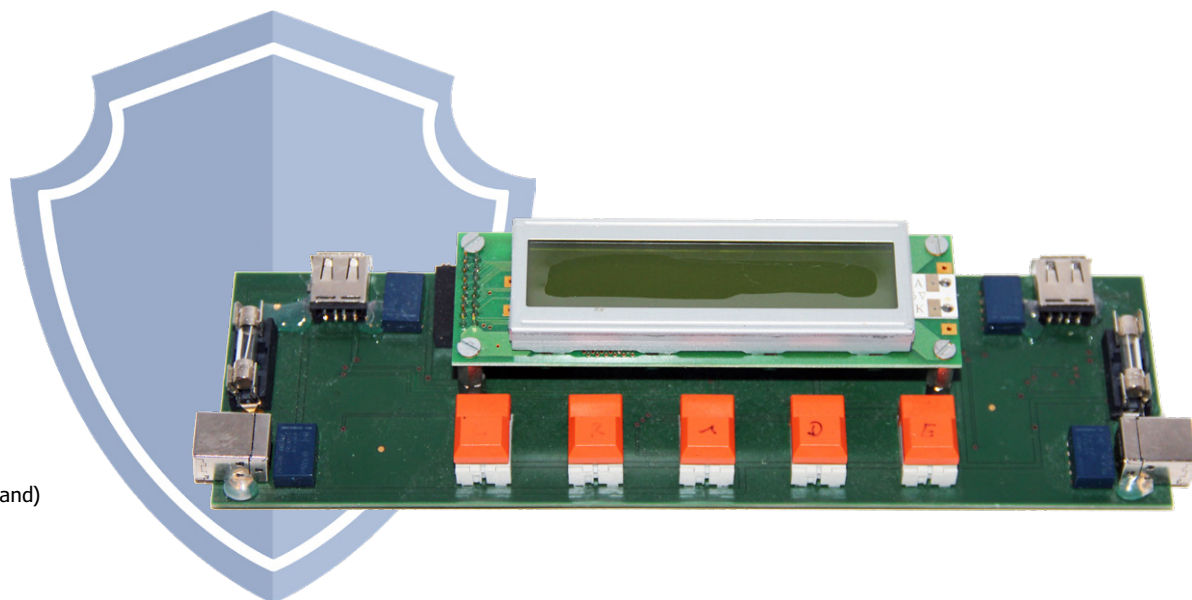


USB-stroombewaking

intelligente bescherming voor USB-poorten



René Gräber (Duitsland)

Ik kwam op het idee voor dit project toen ik aan het experimenteren was met een zelfgebouwde schakeling die (zoals wel vaker) werd gevoed uit de 5V-USB-poort van mijn computer. Ik kreeg een foutmelding van het besturingssysteem: "Stroompiek op Hubpoort". Helaas liep de communicatie met de schakeling ook over deze USB-poort, wat het foutzoeken er niet bepaald makkelijker op maakte.

Ik had dus iets nodig dat de voedingsstroom van een aangesloten USB-apparaat meet, maar die niet onmiddellijk onderbreekt als er kort een grenswaarde wordt overschreden. Maar tegelijk moet de computer wel beschermd blijven, terwijl de USB-communicatie overeind blijft, ook met een externe voeding. Hier is mijn oplossing.

Uit (treurige) ervaringen met voeding uit USB wist ik al dat de meestal opgegeven 500 mA niet altijd in acht genomen wordt, en dat afdoende stroombegrenzing voor USB-poorten op een PC meestal ontbreekt.

De ontwikkeling van dit project verliep in twee stappen. De eerste versie bestond uit meerdere 'losse' printen die via headers met elkaar verbonden waren. Die versie is gemaakt in Eagle en zo uitgevoerd dat de prints met een frees konden worden gemaakt. Daarna kwam een tweede ontwerp voor één enkele print, dat ik gemaakt heb met de CAD-software van DesignSpark. De schakeling van dit tweede ontwerp ziet u in **figuur 2**.

Over de hardware

J4 en J5 zijn twee poorten waarop u apparaten via USB kunt aansluiten. K2 resp. K3 verbindt u met de USB-poort van de computer. De stroom wordt steeds gemeten met een INA219 van Texas Instruments [1]. De INA219 is een current/power monitor met I²C-interface die cyclisch wordt uitgelezen. Er is steeds één stroommonitor per USB-kanaal. Via de shuntweerstand R19 en R30 wordt de spanning gemeten, waaruit de stroom door de shunts kan worden berekend. R22 t/m R25 en R33 t/m R36 bepalen het I²C-adres.

Met de relais Re1 en Re3 kan de stroomtoevoer naar aangesloten apparaten worden ingeschakeld en onderbroken. De zekeringen F1 en F2 dienen daarbij als beveiliging tegen te

hoge stroom. De relais worden aangestuurd door de microcontroller U1.

Enige toelichting is hierbij op zijn plaats. Met een belastingsweerstand aan de NC-pen van de relais krijgen we stabilere spanningswaarden, wat de conversie door de microcontroller betrouwbaarder maakt. Bij open relais kan het anders tot verkeerde waarden leiden. In de tweede versie van het project komen deze weerstanden nog niet voor.

Via Re2 en Re4 kan het aangesloten USB-apparaat van een externe 5-V-voedingsspanning worden voorzien (in plaats van de 5 V uit de computer). Deze spanning komt binnen via J2 en voedt ook de rest van de schakeling.

Het hart van de schakeling is een PIC18F66220 microcontroller [2]. LED1 t/m LED4 geven de status van het apparaat weer en geven eventuele waarschuwingen of storingen aan. De microcontroller meet aan de ADC-ingangen AN1 en AN2 de beide spanningen die de USB-poorten leveren. Ook meet de PIC die spanning aan J2 en aan de ingang AN0 (EXT_PWR). Voor deze drie spanningsmetingen wordt steeds een spanningsdelers met een verhouding van 1 : 4 gebruikt (R13 t/m R18). De deling door vier is met opzet zo gekozen, omdat dan de microcontroller de juiste spanningswaarde kan berekenen met een simpele shift-operatie van twee bits. Een MAX6004 [3] dient als spanningsreferentie, hij levert een referentiespanning van 4,096 V voor de interne 10-bit A/D-converter.

De toetsen S1 t/m S5 zijn verbonden met respectievelijk de interrupt- en de keyboard-pennen van de microcontroller en worden in de firmware ontdekkend en uitgelezen.

J1 is de ICP-programmeer-interface (In-Circuit Programming), terwijl JP5 een 5V-UART-poort is. Hierover kan de communicatie met een PC verlopen.

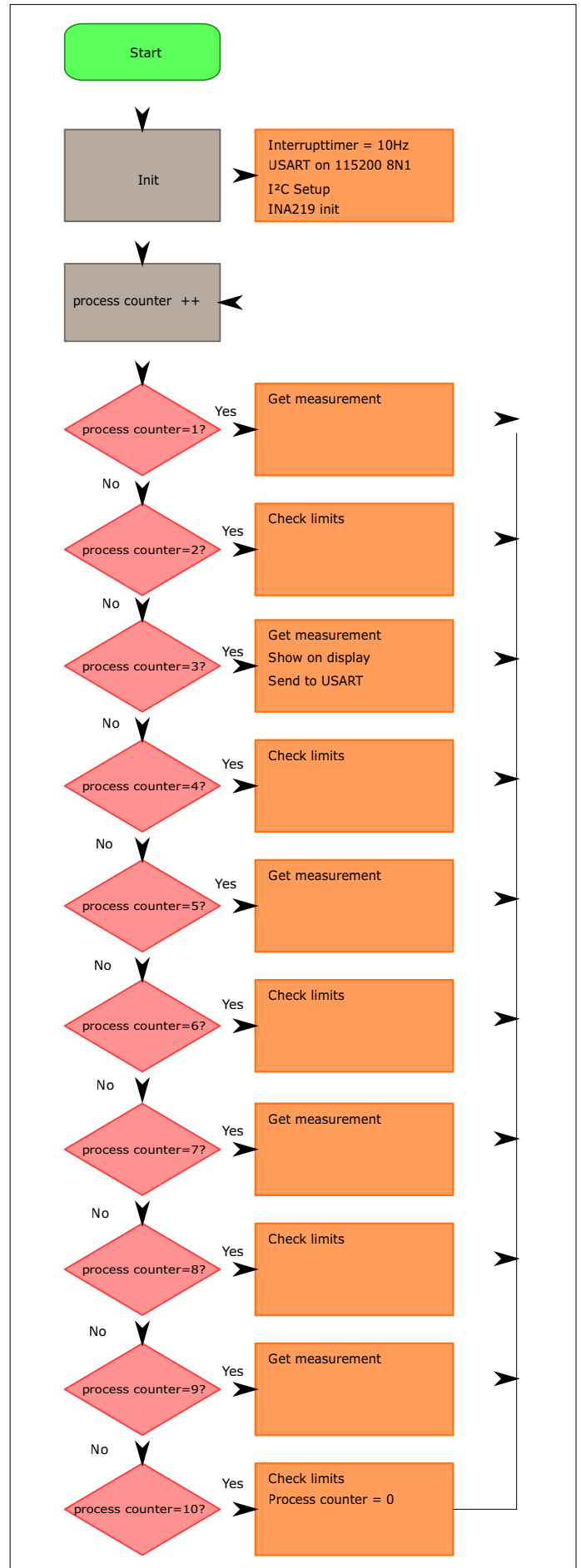
De meetwaarden worden weergegeven op een standaard-display van 2x20 karakters. De aansturing loopt via een vierdraads-verbinding voor de data en drie stuurlijnen voor RS, R/W en E. Via R10 is het contrast in te stellen.

Zonder software lukt het niet

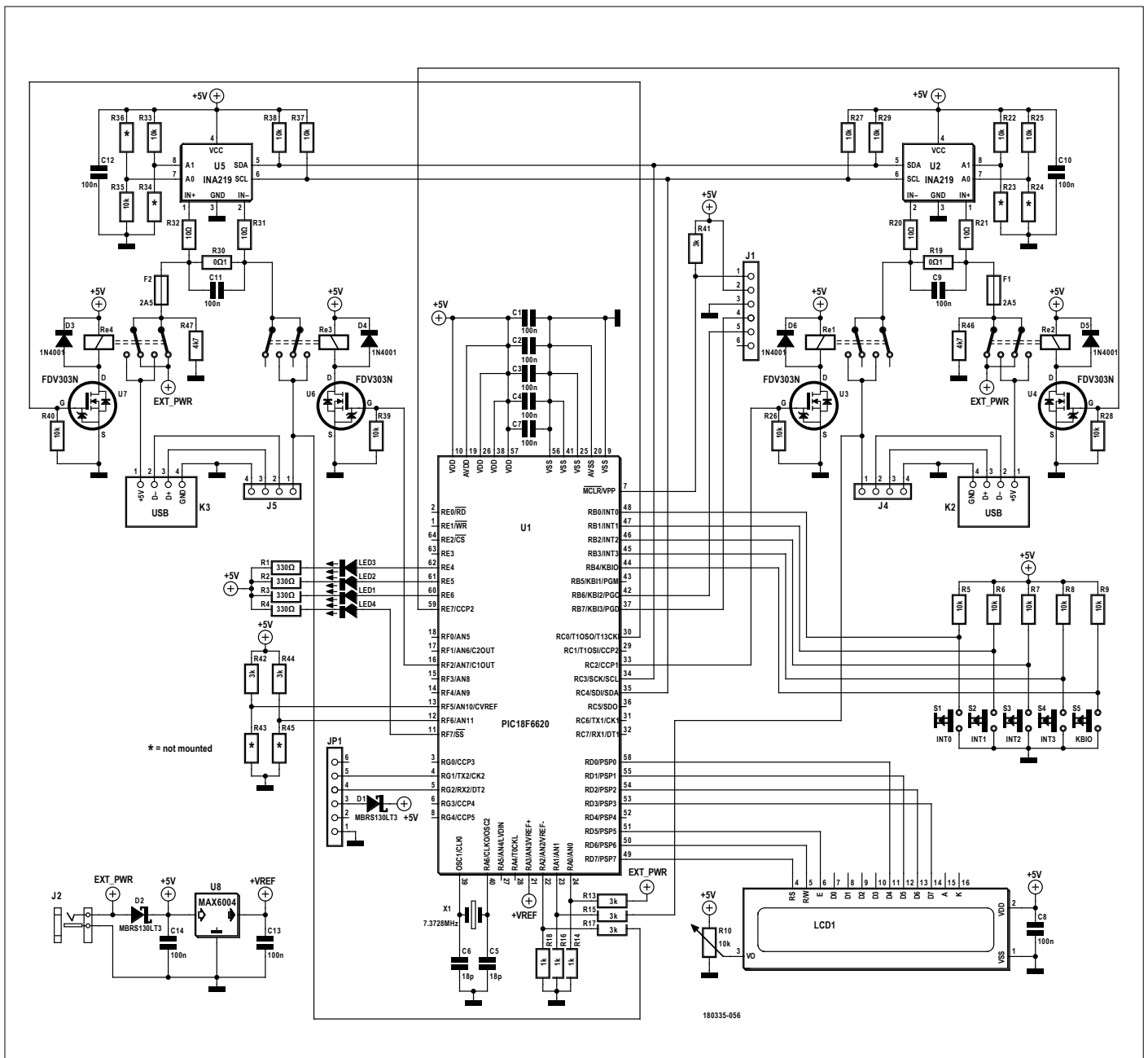
De firmware is geschreven in C met de IDE MPLAB X. In **figuur 1** ziet u het stroomdiagram. Bij het opstarten moet

Listing 1. Berekening van de waarden voor het calibration register van de INA219.

```
/*Berechnung Calibration Register laut Datenblatt
 * R-Shunt = 0R1
 * V-BusMax 6V
 * I - Max 2A2
 * Min LSB = 2A2/32767 => 0,00006714
 * Max LSB = 2A2/4096 => 0,0005371
 * Choose an LSB in the range: Minimum_LSB
 * Current_LSB = 0,00007
 * Cal = trunc[0,04096/(Current_LSB * R-Shunt)] =
0,04096/(0,00007*0r1) => 5851
 * Power_LSB = 20 * Current_LSB = 20 * 0,00007 =
0,0014
 * Max_Current = Current_LSB * 32767 = 0,00007 *
32767 => 2,29369A
 */
```



Figuur 1. Het stroomschema maakt duidelijk wat de controller-firmware doet.



Figuur 2. Schema van de uiteindelijke versie.

het calibration register van de INA219 worden geconfigureerd. Aan de hand van deze ijkwaarden berekent de sensor de stroom- en de vermogensgegevens. Hoe die parameters precies uitpakken wordt bepaald door wat we verlangen van deze schakeling, en door de shunt-weerstanden die we erin hebben toegepast. De berekening ziet u in **listing 1**. Voor dit

project wordt de waarde 5851 in het register van de beide INA219's geschreven.

De sensordata worden cyclisch door de microcontroller uitgelezen en verwerkt. Zodra een ingestelde grenswaarde wordt overschreden moet de microcontroller de stroomtoevoer onderbreken en die pas weer vrijgeven als de gebruiker het kanaal

Weblinks

- [1] Datasheet INA219: <http://www.ti.com/product/INA219>
- [2] Overzicht PIC18F6620: <http://www.microchip.com/wwwproducts/en/PIC18F6620>
- [3] MAX6004: <http://www.maximintegrated.com/en/datasheet/index.mvp/id/1916>
- [4] Projectpagina bij dit artikel (met download): <http://www.elektormagazine.nl/180335-B-03>

Listing 2. De task manager vormt het hart van de firmware.

```
/******  
*****  
*/ \fn      void task_manager(void)  
* \brief    update the tasks  
* \param    none  
* \exception none  
* \return   none  
*****  
*****/  
void task_manager(void){  
  
    static uint8_t prozesscounter = 0;  
  
    update_Status_Task();  
    //Switch case zum Einordnen der Task Updates  
    und zum Organisieren  
    switch(prozesscounter++){  
        case 1:  
        case 5:  
        case 7:  
        case 9:{  
            //Erzeuge alle 10 Hz Update der  
Messwerte  
            update_measurements_task();  
            break;  
        }  
        case 2:  
        case 4:  
        case 6:  
        case 8:{  
            //Limits überprüfen  
            check_limits();  
            break;  
        }  
        case 3: {  
            update_displaystring();  
            update_measurements_task();  
            break;  
        }  
        case 10: {  
            prozesscounter = 0;  
            check_limits();  
            break;  
        }  
    }  
    application_run();  
}
```

inschakelt. Bovendien moet de microcontroller voor het afschakelen een waarschuwing geven.

In de initialisatieroutine wordt de kalibratiewaarde in de beide INA219's weggeschreven, wordt de seriële poort ingesteld op 115200 baud 8N1 en wordt de interrupttimer voor de taskmanager op 10 Hz ingesteld. Vervolgens worden de laatstgebruikte instellingen uit de interne EEPROM ingelezen. Er zijn tien plaatsen in de EEPROM voorzien voor het opslaan van waarden.

De taskmanager (**listing 2**) start de taak elke 100 ms opnieuw (waarden ophalen en controleren, gebruikersinterface). De variabele `processcounter` wordt opgehoogd telkens als de routine een keer doorlopen wordt. Via een switch-statement worden de taken toegewezen en afgewerkt.

Gebruikersmenu

Na elke cyclus wordt de functie `application_run()` uitgevoerd. Achter deze functie zit het gebruikersmenu. Dit menu is uitgevoerd als een toestandsmachine, waarbij elk menu-item zijn eigen functie heeft. Om van het ene menu-item naar het volgende te gaan wijst een pointer naar de functie die bij de volgende cyclus moet worden uitgevoerd.

In het menu kunnen waarden voor het aangeven van een waarschuwing of van een fout met de stroom en de spanning worden ingesteld. Is er een waarschuwings- of een foutsituatie, dan wordt dit aangegeven met de status-LED's. Wordt een foutwaarde overschreden, dan wordt het desbetreffende kanaal afgeschakeld.

De weergave op het display is als volgt:

K 0000mV 0000mA on_off_X, in_ext_X

waarbij:

- k => kanaalnummer (1 of 2)
- 0000mV => de huidige spanning in mV
- 0000mA => de momentele stroom in mA
- on_off_X => kanaal is al dan niet ingeschakeld, er geldt een waarschuwing of er is een storing.
- in_ext_X => voedingsbron van het kanaal, intern of extern

Downloads

Alle software en de DesignSpark-files voor de hardware voor dit project kunnen (uiteeraard gratis) worden gedownload van de projectpagina bij dit artikel [4]. ◀

180335-B-03



IN DE STORE

→ USB Breakout Board voor het experimenteren met USB
www.elektor.nl/usb-breakout