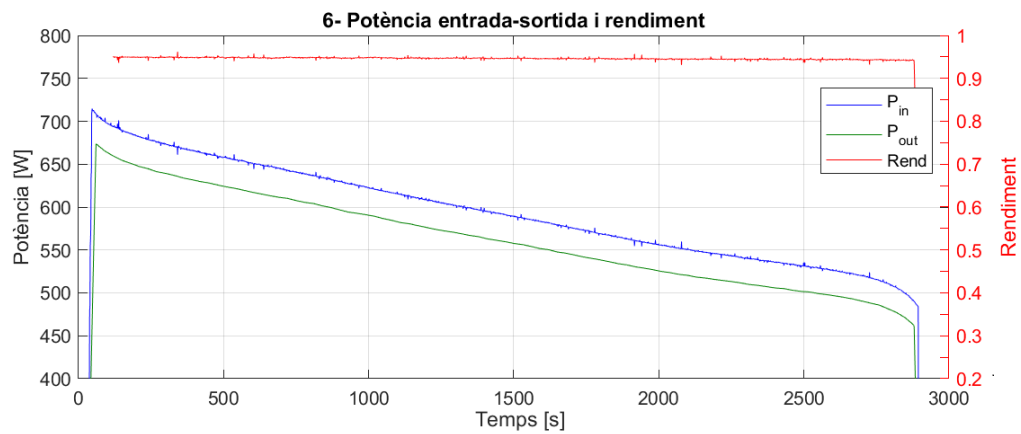


Figura C.6 – Gràfic de potències de contínua i alterna i rendiment pel Cas 1a

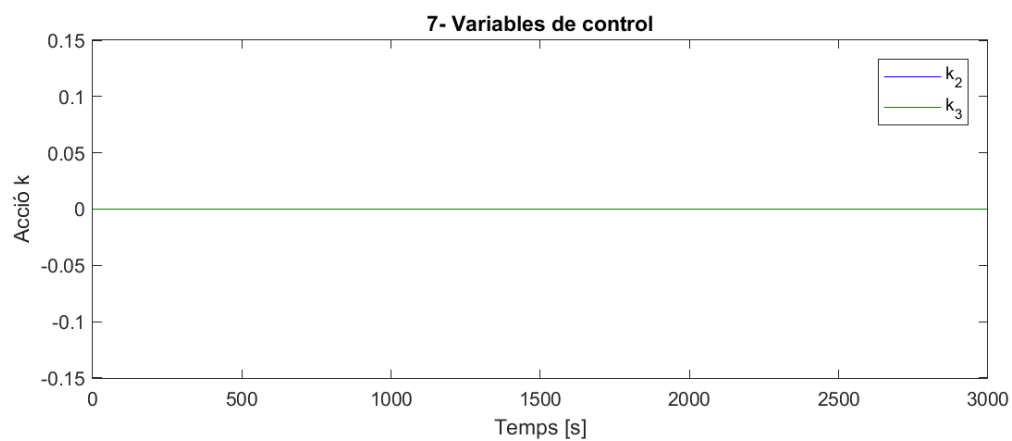


Figura C.7 – Gràfic d'accions de control pel Cas 1a

C.1.2. Cas 1b

Gràfics corresponents a les proves realitzades el 26/4/2018.

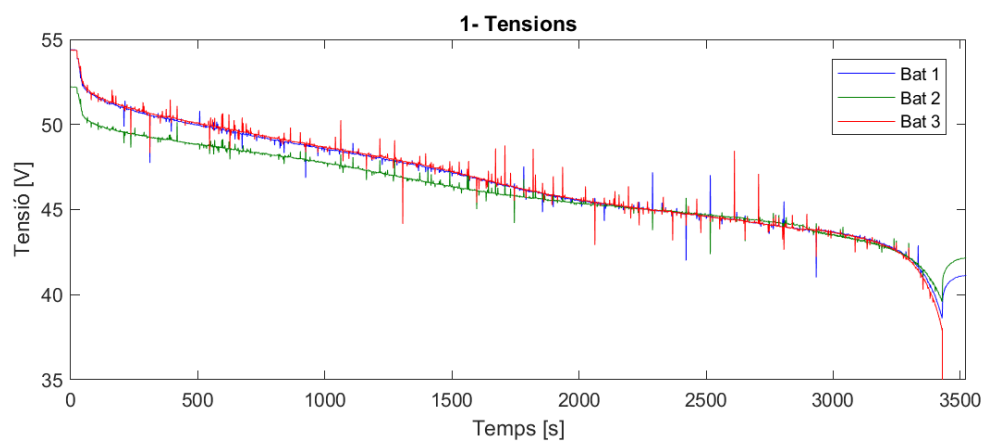


Figura C.8 – Gràfic de tensions pel Cas 1b

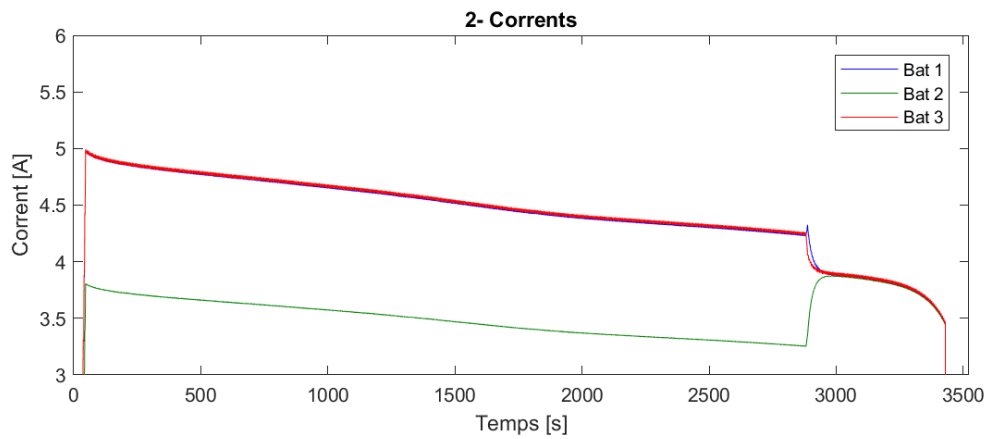


Figura C.9 – Gràfic de corrents pel Cas 1b

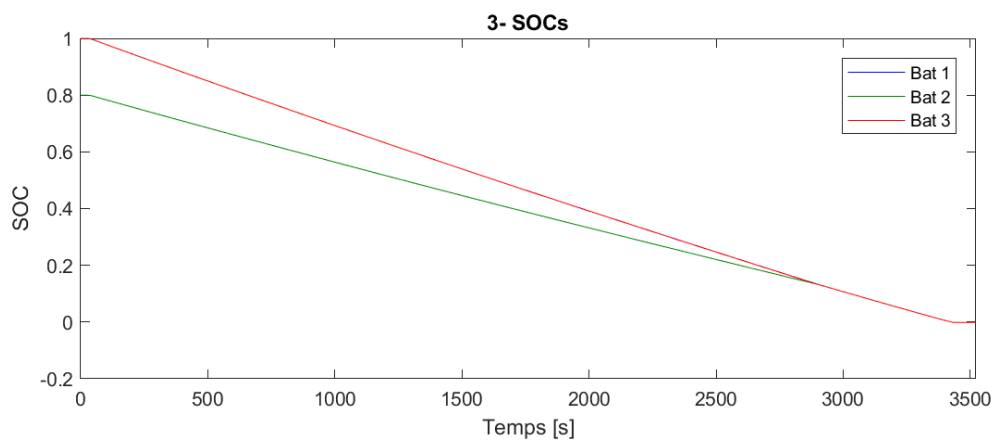


Figura C.10 – Gràfic d'estats de càrrega pel Cas 1b

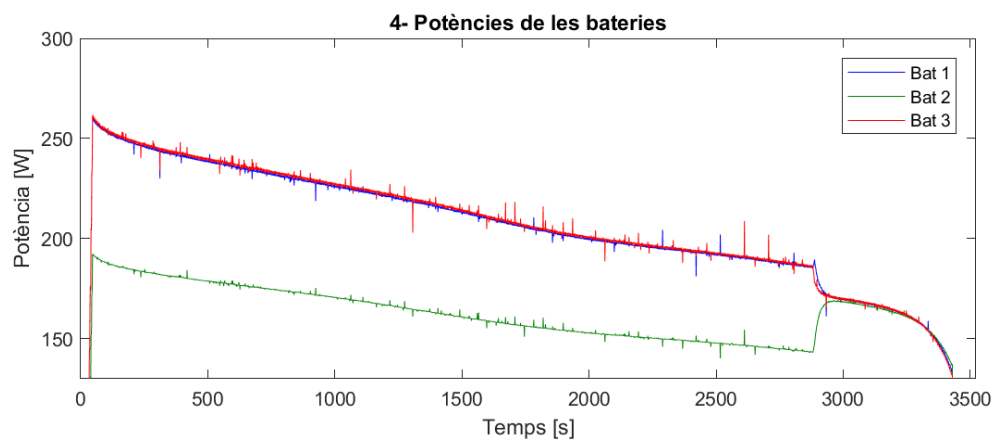


Figura C.11 – Gràfic de potències de les bateries pel Cas 1b

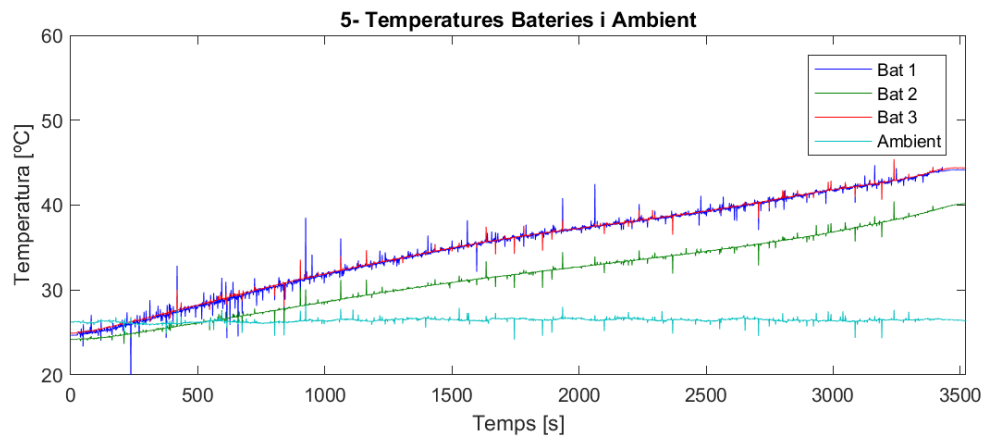


Figura C.12 – Gràfic de temperatures pel Cas 1b

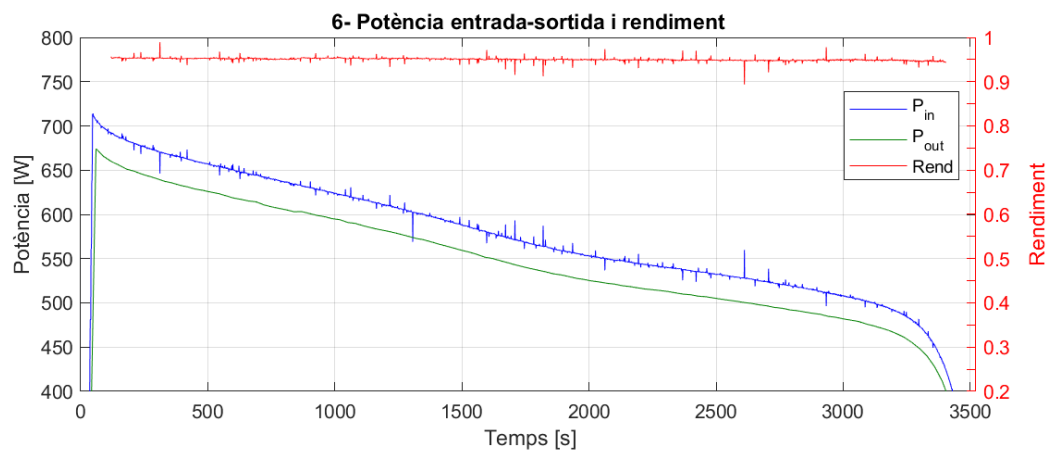


Figura C.13 – Gràfic de potències de contínua i alterna i rendiment pel Cas 1b

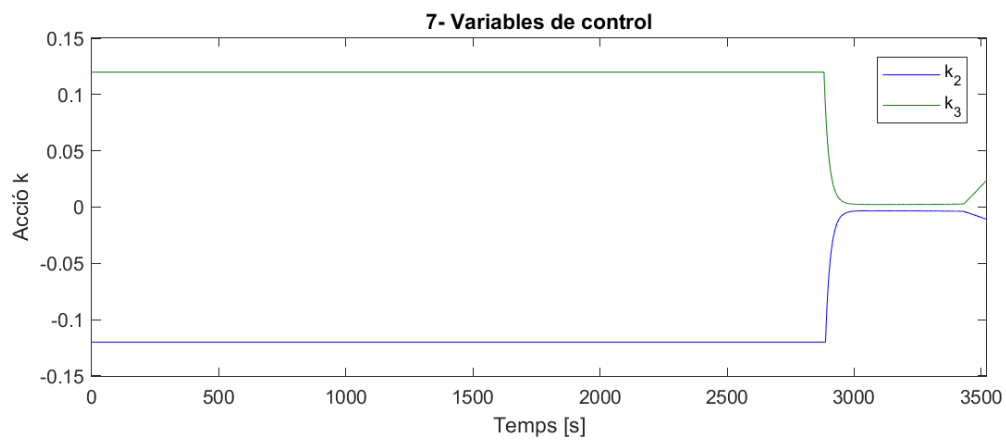


Figura C.14 – Gràfic d'accions de control pel Cas 1b

C.2. Cas 2

C.2.1. Cas 2a

Gràfics corresponents a les proves realitzades el 4/5/2018.

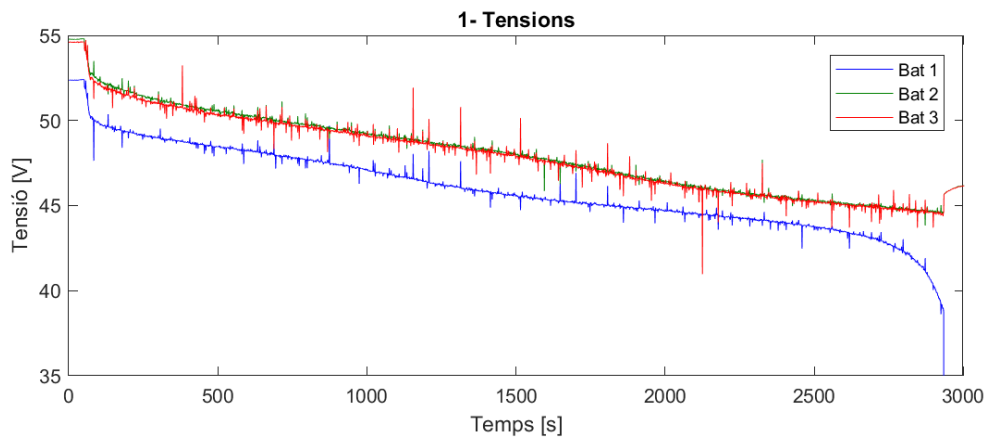


Figura C.15 – Gràfic de tensions pel Cas 2a

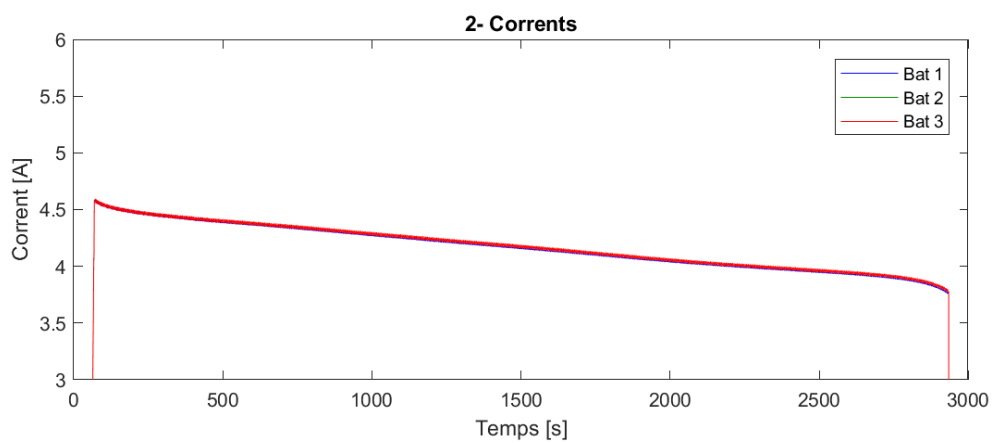


Figura C.16 – Gràfic de corrents pel Cas 2a

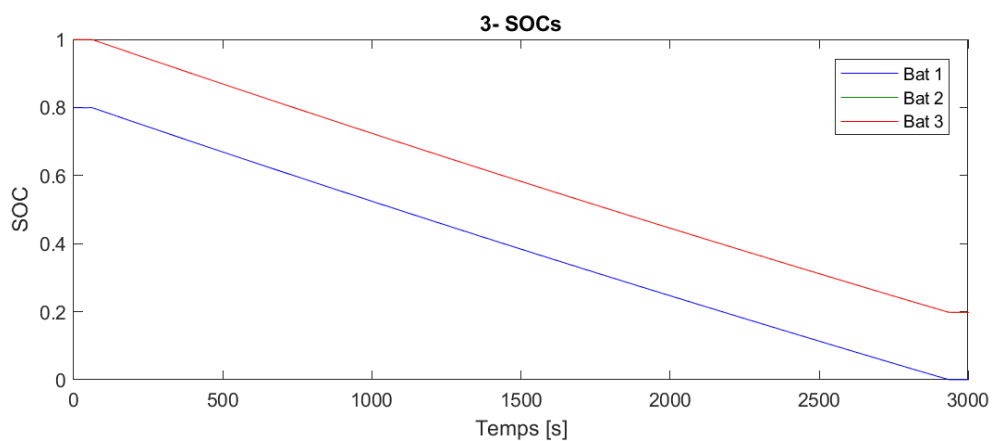


Figura C.17 – Gràfic d'estats de càrrega pel Cas 2a

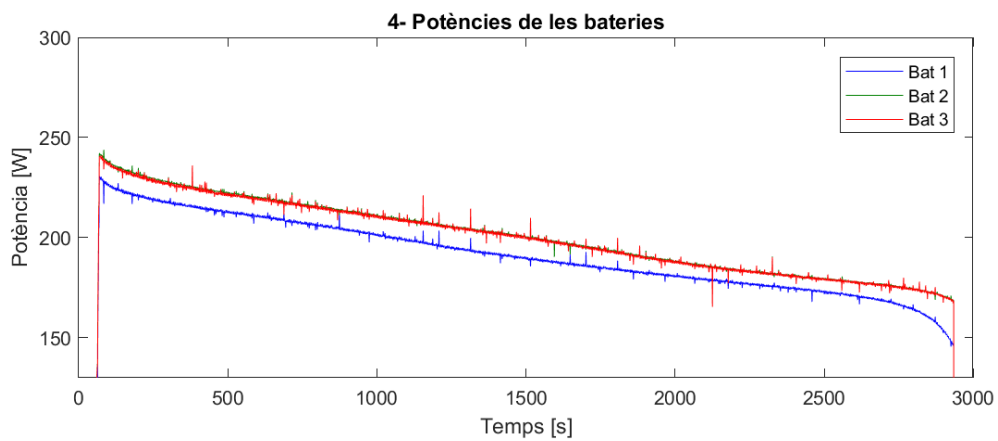


Figura C.18 – Gràfic de potències de les bateries pel Cas 2a

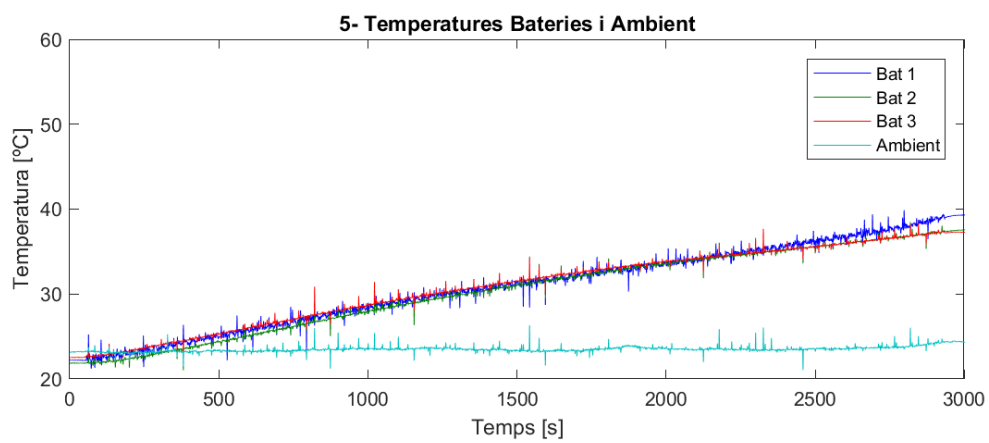


Figura C.19 – Gràfic de temperatures pel Cas 2a

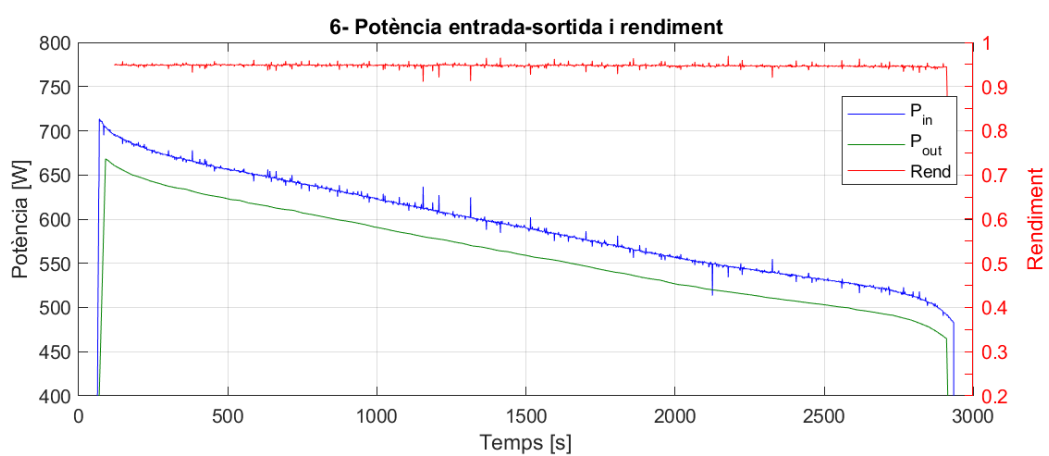


Figura C.20 – Gràfic de potències de contínua i alterna i rendiment pel Cas 2a

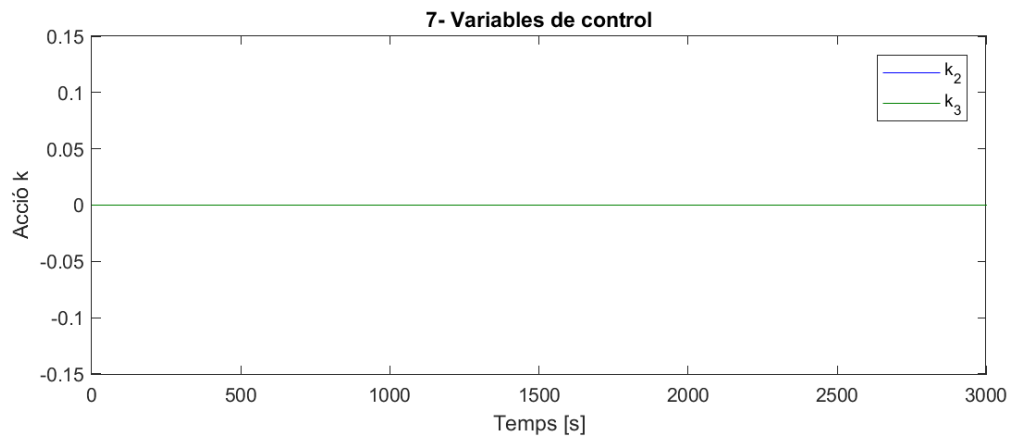


Figura C.21 – Gràfic d'accions de control pel Cas 2a

C.2.2. Cas 2b

Gràfics corresponents a les proves realitzades el 2/5/2018.

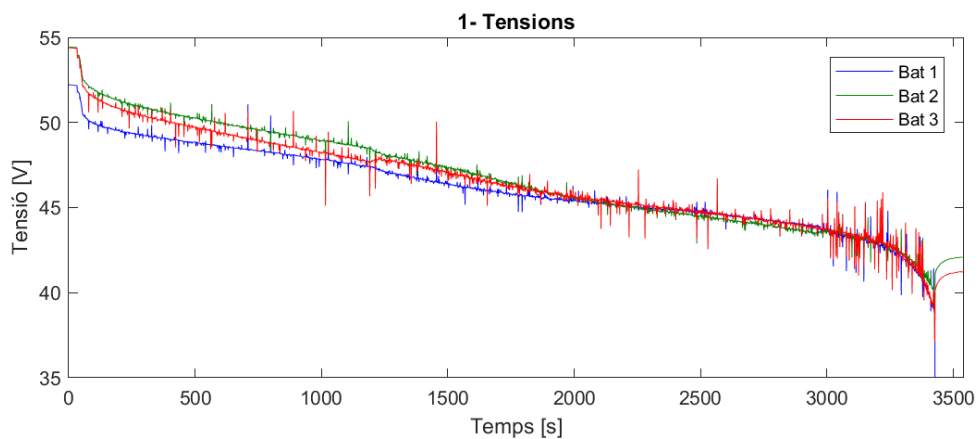


Figura C.22 – Gràfic de tensions pel Cas 2b

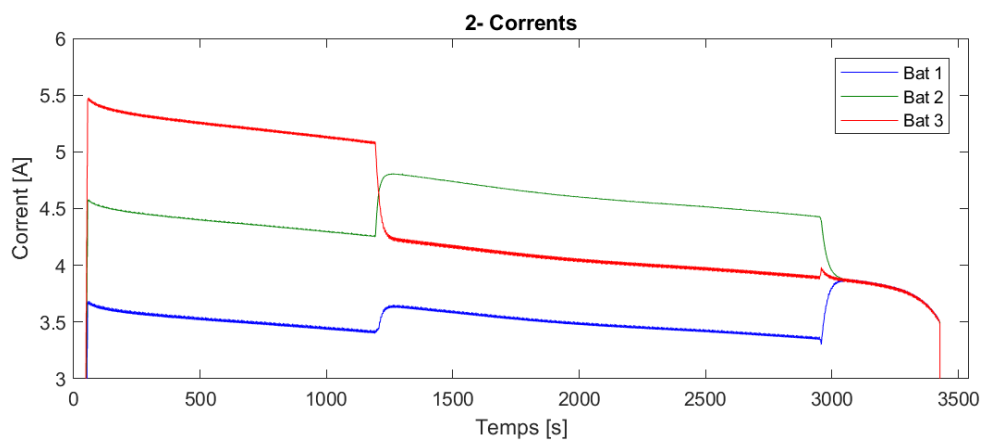


Figura C.23 – Gràfic de corrents pel Cas 2b

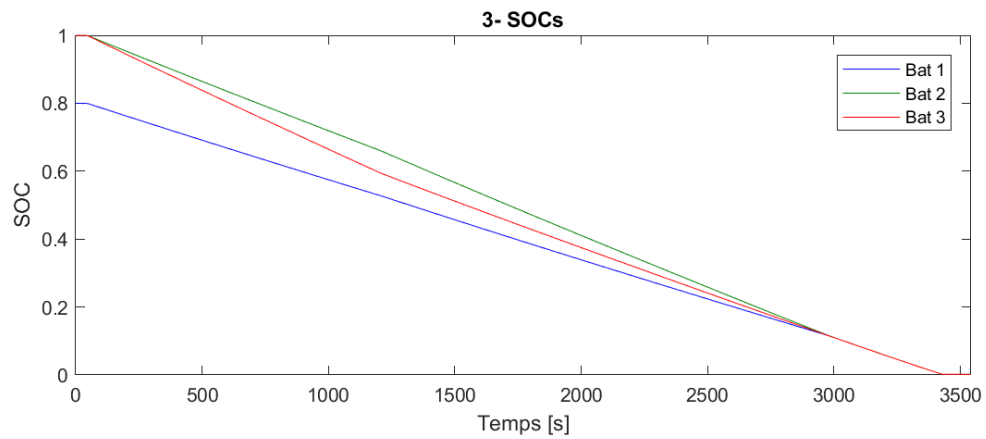


Figura C.24 – Gràfic d'estats de càrrega pel Cas 2b

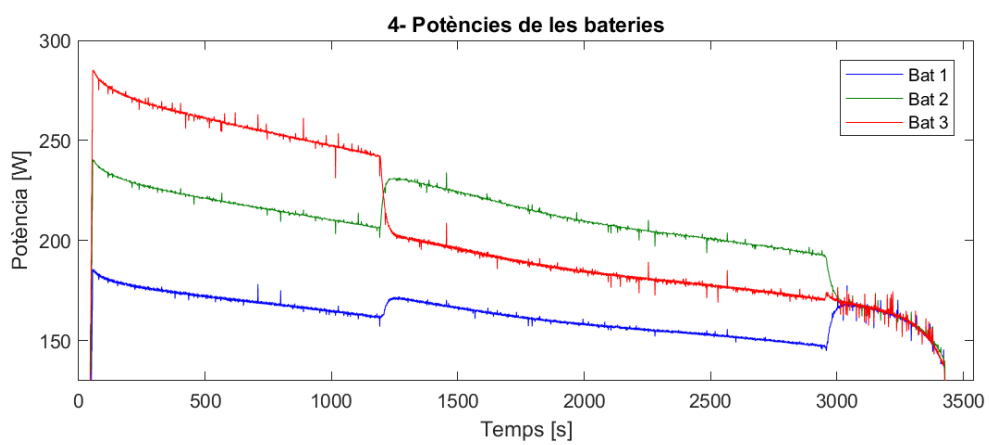


Figura C.25 – Gràfic de potències de les bateries pel Cas 2b

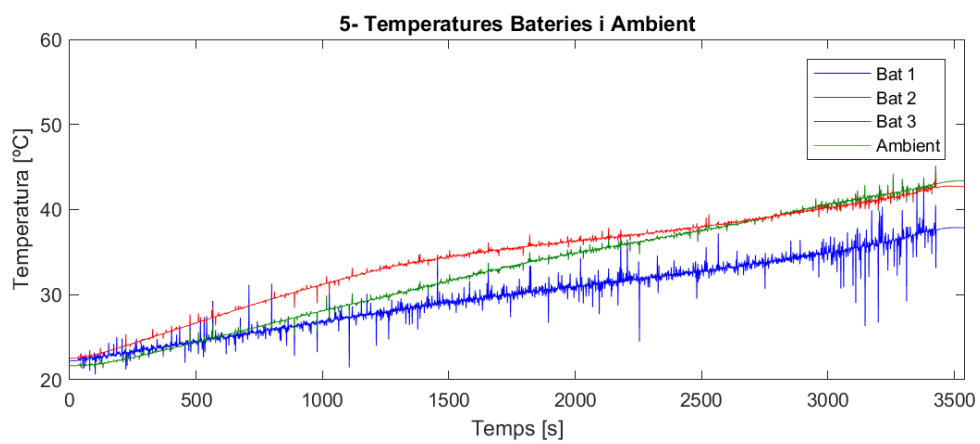


Figura C.26 – Gràfic de temperatures pel Cas 2b

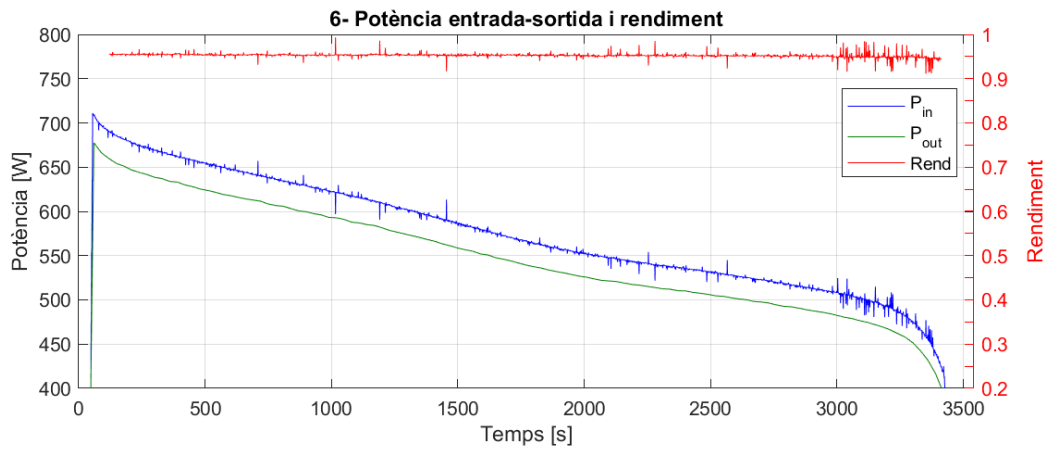


Figura C.27 – Gràfic de potències de contínua i alterna i rendiment pel Cas 2b

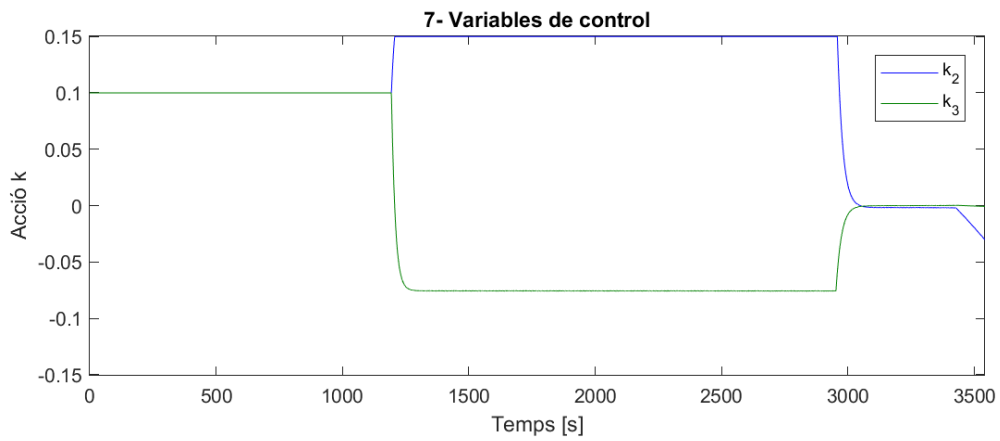


Figura C.28 – Gràfic d'accions de control pel Cas 2b

C.3. Cas 3

C.3.1. Cas 3a

Gràfics corresponents a les proves realitzades el 10/5/2018.

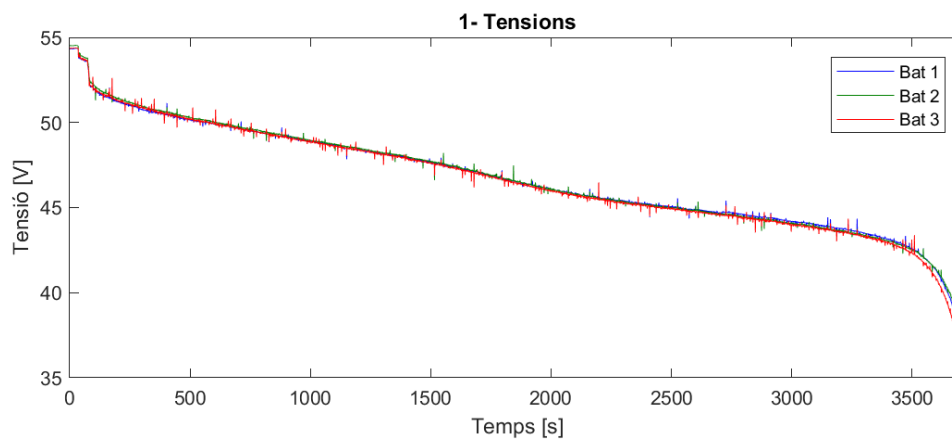


Figura C.29 – Gràfic de tensions pel Cas 3a

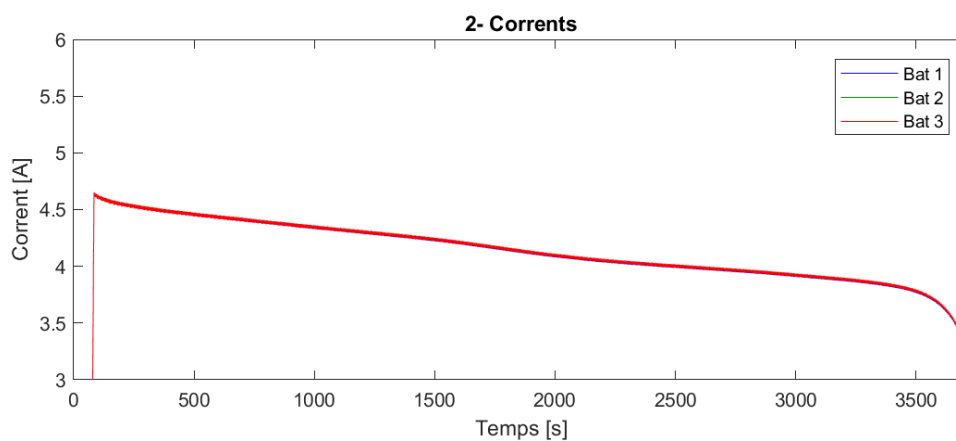


Figura C.30 – Gràfic de corrents pel Cas 3a

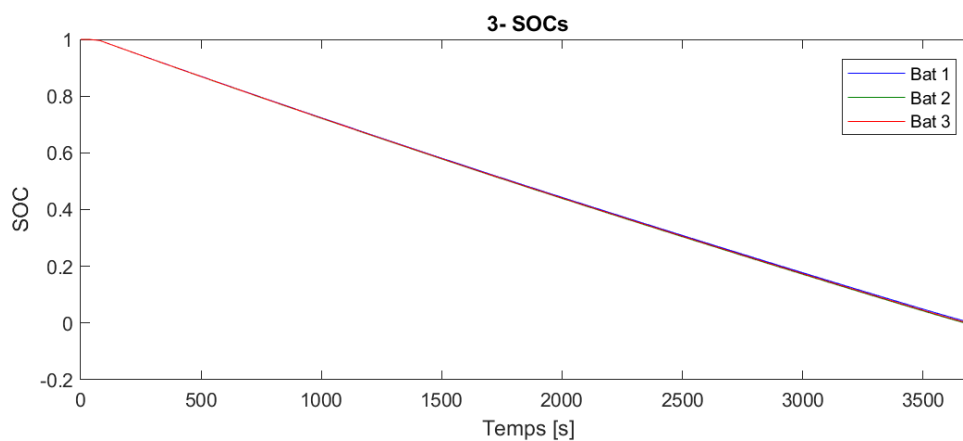


Figura C.31 – Gràfic d'estats de càrrega pel Cas 3a

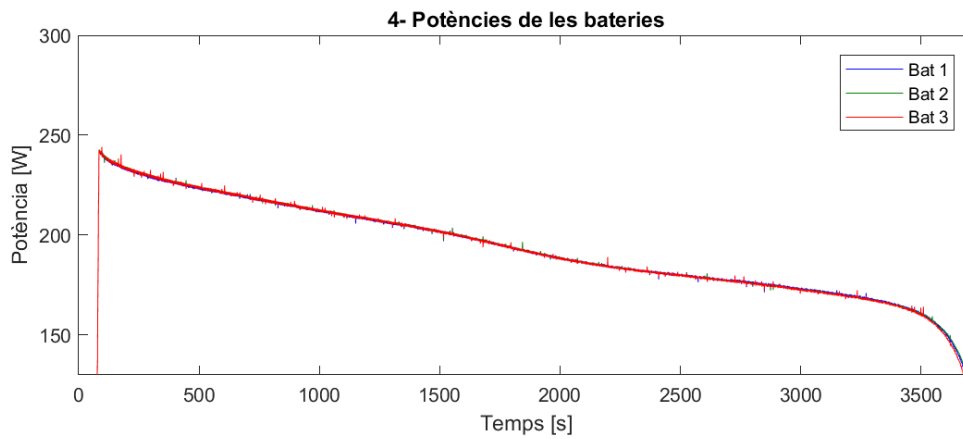


Figura C.32 – Gràfic de potències de les bateries del Cas 3a

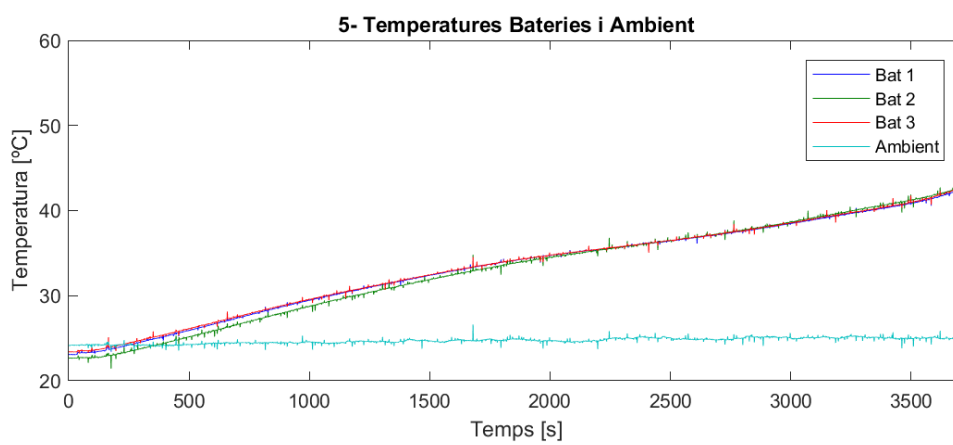


Figura C.33 – Gràfic de temperatures del Cas 3a

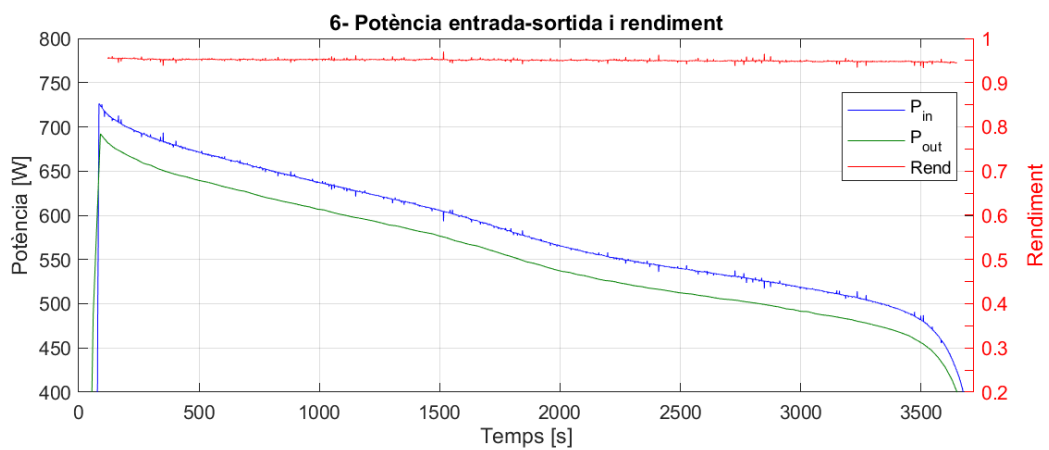


Figura C.34 – Gràfic de potències de contínua i d'alterna i rendiment del Cas 3a

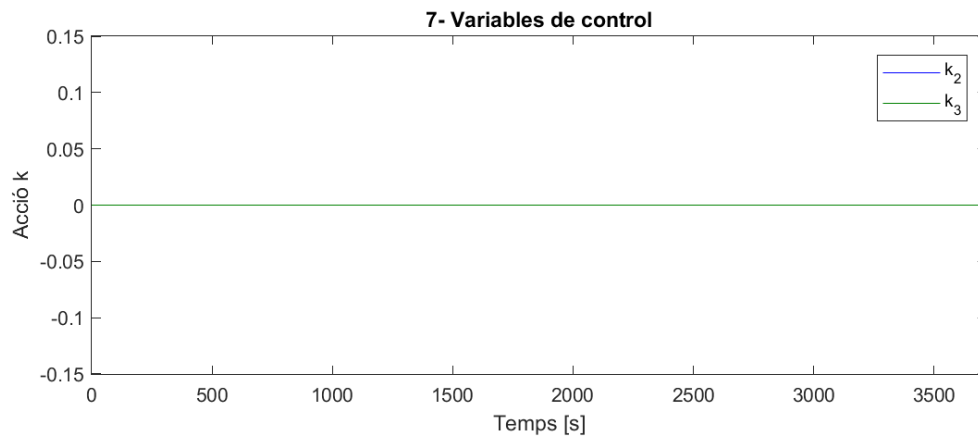


Figura C.35 – Gràfic d'accions de control pel Cas 3a

C.3.2. Cas 3b

Gràfics corresponents a les proves realitzades el 11/5/2018.

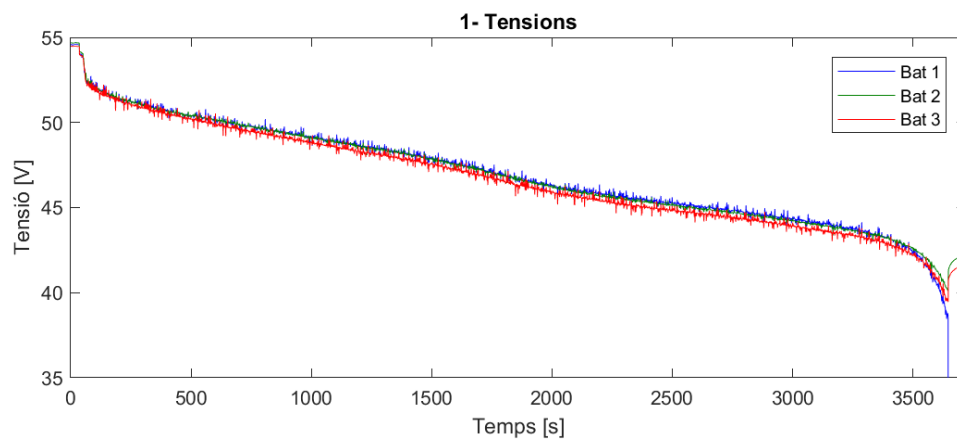


Figura C.36 – Gràfic de tensions pel Cas 3b

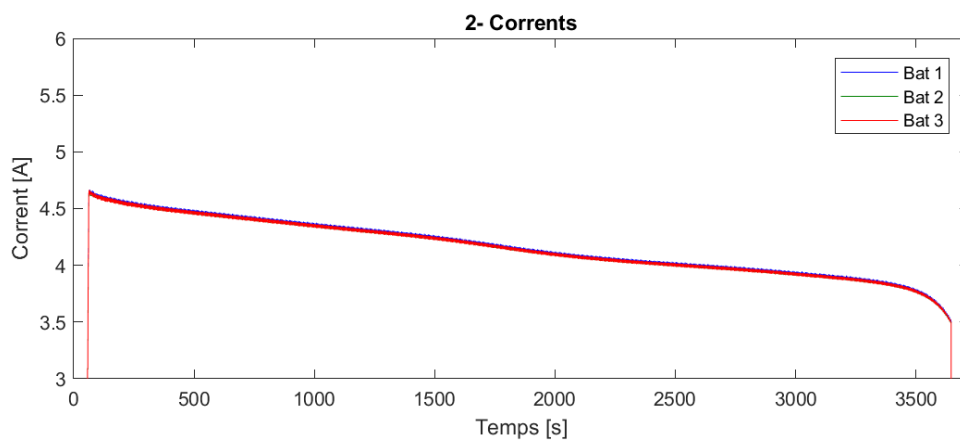


Figura C.37 – Gràfic de corrents pel Cas 3b

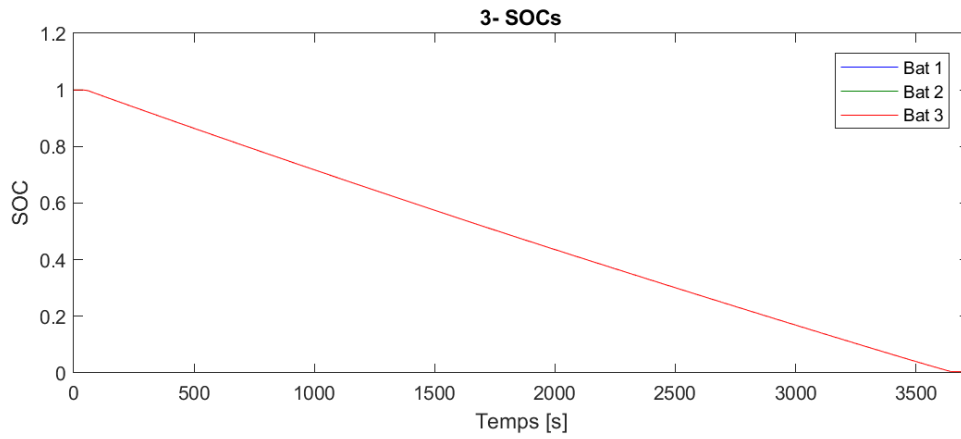


Figura C.38 – Gràfic d'estats de càrrega pel Cas 3b

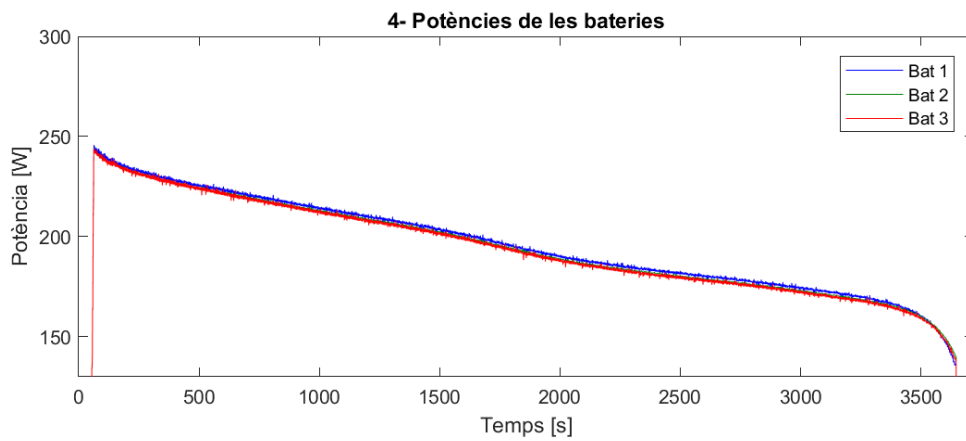


Figura C.39 – Gràfic de potències de les bateries pel Cas 3b

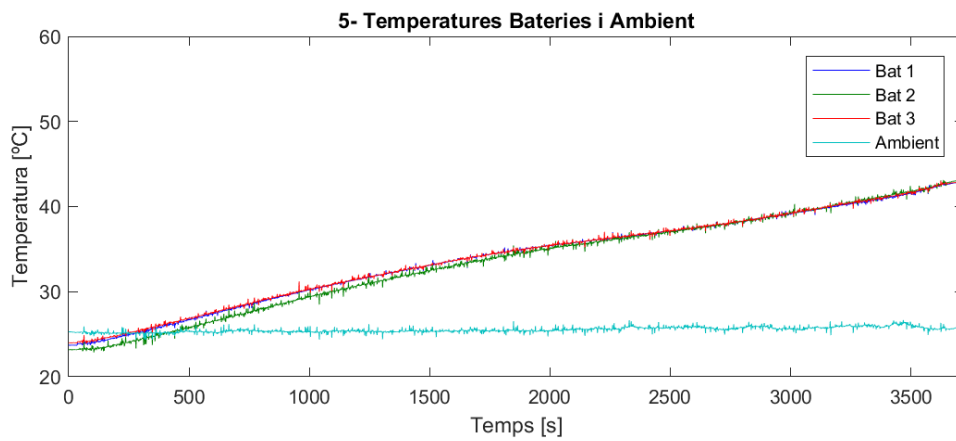


Figura C.40 – Gràfic de temperatures pel Cas 3b

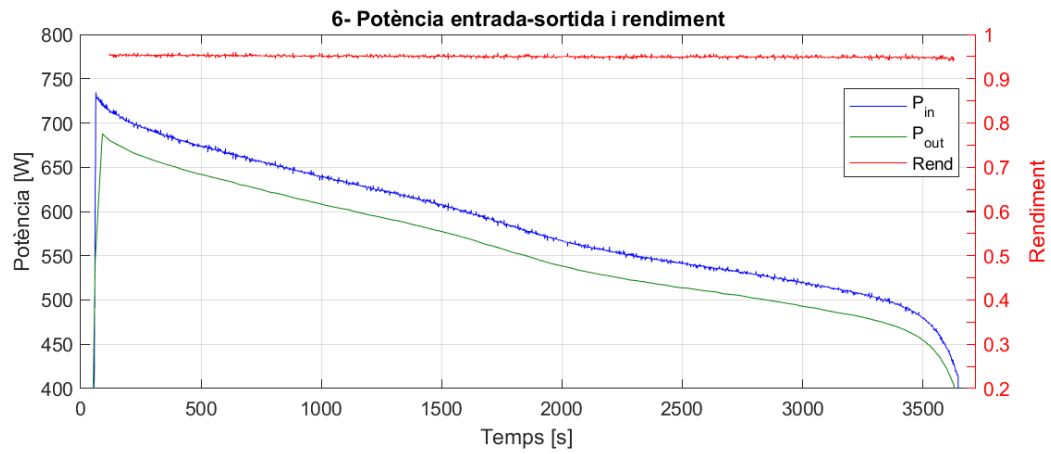


Figura C.41 – Gràfic de potències de contínua i alterna i rendiment pel Cas 3b

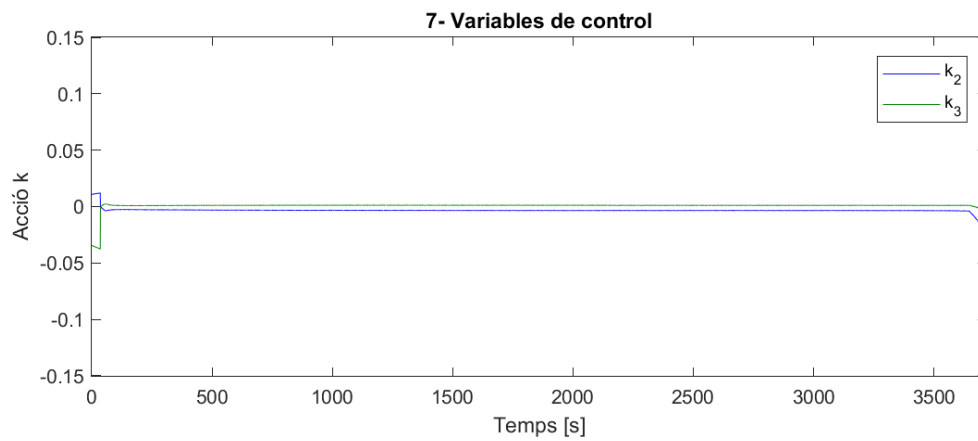


Figura C.42 – Gràfic d'accions de control pel Cas 3b