

# S€M

## Slimme Energiemeter Monitor



Joop Tap (NL)

Tegenwoordig komen er steeds meer 'slimme meters' te hangen in woonhuizen. Met deze meters kan het gas- en elektriciteitsverbruik zowel thuis als op afstand worden uitgelezen.

S€M maakt dit ook voor de verbruiker toegankelijk. En wat is nu leuker dan in één oogopslag je eigen (momentele) energieverbruik te zien? Of nóg beter, het vermogen dat je terug levert aan het net wanneer je zonnepanelen hebt? Vooral als je ze net geplaatst hebt, kan dat zeer verslavend werken.

### Eigenschappen

- Leest dag-, maand- en jaarverbruik gas/elektra
- Leest huidig opgenomen vermogen
- Leest huidig teruggeleverd vermogen
- Logt data op SD-kaart
- Wifi of bedraad netwerk
- Stuurt data via internet naar elke gewenste ontvanger

Als je een zogenaamde 'slimme meter' [1] hebt, kun je daar veel informatie uit halen. Er zijn allerlei apps waarmee je het verbruik mee kunt bijhouden, maar ik wilde een groot led-display waarop real-time de data is te zien. Daarnaast wilde ik de data onbeperkt op kunnen slaan. Veel apps geven slechts de data van de vorige dag en slaan de meterstanden maar enkele maanden op. Weer andere laten je betalen voor deze service.

### Wat doet S€M?

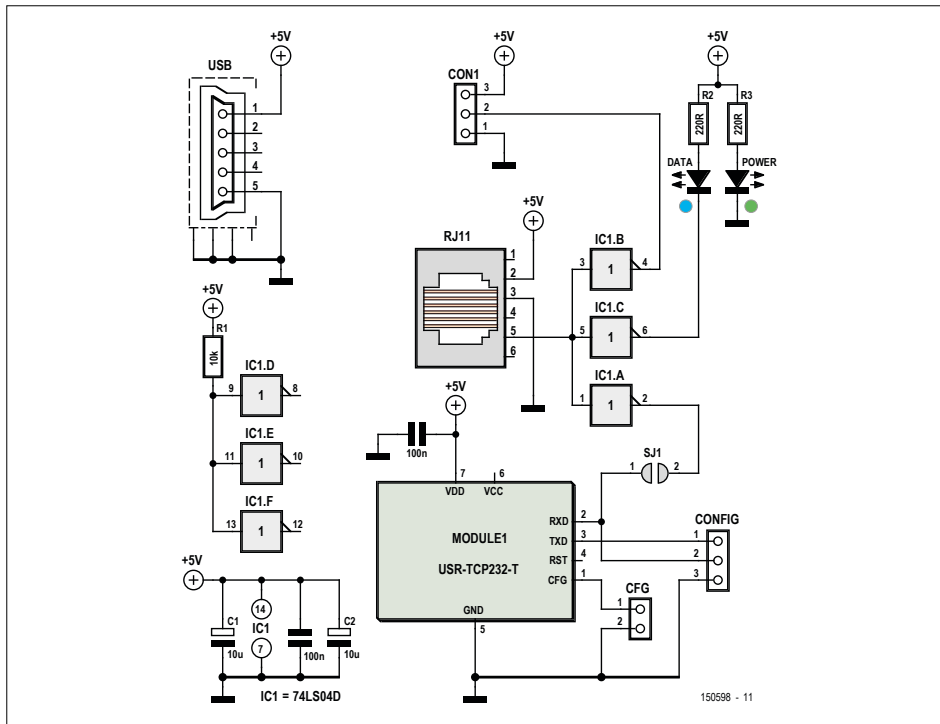
De data van de slimme meter (in mijn geval een Landys + Gyr G350 1-fase meter, volgens DSMR 4.0-protocol) wordt door een meter-interface via wifi in het hele huis beschikbaar gemaakt. Vervolgens kan deze data zowel op de S€M-Controller als op een groot led-display zichtbaar worden gemaakt. Met de Controller kunnen alle standen van de slimme meter worden uitgelezen en het dag-, maand- en jaarverbruik van zowel elektriciteit als gas worden bijgehouden. De data wordt iedere vijftien minuten op een SD-kaart weggeschreven in een komma-gescheiden bestand. Hiermee kunnen later bijvoorbeeld overzichtsgrafieken worden gemaakt.

### Hoe werkt het?

Op de zogenaamde 'P1-poort' van de slimme meter in de meterkast wordt een kleine interface aangesloten. Van

deze interface heb ik twee versies ontworpen: een versie voor 'bedrade' netwerken (**figuur 1**) en een versie die de data via wifi verstuurt (**figuur 2**), zodat deze draadloos in het hele huis beschikbaar is. Deze interface leest iedere 10 s alle data van de slimme meter volgens het DMSR 4.0-protocol met 115200 Baud uit en verstuurt deze als udp-pakket. Om als gangbaar serieel signaal aangemerkt te worden, wordt het uitgelezen signaal eerst geïnverteerd. Dit seriële signaal is ook als TTL-signaal beschikbaar via CON1 (TCP/IP-interface) respectievelijk DATA\_OUT (Wifi-interface).

De TCP/IP-interface bestaat uit niet meer dan een TTL-inverter-IC en een USR-TCP232-T-module die geconfigureerd moet worden. Dit kan volledig via het netwerk gedaan worden met de software die bij de module geleverd wordt. Een tweetal leds voor power en data geven de status van de interface aan.



Figuur 1. De TCP/IP-versie van de interface voor het uitlezen van de data uit de slimme meter is gebaseerd op een USB-TCP232T-module.

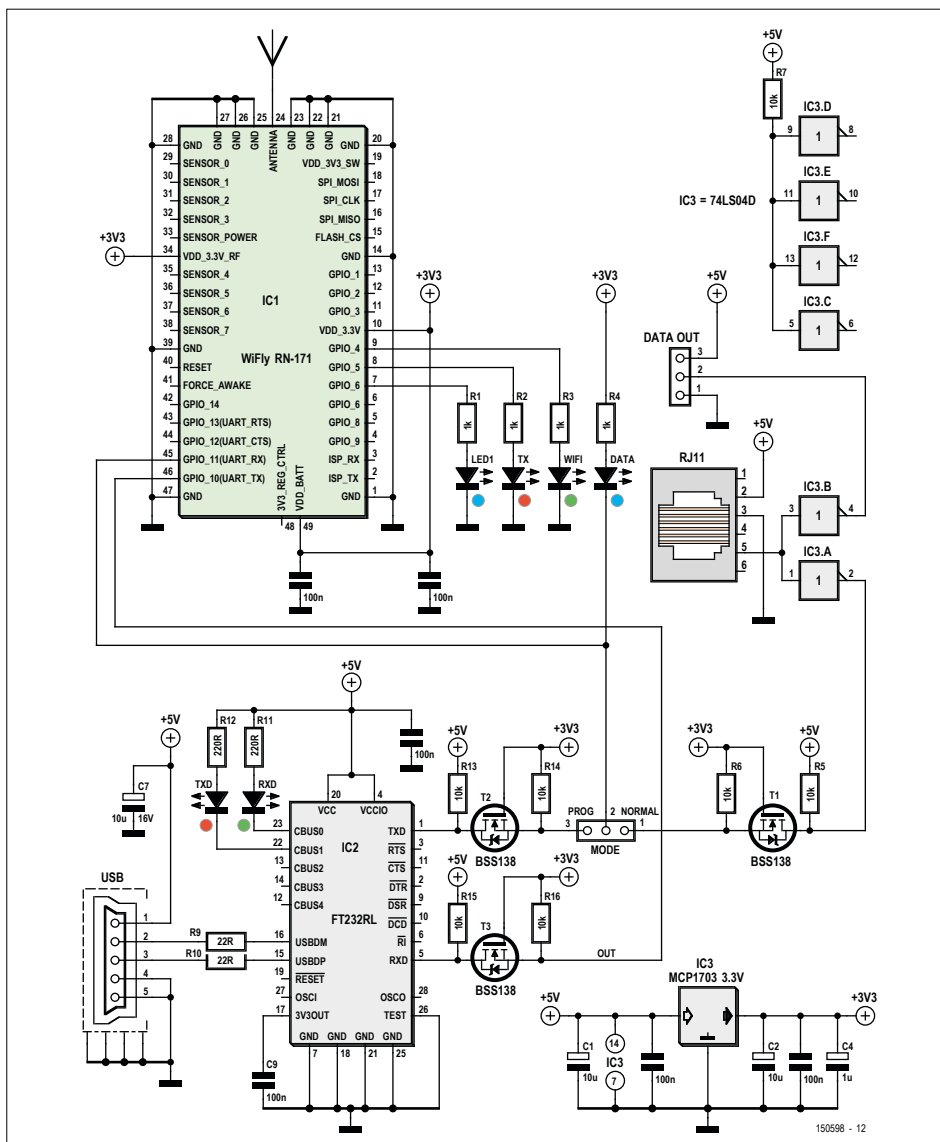
De wifi-interface heeft wat meer onderdelen aan boord, maar de kern is feitelijk de WiFly-module. Deze dient natuurlijk ook geconfigureerd te worden, waarvoor de interface via de mini-USB-connector op een pc aangesloten dient te worden. Hiertoe is IC2 geplaatst, een USB-RS232-converter-chip. Een handleiding voor de configuratie vindt u op [2]. Wanneer deze eerste configuratie is gedaan, kunnen toekomstige wijzigingen via het netwerk middels een 'Telnet-sessie' worden doorgevoerd.

Het datasignaal van de slimme meter wordt ook hier weer geïnverteerd door een 74LS04. Omdat de WiFly-module op 3,3 V werkt, zijn er level-converters aangebracht (T1...T3). Verder zijn er een aantal leds voor de toestandsaanduiding aangebracht: voor netwerkstatus (groen, rustig knipperend als er een netwerkverbinding is, snel knipperend als er geen verbinding is), voor data ontvangen van de slimme meter (blauw) en een rode led geeft aan dat er data via wifi wordt verstuurd. Daarnaast geeft een drietal SMD-leds de status aan bij het programmeren via USB.

Beide interfaces kunnen ingebouwd worden in een kleine kunststof behuizing van Hammond, type 1591MBK. Met een 1:1-kabel met aan beide zijden een RJ11-stecker kunnen de interfaces worden verbonden met de P1-poort van de slimme meter in de meterkast. Als voeding is een 5V-adapter met een mini usb-stecker geschikt. Op de wifi-interface zit een MCP1703 die de 5V-voedingsspanning omzet naar 3,3 V voor de WiFly-module. De data die door de TCP/IP- of wifi-interface wordt verstuurd, moet natuurlijk getoond worden. Dat kan met dit project op twee manieren: via de Controller of via het Led-display.

### Data ontvangen via de Controller

De S $\epsilon$ M-Controller analyseert de data die via wifi van de slimme meter komt. Op een 4-regelig LCD kunnen de diverse gegevens real-time worden afgelezen. De belang-

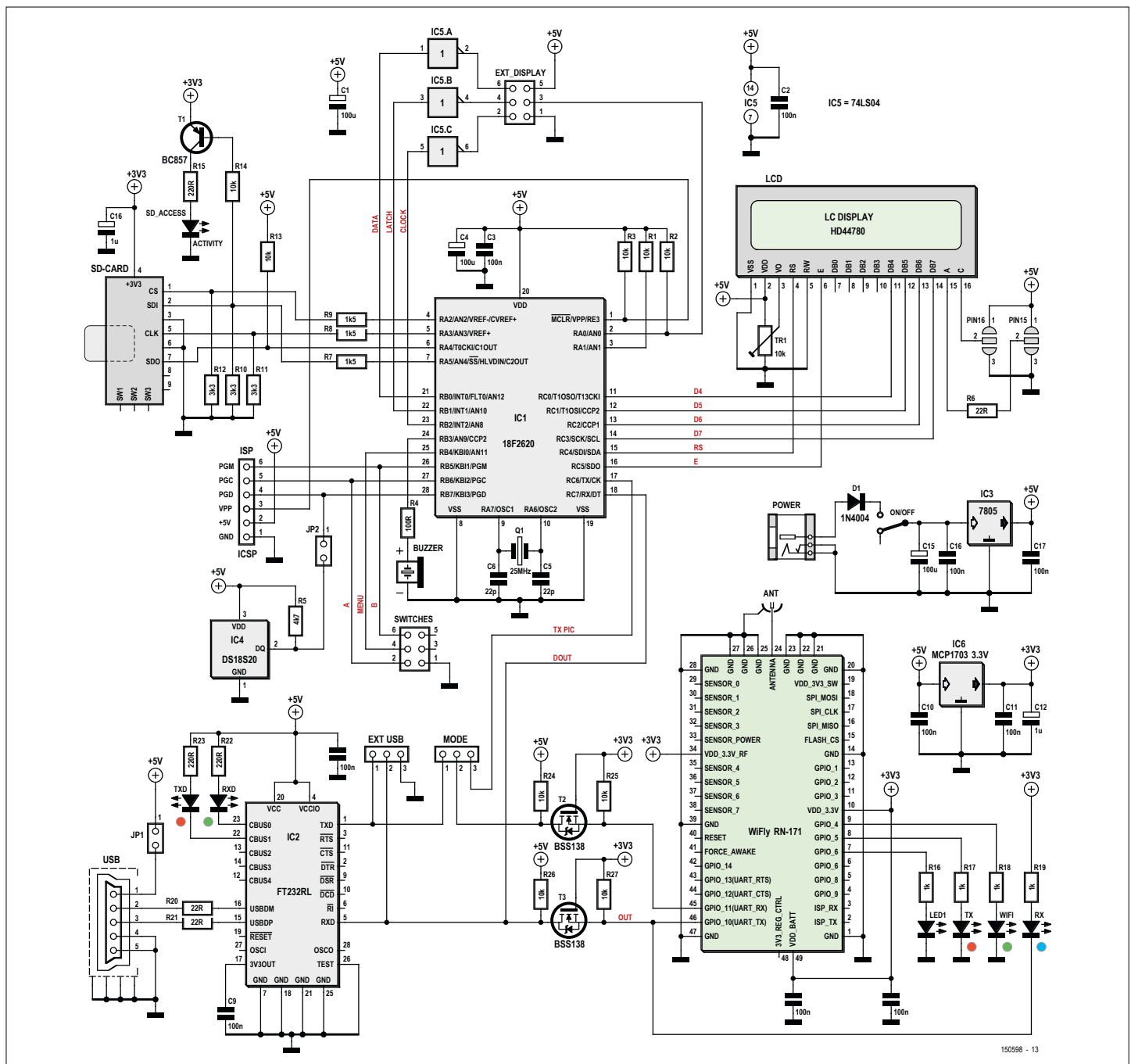


Figuur 2. In de wifiversie van de interface zit wat meer intelligentie om de WiFly RN-171-module aan te sturen.

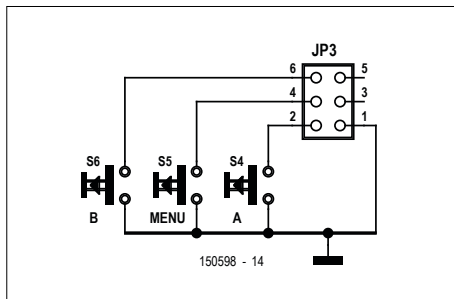
rijkste data wordt ook op een SD-kaart weggeschreven en via wifi doorgestuurd naar het Led-display. Zodra de Controller via wifi data van de slimme meter ontvangt, wordt deze ontcijferd en worden de verschillende meterstanden geanalyseerd. Totalen, saldo's en verhoudingen worden door de microcontroller, een PIC18F2620, berekend. Ieder kwartier worden de belangrijkste parameters op de SD-kaart weggeschreven naar een tekstbestand in een komma-gescheiden formaat. Bij stroomuitval blijft de dag-, maand- en jaar-data bewaard. De overige data wordt

weer opgehaald met het eerstvolgende bericht van de meter. De SD-kaart kan worden verwijderd om de data te bekijken en/of te bewerken op de computer. In het schema in **figuur 3** zien we een vrijwel identieke opbouw van het zend/ontvang-gedeelte rondom de WiFly-module als bij de wifi-interface. Een groene led geeft de wifi-status aan, een blauwe led geeft aan dat er data binnenkomt en een rode led licht op als de data via wifi wordt verzonden naar een Led-display. De data wordt direct naar de PIC gestuurd. Deze

verwerkt de data en handelt de aansturing van het LCD, de zoemer, en het weg-schrijven van de data op SD-kaart af. Als extraatje is een DS18S20 temperatuursensor toegevoegd, die met JP2 aan de PIC geknoopt wordt. Dat laatste is gedaan om uit te komen met het aantal IO van de PIC; IC4 deelt een pen met de ISP-connector. Net als de bedieningsknoppen overigens, die via de 'Switches'-connector aangesloten worden. Voor de bedieningsknoppen (**figuur 4**) is een apart printje ontworpen voor een comfortabele inbouw in een behuizing.



Figuur 3. De Controller verwerkt en bewaard de data en stuurt het Led-display aan via wifi.

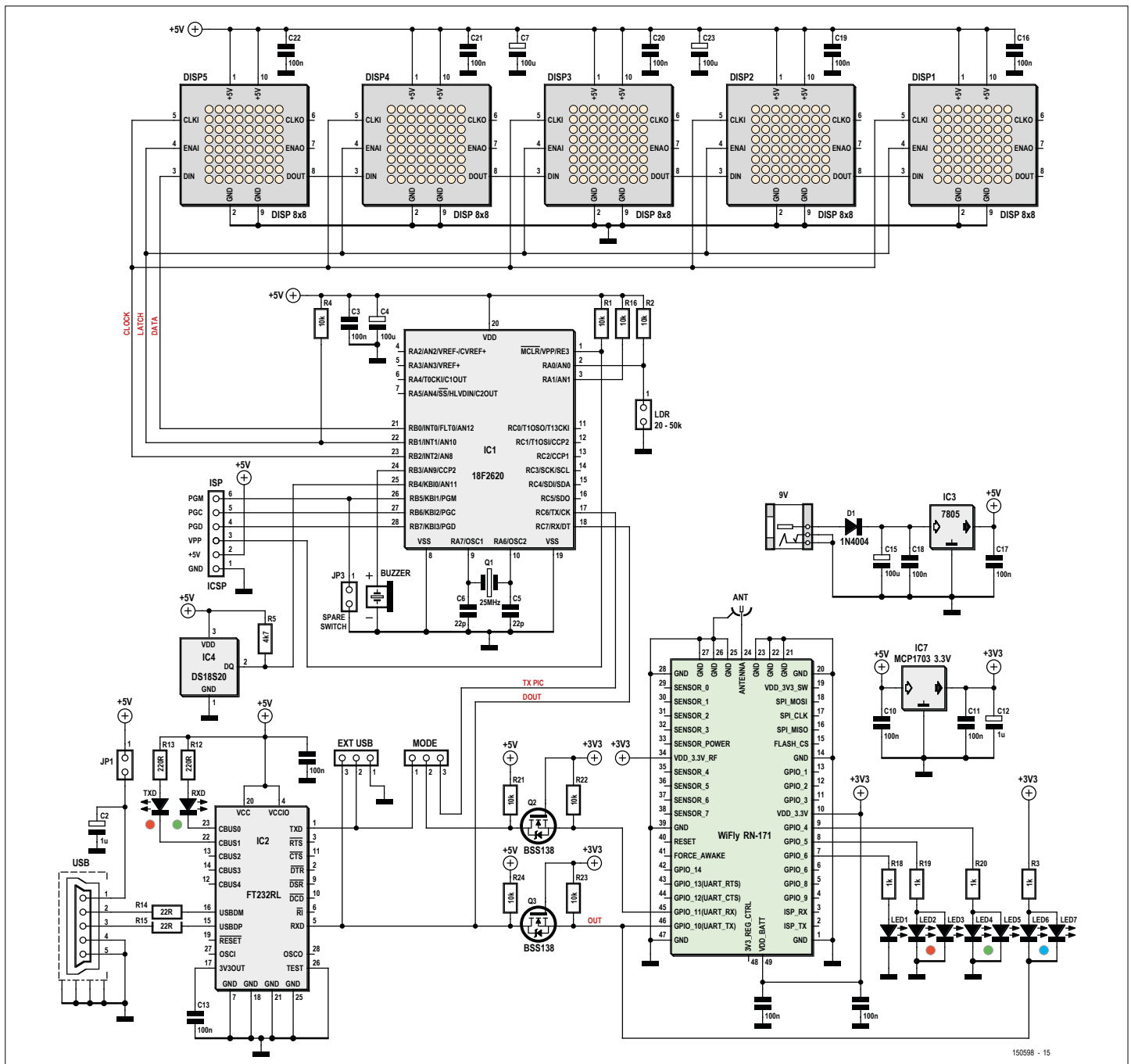


Figuur 4. De bedieningsknoppen van de Controller zitten op een apart printje dat voor de productie geïntegreerd is bij de TCP/IP- en wifi-interface-printen.

Bij SD-kaart-activiteit licht een rode led op (SD-access). Het is belangrijk dat de SD-kaart dan NIET wordt verwijderd, omdat de data anders corrupt kan raken! Het is verstandig om vóór het verwijderen van de SD-kaart het loggen te stoppen (menu#9, toets B). Na het terugplaatsen van de SD-kaart moet de kaart eerst weer gecontroleerd worden (menu#9, toets A). Hierna kan het loggen weer gestart worden.

De data voor het Led-display wordt weer via wifi verzonden door de Controller. In het 'settings'-menu wordt bepaald welke

data wordt getoond; dag-saldo elektriciteit en dag-saldo gas. Ook kan de intensiteit van het Led-display worden ingesteld; automatisch (past zich aan het omgevingslicht aan) of maximale helderheid. Het actueel opgewekte/afgenomen vermogen wordt altijd getoond. Als er op de Controller is gekozen om het piep-signaal bij het ontbreken van een wiferverbinding uit te schakelen, dan geldt dit ook voor het Led-display. Eventueel kunnen de 8x8 led-displays ook rechtstreeks op de PCB van de Controller worden aangesloten via EXT\_DISPLAY.



Figuur 5. Het Led-display maakt gebruik van dezelfde PIC18F2620-microcontroller als de Controller-module en kan dan ook standalone functioneren.

## Netwerkconfiguratie

Voordat de Controller gebruikt kan worden, moet het SSID en password van het thuisnetwerk worden ingegeven.

Ook moet er een IP-nummer aan de interface worden toegekend en het IP-nummer van het Led-display waar de data naartoe moet worden gezonden moet ingesteld worden. Een handleiding hiervoor vindt u op [2]. Als deze configuratie eenmaal is voltooid, wordt de mode-jumper op 'PIC' gezet en kunnen toekomstige wijzigingen aan de instellingen van de Controller via het netwerk worden gedaan met het programma Telnet. Als de wifiverbinding wegvalt, wordt dit op het LCD getoond en klinkt er (naar keuze) een zacht piepje.

Voor de firmware van de controller is gebruik gemaakt van PIC-BASIC en de Proton BASIC Compiler (versie 3.5.5.4) van Crownhill Associates. Er is firmware beschikbaar voor een 1-fase meter en een versie voor een 3-fasen meter volgens het DSMR 4.0-protocol [2]. De software is niet getest met andere meters!

De Controller past in een aluminium behuizing van Monacor, type AH-102/SW. Aan de ene zijkant wordt de SD-kaart geplaatst en zijn de status-leds, de 2,4GHz-wifi-antenne en de aan/uit schakelaar aangebracht. Aan de andere zijkant wordt een 7-12V-adapter aangesloten. Op het front bevindt zich een blauw, 4x20 LCD met achtergrondverlichting en een drietal druktoetsen.

De schakeling is vanzelfsprekend zeer energiezuinig (<1 Watt); we houden immers ons stroomverbruik in de gaten! Zo kan ook de achtergrondverlichting van het LCD automatisch doven als er enige tijd geen knoppen worden ingedrukt.

## Data op het Led-display

De Led-display-schakeling heeft veel weg van de Controller-schakeling. De onderste helft van het schema is vrijwel gelijk. De PIC stuurt nu echter vijf displaymodules aan van het type MAX7219 met 8 x 8 leds. Hierop wordt de ontvangen data getoond.

Ook hier weer geeft een groene led de wifistatus aan, een blauwe led licht op als er data van de Controller wordt ontvan-



gen en een rode led licht op wanneer de data wordt doorgezonden naar een eventueel volgend Led-display. Met een LDR wordt het omgevingslicht gemeten en kan de intensiteit van de leds automatisch worden geregeld. Als de wifiverbinding wegvalt, verschijnt na enige tijd de melding 'wifi?' en klinkt er een zacht piepje.

Via de mini-USB-aansluiting wordt de WiFly geconfigureerd voor het gebruik in het netwerk (mode-jumper op 'PROG'). Zo moet o.a. het SSID van het Wlan worden ingesteld en het bijbehorende wachtwoord. Een handleiding hiervoor vindt u op [2]. Als deze configuratie eenmaal is voltooid, wordt de jumper op 'NORM' gezet en kunnen toekomstige configuratiewijzigingen gedaan worden via Telnet. Op de PCB is een jumper aangebracht waarmee de schakeling in test-mode gaat en alle leds gaan branden.

De PCB is zo ontworpen dat de led-matrix-displays op de PCB worden gestoken (sandwichconstructie). Vervang hiertoe de haakse jumperstrips door rechte uitvoeringen! Het Led-display past in een behuizing van 17 x 6 x 4,5 cm. Ik heb deze gemaakt van enkelzijdige printplaat. Er is slechts een aansluiting voor een 7-12V adapter en de wifi-antenne. Ook het Led-display is uiterst energiezuinig met een opgenomen vermogen <1,5 Watt.

## Andere configuraties

Het is mogelijk om alleen het Led-display te gebruiken in combinatie met de TCP/IP- of wifi-interface, maar dan kan er uiteraard niet op een SD-kaart gelogd worden en is alleen het

gas en elektra-dagverbruik en het opgewekte/afgenomen vermogen af te lezen. Het Led-display dient daarvoor wel voorzien te worden van speciale stand-alone software. De data van de slimmemeter, die via wifi binnenkomt, wordt dan door het Led-display zelf gedecodeerd en geanalyseerd en niet door de Controller. Het display toont dan afwisselend het momentele teruggeleverde of afgenomen vermogen, het dag-saldo elektra en naar keuze het dag-gasverbruik. JP3 dient dan voor de selectie toon wel/geen gasverbruik.

## Zelf construeren

Een aantal handige tips bij de opbouw zijn wellicht verstandig om in acht te nemen. Zo kun je het beste controleren of de PCB van de Controller in de behuizing past vóórdát de soldeerbout opgewarmd wordt. Vervolgens is het raadzaam eerst de USB-connector te monteren, want die is het moeilijkst. Bouw elke module in 'blokjes' op en test telkens de werking hiervan, zodat je niet de hele schakeling hoeft te doorzoeken mocht er iets niet functioneren.

Wanneer alles is opgebouwd en elektronisch werkt, kan er met het programmeren en instellen begonnen worden.

De benodigde firmware en handleidingen hebben we voor u verzameld op [2]. De auteur staat u graag via zijn website [3] te woord bij vragen (Nederlands, Engels). ◀

(150598)

## Weblinks

[1] Slimme meter: [nl.wikipedia.org/wiki/Slimme\\_meter](http://nl.wikipedia.org/wiki/Slimme_meter)

[2] Website bij dit artikel: [www.elektor.nl/150598](http://www.elektor.nl/150598)

[3] Contact met de auteur: <http://slimmemeter.jimdo.com/contact>

Meer informatie over de P1-poort:

<http://domoticx.com/p1-poort-slimme-meter-hardware>

Meer informatie en foto's/video's: <http://slimmemeter.jimdo.com>