

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV ELEKTROENERGETIKY

System pro měření a sběr dat ze solárních systémů
Technická dokumentace

Funkční vzorek

Ing. Jan Macháček

Abstrakt

Předkládaný produkt je systém měření, sběru, analýzy a také vizualizace dat. Jedná se o experimentální aplikaci 100W fotovoltaického měniče, na kterém je pozorován vliv provozních podmínek na jeho zatěžovací charakteristiky. Tento FV panel a přidružený systém měření a sběru dat je nainstalován v solární laboratoři (na střešní terase) Ústavu elektroenergetiky FEKT VUT v Brně. Díky systému sběru, archivace a analýzy naměřených dat jsme schopni pozorovat, definovat a následně vyhodnotit veškeré vnější vlivy, které pozitivně ale především negativně ovlivňují účinnost zařízení.

Klíčová slova

Nekonvenční, alternativní, solární, fotovoltaické, měření.

Abstract

The present product is a system for measuring, collecting, analyzing, and displaying data. A 100W solar energy converter is used to monitor the effect of operating conditions on its loading characteristics. This photovoltaic (*PV*) cell and the associated measuring and data acquisition system have been installed in the solar laboratory (on a terraced roof) in the Department of Electrical Power Engineering, FEEC, BUT. Thanks to the system of collecting, archiving, and analyzing the data measured we can monitor, define, and subsequently interpret all positive and more significantly the negative external effects on the efficiency of the device.

Keywords

Unconventional, alternative, solar, photovoltaic, measuring.

Obsah

Abstrakt	ii
Klíčová slova	ii
Abstract	iii
Keywords	iii
Seznam ilustrací a obrázků	v
Seznam tabulek	vi
Seznam použitých symbolů a zkratk	vii
1 Úvod	1
2 Struktura systému měření, sběru a vizualizace dat	1
3 Ovládací program	2
4 Optimalizační obvod	3
5 Vyhodnocení naměřených dat a prezentace výsledků	4
Přílohy	P 1
Příloha A: Výpis zdrojového kódu měřicího skriptu.	P 2
Příloha B: Plošný spoj konektorového pole	P 18

Seznam ilustrací a obrázků

2.1	Schema systému.	2
4.1	Schéma zvyšujícího měniče.	4

Seznam tabulek

Seznam použitých symbolů a zkratek

Značka	Veličina	Značka jednotky
I	Proud	A
I_k	Proud nakrátko	A
P	Výkon	W
T_p	Teplota panelu	$^{\circ}C$
T_v	Teplota Vydechu	$^{\circ}C$
U	Napětí	V
U_0	Napětí naprázdno	V

1 Úvod

Na Ústavu elektroenergetiky Fakulty elektrotechniky a komunikačních technologií VUT v Brně bylo vyvinuto speciální experimentální měření na stowattovém solárním panelu pro sběr, archivaci a analýzu dat. Měření vzniklo především zpodnětu studentů doktorského studijního programu, kteří na FV panelu ověřují funkčnost zařízení, která vyvíjí vrámci svých disertačních prací. Studenti potřebovali vytvořit takový systém měření, který by jim umožnil nejen měření fyzikálních a elektrických veličin a archivaci naměřených dat, ale i jejich praktickou on-line vizualizaci. Což ve svém důsledku významně urychlilo odezvy na případné změny vkonfiguraci testovaných zařízení a zároveň umožnilo okamžitou prezentaci výsledků experimentu.

Zmiňovaný systém měření tvoří spojení dvou hlavních částí a to hardwarové a softwarové. Jelikož jsme vnašem případě měli omezené možnosti, co se nákupu průmyslově vyráběných měřících systémů týká, byli jsme nuceni použít stávající hardwarové a softwarové vybavení laboratoře. Uvedené se však nakonec ukázalo být výhodou, protože byl vyvinut takový systém, který přesně odpovídá našim požadavkům a nemuseli jsme činit žádné kompromisy nebo platit za hotový hardware a software jehož možnosti nevyužijeme.

Hardwarová část se skládá zcentrálního řídicího PC, měřící karty (NI PCI-6023E), optimalizačního obvodu smikroprocesorem a fakultního serveru.

zSoftwarovou část tvoří program, pomocí kterého PC komunikuje s měřící kartou a ovlivňuje průběh měření. Tímto programem je vnašem případě skript vývojového prostředí Matlab s Data Acquisition Toolboxem, dále program mikroprocesoru, který řídí samotnou optimalizaci a softwarové vybavení fakultního serveru, ze kterého využíváme Apache se serverovým skriptovacím jazykem PHP a relačním databázovým systémem MySQL.

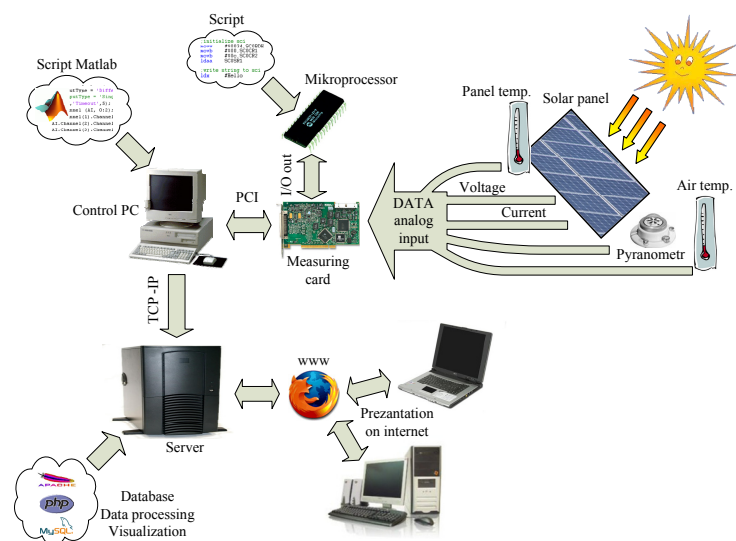
Systém jako celek je otevřený vůči různým dalším modifikacím, protože každá úloha má individuální požadavky a není nikterak náročné systém těmto požadavkům přizpůsobit. Stávající systém je odladěn pyro komplexní analýzu vlivu provozních podmínek na zatěžovací charakteristiky solárních měničů energie soptimalizací jejich zatížení.

2 Struktura systému měření, sběru a vizualizace dat

Jak je nastíněno vúvodní kapitole, náš systém slouží kměření a ukládání dat, jejichž zdrojem je solární panel. Jedná se o panel svýkonem 100 W, který je umístěn na střešní terase solární laboratoře Ústavu elektroenergetiky budovy U2 v areálu děkanátu VUT v Brně na ulici Údolní 53.

Sledovány jsou následující veličiny: intenzita slunečního záření a okamžitý výkon panelu, což jsou pro nás primární měřené údaje, a dále jsou měřeny teplota vzduchu

a teplota panelu, což jsou doplňující měřené veličiny. Teplotní čidlo měření teploty vzduchu je umístěno v ochranném "domečku", aby byl eliminován vliv proudění vzduchu, vliv deště a přímý sluneční osvit. Teplotní čidlo měřící teplotu panelu je umístěno prakticky uprostřed FV panelu a to na jeho vrchní (osvětlené) straně. Pro úplnost měříme teplotu i spodní (neosvětlené) strany panelu. Okamžitý výkon panelu měříme jak skutečný, tak optimalizovaný. Skutečný výkon je měřen v obvyklém zapojení FV panel a akumulátor. Struktura a tok dat jednotlivými fázemi systému je znázorněn viz Obr 2.1.



Obr. 2.1: Schema systému.

Na analogové vstupy měřící karty jsou včasové smyčce 180 vteřin přivedeny signály ze všech měřících čidel. Karta tyto signály digitalizuje a surová data předá ovládacímu programu (skriptu v Matlabu). Tento skript komunikuje nejenom směřící kartou a řídí tak celé měření, ale navíc pomocí digitálních výstupů karty ovládá chod optimalizačního obvodu. Aby bylo možno data přehledně archivovat a dále zpracovávat, jsou skriptem odeslána do MySQL databáze na server a celá smyčka se opakuje. Sdaty, která jsou uložena v databázi, je již možné nezávisle na probíhajícím měření dále pracovat, to znamená provádět například libovolné třídění a doplňující výpočty. Takto upravená a připravená data je možné prakticky v reálném čase prezentovat on-line na internetu.

3 Ovládací program

Řídicí skript výpočetního prostředí Matlab, pracuje prakticky v následujících krocích:

- běh programu je nastaven na nekonečnou smyčku, běžící v určených časových intervalech,
- nastavení doby měření, v našem případě nastaven začátek na 6:00 a konec v 21:00 hodin,

- inicializace měřicí karty,
- inicializace analogových vstupních kanálů,
- inicializace digitálních výstupních kanálů,
- nastavení měřicí frekvence a počtu vzorku v měřicí periodě,
- nastavení rozsahu kanálu,
- nastavení adresování isolačního multiplexeru,
- samotné měření,
- načtení hodnot z karty do proměnné,
- přepočítání hodnot dle měřené veličiny,
- vypnutí isolačního multiplexeru,
- ukládání dat do souboru,
- plnění databáze na serveru.

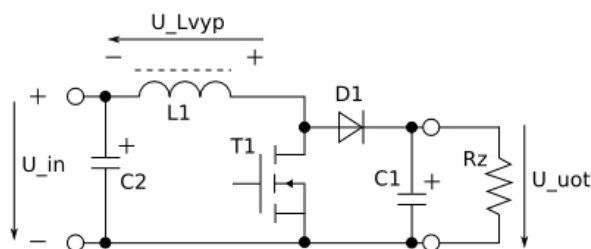
4 Optimalizační obvod

Vpředešlých kapitolách bylo několikrát naznačeno, že kromě standardního výkonu panelu je měřen také výkon panelu, kdy je optimalizováno jeho zatížení. Ve zkratce se jedná o tuto problematiku.

Fotovoltaický článek je v podstatě velkoplošná polovodičová dioda pracující v generátorickém režimu. Její voltampérová charakteristika má mezní veličiny U_0 a I_k . Tyto veličiny se v závislosti na změně sluneční intenzity a teploty článku mění. Násobením okamžitých hodnot napětí a proudu dostaneme charakteristiku výkonovou a nejvyšší bod na této charakteristice představuje maximální hodnotu možného výkonu, který je fotovoltaický článek schopen za daných provozních podmínek dodat do zátěže. Chceme-li tohoto maximálního výkonu dosáhnout musí být napětí na spotřebiči rovno právě napětí optimálnímu. Při déle trvající sluneční intenzitě nebo zhoršených podmínkách chlazení článku, dochází ke zvýšení povrchové teploty článku až na $80\text{ }^\circ\text{C}$. Při takto vysokých teplotách dochází ke změně elektrických vlastností článku, která vede ke snížení zatěžovací charakteristiky směrem k nižšímu napětí. Pokles optimálního napětí způsobí snížení dodávaného výkonu do akumulátoru. Jelikož k tomuto jevu dochází právě při největší sluneční intenzitě, můžeme ztratit významnou část z dosažitelné denní výroby fotovoltaického článku (v závislosti na stupni nabití akumulátoru).

Ke kompenzaci tohoto jevu může sloužit optimalizační zařízení, které pracuje na principu zvyšujícího DC/DC měniče viz Obr. 4.1.

Je-li výstupní napětí dáno úrovní nabití akumulátoru lze pulzním řízením tranzistoru T_1 snižovat vstupní napětí na kondenzátoru C_2 tak, aby odpovídalo právě optimálnímu provoznímu napětí fotovoltaického generátoru.



Obr. 4.1: Schéma zvyšujícího měniče.

Chod takového zařízení je právě testován vnaší solární laboratoři a kzáznamu a vyhodnocení dat byl zkonstruován náš měřicí systém.

5 Vyhodnocení naměřených dat a prezentace výsledků

Naměřená data nestačí jen uložit do databáze, ale je potřeba je zpracovat. To znamená provést, pokud jsou potřeba, nutné doplňující výpočty, data setřídít, výsledky vyhodnotit a zobrazit.

Vnašem případě provádíme několik doplňujících výpočtů. Kměřeným výkonům, intenzitě dopadajícího slunečního záření a teplotám dále počítáme:

- panelem vyrobenou denní energii,
- energetický zisk optimalizace zatížení,
- kumulativní a okamžitou účinnost přeměny energie FV panelu,
- měsíční souhrn panelem vyrobené energie,
- měsíční souhrn energetického zisku optimalizace zatížení,
- měsíční souhrn kumulativní a okamžité účinnosti přeměny energie FV panelu,
- roční souhrn panelem vyrobené energie zobrazené po měsících,
- celkový souhrn panelem vyrobené energie zobrazené po rocích.

Uvedené výpočty, jejich výsledky, grafy a další doplňující informace jsou prezentovány na internetových stránkách laboratoře. HTML kód stránek je generován skriptem jazyka PHP umístěným na serveru, který komunikuje s databází. Veškeré výpočetní operace se odehrávají na straně serveru, takže zobrazení stránek včetně grafů je velice rychlé a to i vnašem případě, kdy pracujeme a zobrazujeme řádově tisíce naměřených hodnot. Výsledky měření jsou, tomu kdo se o ně zajímá, snadno a okamžitě dostupné. Výhodou je, že prezentace výsledků měření je koncipována tak, aby bylo možno zobrazit nejen právě probíhající měření, ale i historii všech měření, která pro danou úlohu proběhla. Grafů naměřených a vypočtených hodnot jsou dostupné na: <http://www.ueen.feec.vutbr.cz/laboratory-of-unconventional-energy-conversion/>.

Přílohy

Příloha A: Výpis zdrojového kódu měřicího skriptu.

```

clear
a=1;      %hodnota pro nekonecnou smycku
prvni_spus=1;  %inicializace merici karty po spusteni
smaz=0; %zakazat smazani vseh promenych
pic=0;

while a==1      %nekonecna smycka ze ktere se neda vyskocit
    global test_ftp test_mput
% -----
    zacni=6; %v kolik hodin zacit
    skonci=21; %v kolik hodin skonci
    %pauza=90; %pauza mezi merenimi v sekundach
    nap=6.5555; %konstanta napetoveho delice
    pr2=4.16666; %konstanta bocniku nabijeni
    bocnik_ampermetru_a=0.056;
    bocnik_ampermetru_b=0.060;
    odpor_termo=1001;
    napeti_termo=4.9839;
    %napeti_termo=5;

% -----
    cas_mereni=clock; %precteni aktualniho casu
%-----inicializace a nastaveni karty-----
    if prvni_spus==1 | (inic==1 & cas_mereni(4)==(zacni-1))
        %Inicializace analogove casti karty
        AI = analoginput ('nidaq', 1);
        AI.InputType = 'Differential';
        %AI.InputType = 'SingleEnded'; % konfigurace kanalu na mereni proti zemi
        set(AI,'Timeout',5); %nastaveni casu pro odpoved merici karty
        addchannel (AI, 0:2); % zadani vstupnich kanalu
        AI.Channel(1).ChannelName = 'kanal_0'; %pojmenovani jednotlivych kanalu
        AI.Channel(2).ChannelName = 'kanal_1'; %pojmenovani jednotlivych kanalu
        AI.Channel(3).ChannelName = 'kanal_2'; %pojmenovani jednotlivych kanalu

        %AI.kanal_1.SensorRange = [-0.5 0.5]; %nastaveni vstupni citlivosti

```

```

%AI.kanal_2.InputRange = [-0.5 0.5]; %nastaveni vstupni citlivosti
%AI.kanal_2.UnitsRange = [0 1]; %nastaveni vstupni citlivosti

%Inicializace digitalni casti karty
dio = digitalio('nidaq',1);
addline(dio,0:7,'out');

%pocatecni hodnota dio portu
putvalue(dio.Line(1),0);
putvalue(dio.Line(2),0);
putvalue(dio.Line(3),0);
putvalue(dio.Line(4),0);
putvalue(dio.Line(5),0);
putvalue(dio.Line(6),0);
putvalue(dio.Line(7),0);
putvalue(dio.Line(8),0);

%nastavit horní mez kmitočtové citlivosti
Fs = 55000; % frekvence mereni
set (AI, 'SampleRate', Fs);
duration = 0.01; % doba trvani mereni v sekundach
set (AI, 'SamplesPerTrigger', duration*Fs);

inic=0; %zakazat inicializaci merici karty
prvni_spus=0; %zakazat prvni spusteni
end
%-----
cas_mereni=clock; %precteni aktualniho casu
%*****smycka pro mereni*****
while cas_mereni(4)>=zacni & cas_mereni(4)<skonci
    o=1;
    n=1;
    a=1;
    p=1;

AI.kanal_1.InputRange =[0 0.05]; %rozsah pro mereni proudu do akumulatoru
AI.kanal_2.InputRange =[0 10]; %rozsah pro mereni pyranometru

```

```
for a=1:4
    %vypnutí a zapnutí napajeni modulu optimalizace pro synchronizaci s PC (prepi
    putvalue(dio.Line(7),1);    % prenastaveni merici pozice
    pause(3);    % pro reset picu
    putvalue(dio.Line(7),0);    % prenastaveni merici pozice
%%    pause(0.3);    % pro rele
%%    putvalue(dio.Line(8),1);    % synchronizace s picem
%%    pause(1.3);    % pauza pro synchronizaci
%%    putvalue(dio.Line(8),0);    % synchronizace s picem

%pause(1.8);
pause(3.8);

%mux_1 proud na panelu_1
AI.kanal_0.InputRange =[0 0.5];
putvalue(dio.Line(1:4),[1 0 0 0]);    %nastaveni muxu

mereni_opti;

Ipanel_a_op(a)=1.0325*pom_prumer(1,1)*pom_prumer(1,1)+16.3*pom_prumer(1,1);
    if Ipanel_a_op(a)<0.01
        Ipanel_a_op(a)=0;
    else
        end

    putvalue(dio.Line(1:4),[0 0 0 0]);

%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 1 1 0]);    %nastaveni muxu

mereni_opti;

V24=pom_prumer(1,1)*nap;
pom_Iaku_op=mean(Iaku_op);
Wopti(p)=V24*pom_Iaku_op;
```

```
putvalue(dio.Line(1:4), [0 0 0 0]);
putvalue(dio.Line(7), 1);    % prenastaveni merici pozice
pause(0.3);

%mux_2 proud na panelu_2
putvalue(dio.Line(1:4), [0 1 0 0]);    %nastaveni muxu

mereni_neopti;

Ipanel_b_nop(a)=0.6496*pom_prumer(1,1)*pom_prumer(1,1)+14.27*pom_prumer(1,1)
if Ipanel_b_nop(a)<0.01
Ipanel_b_nop(a)=0;
else
end

putvalue(dio.Line(1:4), [0 0 0 0]); % prepnuti multiplexeru do nulove pozice

%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4), [1 1 1 0]);    %nastaveni muxu

mereni_neopti;

V24=pom_prumer(1,1)*nap;
putvalue(dio.Line(1:4), [0 0 0 0]);

%mux_9 mereni napeti na akumulatoru
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4), [1 0 0 1]);    %nastaveni muxu

mereni_neopti;

V12a=pom_prumer(1,1)*nap;
pom_Iaku_nop=mean(Iaku_nop);
Wneopti(p)=V12a*pom_Iaku_nop;
V12b=V24-V12a;
```



```
p=p+1;

    putvalue(dio.Line(1:4),[0 0 0 0]);
pause(1);    % pro reset picu
    putvalue(dio.Line(7),0);    % pre nastaveni merici pozice
%%    pause(0.3);    % pro rele
%%    putvalue(dio.Line(8),1);    % synchronizace s picem
%%    pause(1.3);    % pauza pro synchronizaci
%%    putvalue(dio.Line(8),0);    % synchronizace s picem

%pause(1.7);    % pauza pro zoptimalizovani
pause(3.7);
    %druhe mereni
    %mux_3 napeti na panelu
    AI.kanal_0.InputRange =[0 5];
    putvalue(dio.Line(1:4),[1 1 0 0]);    %nastaveni muxu

    mereni_opti;

    Vpanel_op(a)=pom_prumer(1,1)*nap;
    if Vpanel_op(a)<0.5
        Vpanel_op(a)=0;
    else
        end
    putvalue(dio.Line(1:4),[0 0 0 0]);

    %mux_7 mereni napeti na akumulatorech
    AI.kanal_0.InputRange =[0 5];
    putvalue(dio.Line(1:4),[1 1 1 0]);    %nastaveni muxu

    mereni_opti;

    V24=pom_prumer(1,1)*nap;
    pom_Iaku_op=mean(Iaku_op);
    Wopti(p)=V24*pom_Iaku_op;
```

```
putvalue(dio.Line(1:4), [0 0 0 0]);
putvalue(dio.Line(7), 1);    % prenastaveni merici pozice
pause(0.3);

%mux_3 napeti na panelu
putvalue(dio.Line(1:4), [1 1 0 0]);    %nastaveni muxu

mereni_neopti;

Vpanel_nop(a)=pom_prumer(1,1)*nap;
if Vpanel_nop(a)<0.5
Vpanel_nop(a)=0;
else
end
putvalue(dio.Line(1:4), [0 0 0 0]);

%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange = [0 5];
putvalue(dio.Line(1:4), [1 1 1 0]);    %nastaveni muxu

mereni_neopti;

V24=pom_prumer(1,1)*nap;
putvalue(dio.Line(1:4), [0 0 0 0]);

%mux_9 mereni napeti na akumulatoru
AI.kanal_0.InputRange = [0 5];
putvalue(dio.Line(1:4), [1 0 0 1]);    %nastaveni muxu

mereni_neopti;

V12a=pom_prumer(1,1)*nap;
pom_Iaku_nop=mean(Iaku_nop);
Wneopti(p)=V12a*pom_Iaku_nop;
V12b=V24-V12a;

p=p+1;
```

```
    putvalue(dio.Line(1:4),[0 0 0 0]);
    pause(1);    % pro reset picu

%%        pause(0.3);    % pro rele

    if V12a<13
    putvalue(dio.Line(6),0);
    else
    putvalue(dio.Line(6),1);
    end

    if V12b<13
    putvalue(dio.Line(5),0);
    else
    putvalue(dio.Line(5),1);
    end

    putvalue(dio.Line(7),0);    % pre nastaveni merici pozice
%%        pause(0.3);    % pro rele
%%        putvalue(dio.Line(8),1);    % synchronizace s picem
%%        pause(1.3);    % pauza pro synchronizaci
%%        putvalue(dio.Line(8),0);    % synchronizace s picem

%pause(1.6);
pause(3.6);    % pauza pro zoptimalizovani

    %treti mereni
    %mux_4 mereni teploty v boudicce
    AI.kanal_0.InputRange =[0 5];
    putvalue(dio.Line(1:4),[0 0 1 0]);    %nastaveni muxu

    mereni_opti;

    cidlo_bu=pom_prumer(1,1);
    odpor_bu=cidlo_bu/((napeti_termo-cidlo_bu)/odpor_termo);
```

```
teplota_bu(a)=0.000000003355*odpor_bu*odpor_bu*odpor_bu-0.000032216336*odpor
teplota_bu(a)=teplota_bu(a)-0.3;

putvalue(dio.Line(1:4),[0 0 0 0]);

%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 1 1 0]); %nastaveni muxu

mereni_opti;

V24=pom_prumer(1,1)*nap;
pom_Iaku_op=mean(Iaku_op);
Wopti(p)=V24*pom_Iaku_op;

putvalue(dio.Line(1:4),[0 0 0 0]);
putvalue(dio.Line(7),1); % prenastaveni merici pozice
pause(0.3);

%mux_5 mereni teploty na panelu
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 0 1 0]); %nastaveni muxu

mereni_neopti;

cidlo_pa=pom_prumer(1,1);
odpor_pa=cidlo_pa/((napeti_termo-cidlo_pa)/odpor_termo);
teplota_pa(a)=0.000000003355*odpor_pa*odpor_pa*odpor_pa-0.000032216336*odpor
%teplota_pa(a)=teplota_pa(a)+1;

putvalue(dio.Line(1:4),[0 0 0 0]);

%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 1 1 0]); %nastaveni muxu

mereni_neopti;
```

```
V24=pom_prumer(1,1)*nap;
putvalue(dio.Line(1:4),[0 0 0 0]);

%mux_9 mereni napeti na akumulatoru
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 0 0 1]); %nastaveni muxu

mereni_neopti;

V12a=pom_prumer(1,1)*nap;
pom_Iaku_nop=mean(Iaku_nop);
Wneopti(p)=V12a*pom_Iaku_nop;
V12b=V24-V12a;

p=p+1;

putvalue(dio.Line(1:4),[0 0 0 0]);
pause(1); % pro reset picu
putvalue(dio.Line(7),0); % prenastaveni merici pozice
%% pause(0.3); % pro rele
%% putvalue(dio.Line(8),1); % synchronizace s picem
%% pause(1.3); % pauza pro synchronizaci
%% putvalue(dio.Line(8),0); % synchronizace s picem

%pause(1.5);
pause(3.5); % pauza pro zoptimalizovani
%ctvrte mereni
%mux_6 mereni teploty pod panelem
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[0 1 1 0]); %nastaveni muxu

mereni_opti;

cidlo_pod=pom_prumer(1,1);
odpor_pod=cidlo_pod/((napeti_termo-cidlo_pod)/odpor_termo);
teplota_pod(a)=0.000000003355*odpor_pod*odpor_pod*odpor_pod-0.000032216336*
```

```
%teplota_pod(a)=teplota_pod(a)+1;

putvalue(dio.Line(1:4),[0 0 0 0]);

%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 1 1 0]); %nastaveni muxu

mereni_opti;

V24=pom_prumer(1,1)*nap;
pom_Iaku_op=mean(Iaku_op);
Wopti(p)=V24*pom_Iaku_op;

putvalue(dio.Line(1:4),[0 0 0 0]);
putvalue(dio.Line(7),1); % prenastaveni merici pozice
pause(0.3);

%mux_4 mereni teploty v boudicce
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[0 0 1 0]); %nastaveni muxu

mereni_neopti;

cidlo_bu=pom_prumer(1,1);
odpor_bu=cidlo_bu/((napeti_termo-cidlo_bu)/odpor_termo);
teplota_bu(a+1)=0.000000003355*odpor_bu*odpor_bu*odpor_bu-0.000032216336*odp
teplota_bu(a+1)=teplota_bu(a+1)-0.3;

putvalue(dio.Line(1:4),[0 0 0 0]);

%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 1 1 0]); %nastaveni muxu

mereni_neopti;
```

```

V24=pom_prumer(1,1)*nap;
putvalue(dio.Line(1:4),[0 0 0 0]);

%mux_9 mereni napeti na akumulatoru
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 0 0 1]); %nastaveni muxu

mereni_neopti;

V12a=pom_prumer(1,1)*nap;
pom_Iaku_nop=mean(Iaku_nop);
Wneopti(p)=V12a*pom_Iaku_nop;
V12b=V24-V12a;

p=p+1;

putvalue(dio.Line(1:4),[0 0 0 0]);
pause(1); % pro reset picu
putvalue(dio.Line(7),0); % prenastaveni merici pozice
%% pause(0.3); % pro rele
%% putvalue(dio.Line(8),1); % synchronizace s picem
%% pause(1.3); % pauza pro synchronizaci
%% putvalue(dio.Line(8),0); % synchronizace s picem

%pause(1.7);
pause(3.6); % pauza pro zoptimalizovani
%pate mereni
%mux_2 proud na panelu_2
AI.kanal_0.InputRange =[0 0.5];
putvalue(dio.Line(1:4),[0 1 0 0]); %nastaveni muxu

mereni_opti;

Ipanel_b_op(a)=0.6496*pom_prumer(1,1)*pom_prumer(1,1)+14.27*pom_prumer(1,1);
if Ipanel_b_op(a)<0.01
Ipanel_b_op(a)=0;
else

```

```
end

putvalue(dio.Line(1:4),[0 0 0 0]);

%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 1 1 0]); %nastaveni muxu

mereni_opti;

V24=pom_prumer(1,1)*nap;
pom_Iaku_op=mean(Iaku_op);
Wopti(p)=V24*pom_Iaku_op;

putvalue(dio.Line(1:4),[0 0 0 0]);
putvalue(dio.Line(7),1); % prenastaveni merici pozice
pause(0.3);

%mux_1 proud na panelu_1
putvalue(dio.Line(1:4),[1 0 0 0]); %nastaveni muxu

mereni_neopti;

Ipanel_a_nop(a)=1.0325*pom_prumer(1,1)*pom_prumer(1,1)+16.3*pom_prumer(1,1);
if Ipanel_a_nop(a)<0.01
Ipanel_a_nop(a)=0;
else
end

putvalue(dio.Line(1:4),[0 0 0 0]); % prepnuti multiplexeru do nulove pozice
%mux_7 mereni napeti na akumulatorech
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4),[1 1 1 0]); %nastaveni muxu

mereni_neopti;

V24=pom_prumer(1,1)*nap;
```



```
putvalue(dio.Line(1:4), [0 0 0 0]);

%mux_9 mereni napeti na akumulatoru
AI.kanal_0.InputRange =[0 5];
putvalue(dio.Line(1:4), [1 0 0 1]); %nastaveni muxu

mereni_neopti;

V12a=pom_prumer(1,1)*nap;
pom_Iaku_nop=mean(Iaku_nop);
Wneopti(p)=V12a*pom_Iaku_nop;
V12b=V24-V12a;

p=p+1;

putvalue(dio.Line(1:4), [0 0 0 0]);
% putvalue(dio.Line(7),0); % prenastaveni merici pozice
end

%maxima
max_Iaku_nop=max(Iaku_nop);
max_Wneopti=max(Wneopti);
max_pyranom_nop=max(pyranom_nop);
max_Iaku_op=max(Iaku_op);
max_Wopti=max(Wopti);
max_pyranom_op=max(pyranom_op);

%minima
min_Iaku_nop=min(Iaku_nop);
min_Wneopti=min(Wneopti);
min_pyranom_nop=min(pyranom_nop);
min_Iaku_op=min(Iaku_op);
min_Wopti=min(Wopti);
min_pyranom_op=min(pyranom_op);

%prumer
prum_Iaku_nop=mean(Iaku_nop);
```

```

prum_Wneopti=mean(Wneopti);
prum_pyranom_nop=mean(pyranom_nop);
prum_Iaku_op=mean(Iaku_op);
prum_Wopti=mean(Wopti);
prum_pyranom_op=mean(pyranom_op);
prum_Ipanel_a_op=mean(Ipanel_a_op);
prum_Ipanel_b_op=mean(Ipanel_b_op);
prum_Ipanel_a_nop=mean(Ipanel_a_nop);
prum_Ipanel_b_nop=mean(Ipanel_b_nop);
prum_Vpanel_op=mean(Vpanel_op);
prum_Vpanel_nop=mean(Vpanel_nop);
prum_teplota_bu=mean(teplota_bu);
prum_teplota_pa=mean(teplota_pa);
prum_teplota_pod=mean(teplota_pod);

%prumer celkovy
pyranom=(prum_pyranom_nop+prum_pyranom_op)/2;

%skusit upravit vybijeni na 11V

%-----
cas_mereni=clock;
hodiny=cas_mereni(4);
minuty=cas_mereni(5);

%---vytvoreni souboru s nazvem -podle data
rok=num2str(cas_mereni(1));
mesic=num2str(cas_mereni(2));
den=num2str(cas_mereni(3));
datum=[den,'_',mesic,'_',rok,'.txt'];
%-----
s = fopen(datum,'a'); %otevreni souboru pro pripisovani dat mer_
fprintf(s,'%02d:%02d % 6.2fV % 6.2fA % 6.2fA % 6.2fV % 6.2fA %6.2fW % 6.2fW
fprintf(s,' % 6.2fV % 6.2fA % 6.2fA % 6.2fV % 6.2fA % 6.2fW % 6.1fC % 6.
fclose(s); %zavreni souboru

%ukladani dat do souboru

```

```

m = fopen('dat.txt','a');    %otevreni souboru pro pripisovani dat
fprintf(m, '''%s-%s-%s'' ,%d,%d,%1.1f,%1.1f,%1.2f,%1.2f,%1.2f\n',rok,mesic,den)
fclose(m);    %zavreni souboru

%-----
%ukladani zalohy dat do souboru
z = fopen('zaloha_mesic.txt','a');    %otevreni souboru pro pripisovani dat
fprintf(z, '''%s-%s-%s'' ,%d,%d,%1.1f,%1.1f,%1.2f,%1.2f,%1.2f\n',rok,mesic,den)
fclose(z);    %zavreni souboru

%ukladani souboru na ftp server
f_tipko = ftp('palcat.feec.vutbr.cz','datalabueen','piopa7ac');
if test_ftp==1
    cd (f_tipko, '/');
    cd (f_tipko, 'home2/web/UEEN/laboratory-of-unconventional-energy-convers
    mput(f_tipko,'dat.txt');
    close(f_tipko); %zavreni ftp spojeni
end

if V12a<13
putvalue(dio.Line(6),0);
else
putvalue(dio.Line(6),1);
end

if V12b<13
putvalue(dio.Line(5),0);
else
putvalue(dio.Line(5),1);
end

% tic;
% while toc<=pauza
% end

if test_ftp==1
    if test_mput==1
        delete('dat.txt');

```

```
        end
    end
    cas_mereni=clock;    %precteni aktualniho casu
    smaz=1; %povolit smazani vsech promenych
end

%*****

%dat sem mereni akumulatoru pres noc a jeho vybitu na hodnotu 11V pokud se
%nestihne jeho vybiti ukoncit vybijeni hodinu pred zahajenim mereni
%podminku smaz povolit az tehdy budeli vybito na 11V nebo bude li hodinu
%pred merenim. jeste zkontrolovat podminku u smycky s merenim

    if smaz==1
        putvalue(dio.Line(6),0);
        putvalue(dio.Line(5),0);
        delete (AI);
        delete (dio)
        clear    %smazani vsech promenych
        smaz=0; %zakazat smazani vsech promenych
        inic=1; %povolit inicializaci merici karty
        prvni_spus=0;    %zakazat prvni spusteni
        a=1;
    end

end

end
```

Příloha B: Plošný spoj konektorového pole

