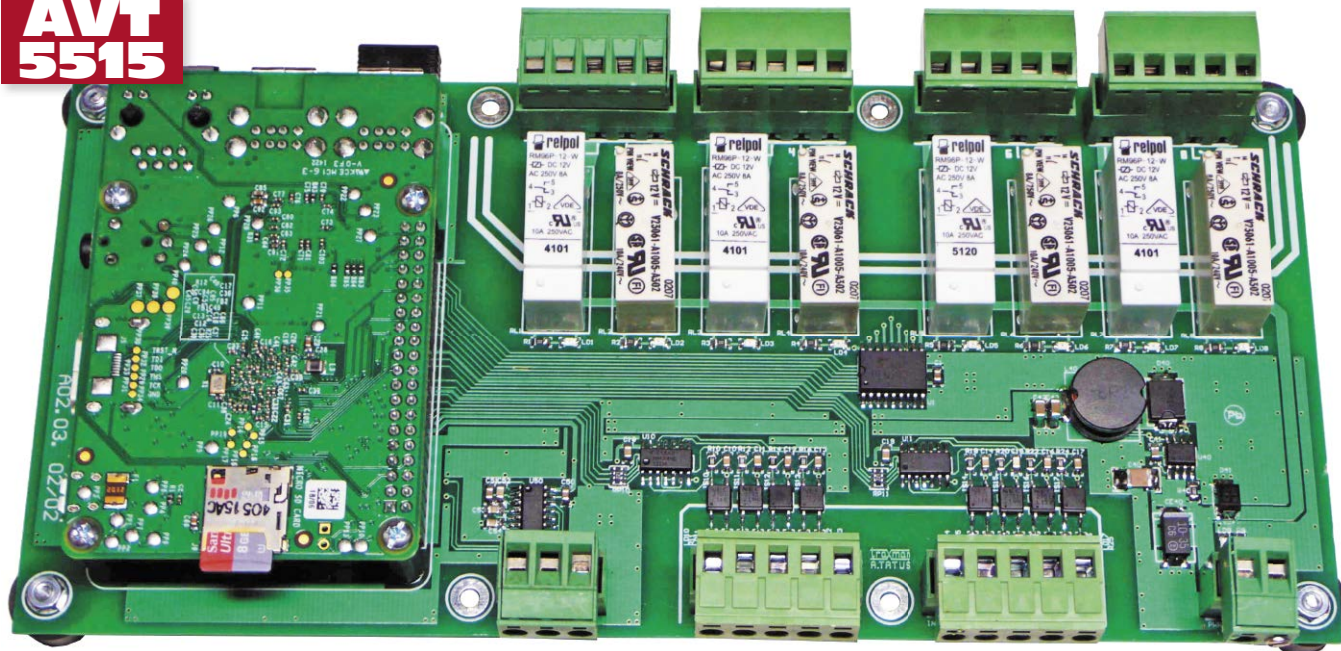


**AVT
5515**


HABoard – moduł automatyki domowej dla RPi+

Moduł „automatyki domowej” dla Raspberry Pi+. Opisujemy projekt ułatwia zastosowanie Raspberry Pi+ w systemach automatyki domowej. Zawiera zasilacz oraz drivery wejść/wyjść, które konwertują poziom napięcia wejściowego na bezpieczny dla naszego mikrokomputera oraz umożliwiają mu sterowanie dołączonymi urządzeniami.

Rekomendacje: płytkę przyda się wszystkim, którzy planują zastosowanie Raspberry Pi do sterowania dołączonymi urządzeniami.

Schemat ideowy płytki podzielono na schematy poszczególnych bloków funkcjonalnych, które zostaną omówione oddzielnie.

Ze względu na zasilanie modułu napięciem +12 V DC z typowego zasilacza (w tym buforowego w razie aplikacji wymagającej podtrzymania zasilania) konieczne jest zapewnienie +5 V dla zasilania Raspberry Pi. W tym celu zastosowano przetwornicę obniżającą napięcie opartą o układ scalony ADP2302-50 (U40), której schemat ideowy zamieszczono na **rysunku 1**. Wejście przetwornicy jest zabezpieczone przed przepięciami diodą D41. Dioda świecąca LD9 sygnalizuje doprowadzenie zasilania. Ze względu na wysoką częstotliwość kluczkowania, istotny jest dobór elementów przetwornicy. Filtr wejściowy wykonano z użyciem kondensatorów o obniżonym ESR: CE40 (tantalowy) i C40 w obudowie 1812 z dielektrykiem X5R. Podobne wymagania dotyczą filtra wyjściowego (kondensatory C42, C43), gdzie w miejsce typowego kondensatora elektrolitycznego, zastosowano nowoczesne kondensatory

ceramiczne 22 µF/16 V z dielektrykiem X5R. Przetwornica jest w stanie zapewnić prąd 1,5...2 A wystarczający do zasilania Raspberry i typowych kart Wi-Fi.

Do monitorowania sygnałów cyfrowych zastosowano typowy układ separacji opartej o transoptor, filtr RC i bramkę Schmidta (**rysunek 2**). Układ sprzętowo eliminuje krótkie zakłócenia impulsowe ze styków mechanicznych. W szereg z diodą transoptora włączono diodę LED sygnalizującą obecność napięcia. Do polaryzacji wejść można wykorzystać +12 V zasilające moduł lub napięcie zewnętrzne w wypadku wymaganej pełnej separacji. Wejścia przeznaczone są do sterowania w standardzie NPN. Sygnały wejściowe podzielone są na dwie grupy po cztery wejścia, umożliwia to sterowanie z dwóch osobnych potencjałów z zachowaniem izolacji.

Wyjścia cyfrowe (**rysunek 3**) zrealizowano w oparciu o przekaźniki. Cztery grupy składające się z przekaźnika zwiernego i przełącznego zapewniają elastyczność połączeń. Cewki przekaźników sterowane

W ofercie AVT*
AVT-5515 A

Podstawowe informacje:

- Dwie grupy, po cztery wejścia cyfrowe z optoizolacją.
- Cztery wyjścia przekaźnikowe ze stykiem zwiernym 230 V/8 A.
- Cztery wyjścia przekaźnikowe ze stykiem przełącznym 230 V/8 A.
- Interfejs szeregowy RS232.
- Zegar czasu rzeczywistego z podtrzymaniem baterijnym DS1338.
- Wbudowany zasilacz impulsowy 5 V/1,5 A dla Raspberry Pi.
- Zasilanie +12 V, możliwa praca z zasilaczem buforowym systemu antywłamaniowego.

Dodatkowe materiały na FTP:

<ftp://ep.com.pl>, user: 82218, pass: aagt5gjb

• wzory płytek PCB

• Projekty pokrewne na FTP:

(wymienione artykuły są w całości dostępne na FTP)

- AVT-5513 Moduły do komunikacji szeregowej Xbee dla Raspberry Pi i nie tylko (EP 9/2015)
- AVT-1854 RaspbPI_PLUS_GPIO. Moduł rozszerzeń GPIO Pi B + (EP 6/2015)
- AVT-1851 RaspbPI_DAC – przetwornik audio dla Raspberry Pi (EP 4/2015)
- AVT-1827 RaspbPI_NFC – płytkę czytnika RFID dla Raspberry Pi i nie tylko (EP 9/2014)
- AVT-5459 RaspbPI_GSM Płytkę z modemem GSM dla Raspberry Pi (EP 7/2014)
- AVT-5431,-32,-33 Moduły rozszerzeń dla Raspberry Pi (4) – RaspbPI_LCD, RaspbPI_Relay, RaspbPI_LED8_PWM_Expander (EP 1/2014)

* Uwaga:
Zestawy AVT mogą występować w następujących wersjach:
AVT xxxx UK to zaprogramowany układ. Tylko i wyłącznie. Bez elementów dodatkowych.
AVT xxxx A płytka drukowana PCB (lub płytki drukowane, jeśli w opisie wyraźnie zaznaczono), bez elementów dodatkowych.
AVT xxxx A+ płytka drukowana i zaprogramowany układ (czyli połączenie wersji A i wersji UK) bez elementów dodatkowych.
AVT xxxx B płytka drukowana (lub płytki) oraz komplet elementów wymienionych w załączniku pdf.
AVT xxxx C to nie innego jak zmontowany zestaw B, czyli elementy wmontowane w PCB. Należy mieć na uwadze, że o ile nie zaznaczono wyraźnie w opisie, zestaw ten nie ma obudowy ani elementów dodatkowych, które nie zostały wymienione w załączniku pdf.
AVT xxxx CD oprogramowanie (nieczęsto spotykana wersja, lecz jeśli występuje, to niezbędne oprogramowanie można pobrać, klikając w link umieszczony w opisie kitu)

Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf! Podczas składania zamówienia upewnij się, którą wersję zamawiasz! (UK, A, A+, B lub C). <http://sklep.avt.pl>

typu ZL307-2x20 o wysokości 7 mm, montowane jedno na drugim. Przy montażu w rozdzielnicy elektrycznej na szynie TS35 płytkę HAB można zamocować przy wykorzystaniu uchwytów Weidmuller typu RS100. Budowa mechaniczna zapewnia dostęp do wszystkich złączy Raspberry Pi. Wszystkie zastosowane złącza sygnałowe są rozłączane.

Uruchomienie

Moduł nie wymaga uruchamiania. Należy jedynie skonfigurować system do obsługi poszczególnych peryferiów. W celu wykorzystania RTC konieczne jest dodanie obsługi magistrali I²C. W tym celu sprawdzamy czy w pliku `sudo nano /etc/modules` znajduje się definicja `i2c-dev`. Jeżeli nie, to musimy ją dodać, zapisać zmiany i zrestartować PI. Po uruchomieniu należy pobrać narzędzia odpowiadające za obsługę I²C:

```
sudo apt-get install python-smbus
sudo apt-get install i2c-tools
```

Po zainstalowaniu, w pierwszej kolejności sprawdzamy w konsoli prawidłowe działanie interfejsu I²C za pomocą polecenia

`sudo i2cdetect -y 1`. Powinno pojawić się urządzenie pod adresem 0x68 – jest to nasz RTC (DS1338). Następnie ładujemy moduł zegara:

```
sudo modprobe rtc-ds1307
sudo bash echo ds1307 0x68 > /
sys/class/i2c-adapter/i2c-1/
new_device
```

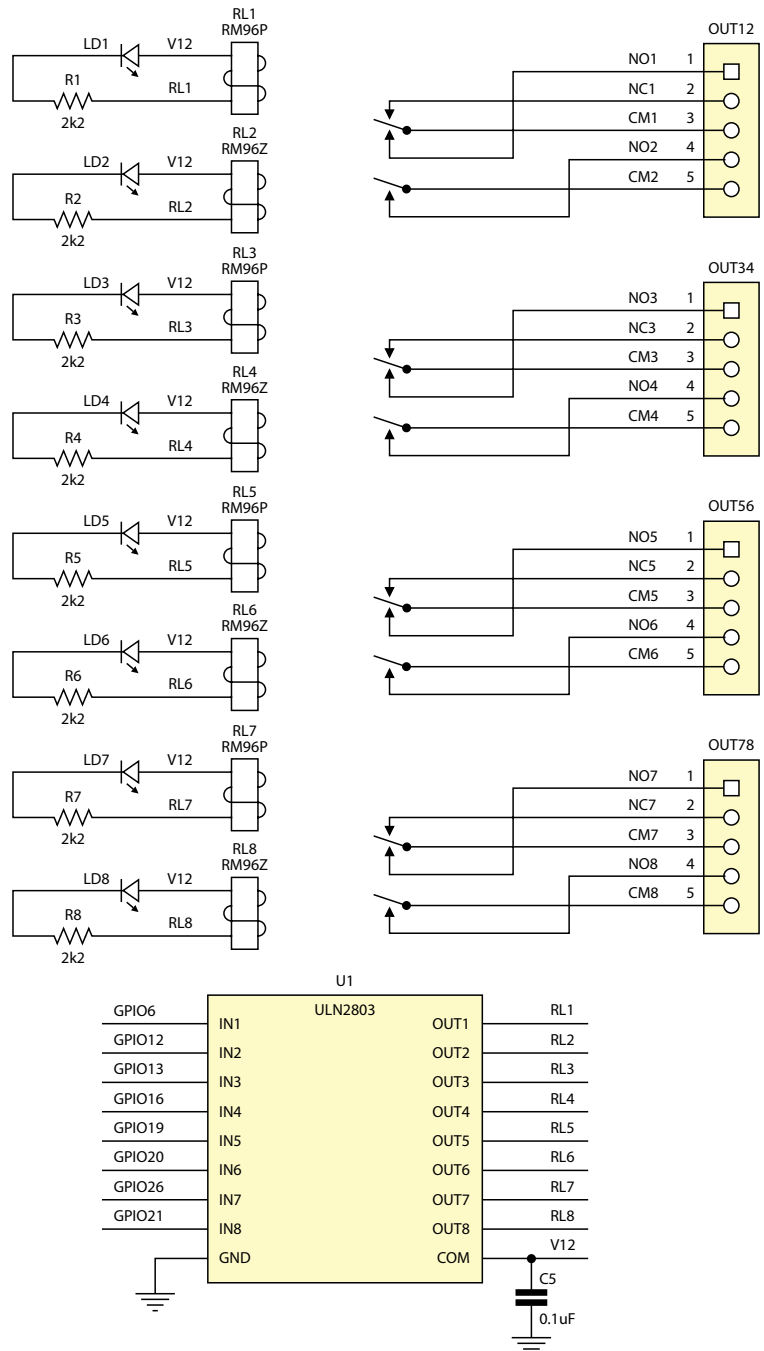
Wykaz elementów

Rezystory: (SMD 0805)
 R1...R9, R11, R13, R15, R17, R19, R21, R23, R25: 2,2 kΩ
 R10, R12, R14, R16, R18, R20, R22, R24: 10 kΩ
 R40: 100 kΩ
 RP10, RP11: 220 Ω (CRA06S08 rezystor PACK 4x)

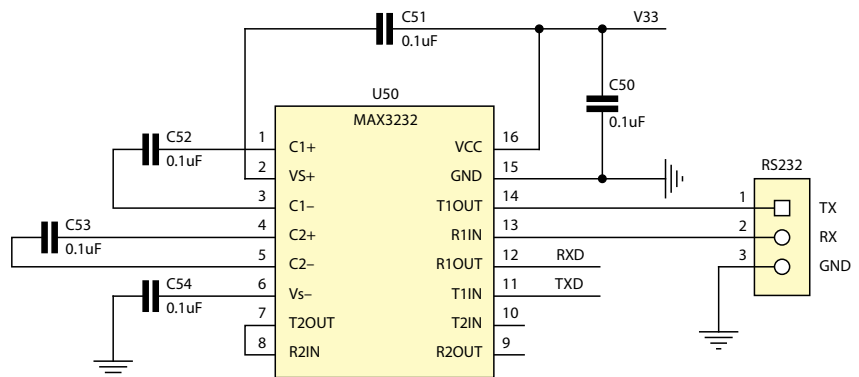
Kondensatory:
 C1, C18, C19, C30, C41, C50...C54: 100 nF (SMD 0805, X5R, 25 V)
 C10...C17: 1 nF (SMD 0805, X5R, 25 V)
 C40: 0,1 μF (SMD 1812, X5R, 50 V)
 C42, C43: 22 μF (SMD 1206, X5R, 16 V)
 CE40: 10 μF/35 V (SMD „C”)

Półprzewodniki:
 D40: SK33 (diody Schottky)
 D41: P6SMB16A (transil 16 V)
 LD1...LD17: dioda LED, SMD, zielona
 IS10...IS17: LTV357
 U1: ULN2803 (SO18W)
 U10, U11: HC132 (SO14)
 U30: DS1338Z (SO8)
 U40: ADP2302 (SO8TP)
 U50: MAX3232 (SO16)

Inne:
 BAT: bateria litowa CR1220 z podstawką do druku KEYS3000
 GPIO: złącze kielichowe 7 mm, 2x20 pin – 2 szt.
 IN14, IN58, OUT12, OUT34, OUT56, OUT78: złącze MC5 rozłączane, proste, kompletne, 5x5 mm
 L40: 6,8 μH (dławik SMD, SDR1307, 6,8 Ω)
 PWR: złącze MC5 rozłączane, proste, kompletne, 2x5 mm
 RL1, RL3, RL5, RL7: RM96P (przełącznik, styki przełączne, 12 V DC)
 RL2, RL4, RL6, RL8: RM96Z (przełącznik, styki zwierne, 12 V DC)
 RS232: złącze MC5, rozłączane, proste, kompletne, 3x5 mm
 XT: kwarc 32768 kHz (SMD)

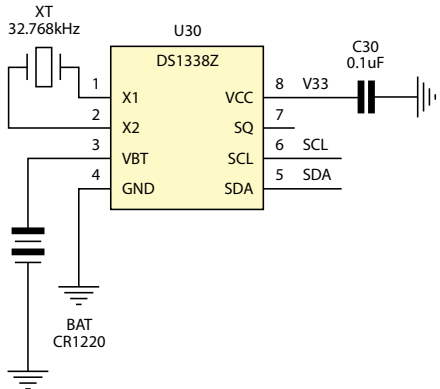


Rysunek 3. Schemat wyjść cyfrowych



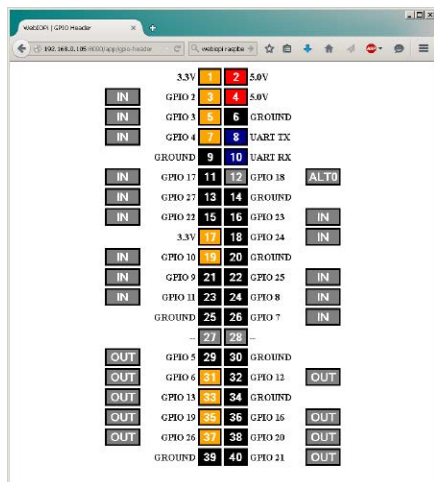
Rysunek 4. Schemat interfejsu RS232

Ustawienia czasu i daty systemowej dokonujemy poleceniem `sudo date`. Zapis czasu systemowego do RTC wykonujemy poleceniem `sudo hwclock -w`. Sprawdzenie poprawności zapisu `sudo hwclock -r`. Aby czas systemowy



Rysunek 5. Schemat zegara RTC

GPIO			
V33	1	2	V50
SDA	3	4	V50
SCL	5	6	GND
GPIO4	7	8	TXD
GND	9	10	RXD
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
V33	17	18	GPIO24
GPIO10	19	20	GND
GPIO09	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
	27	28	
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21



Rysunek 6. Przyporządkowanie GPIO

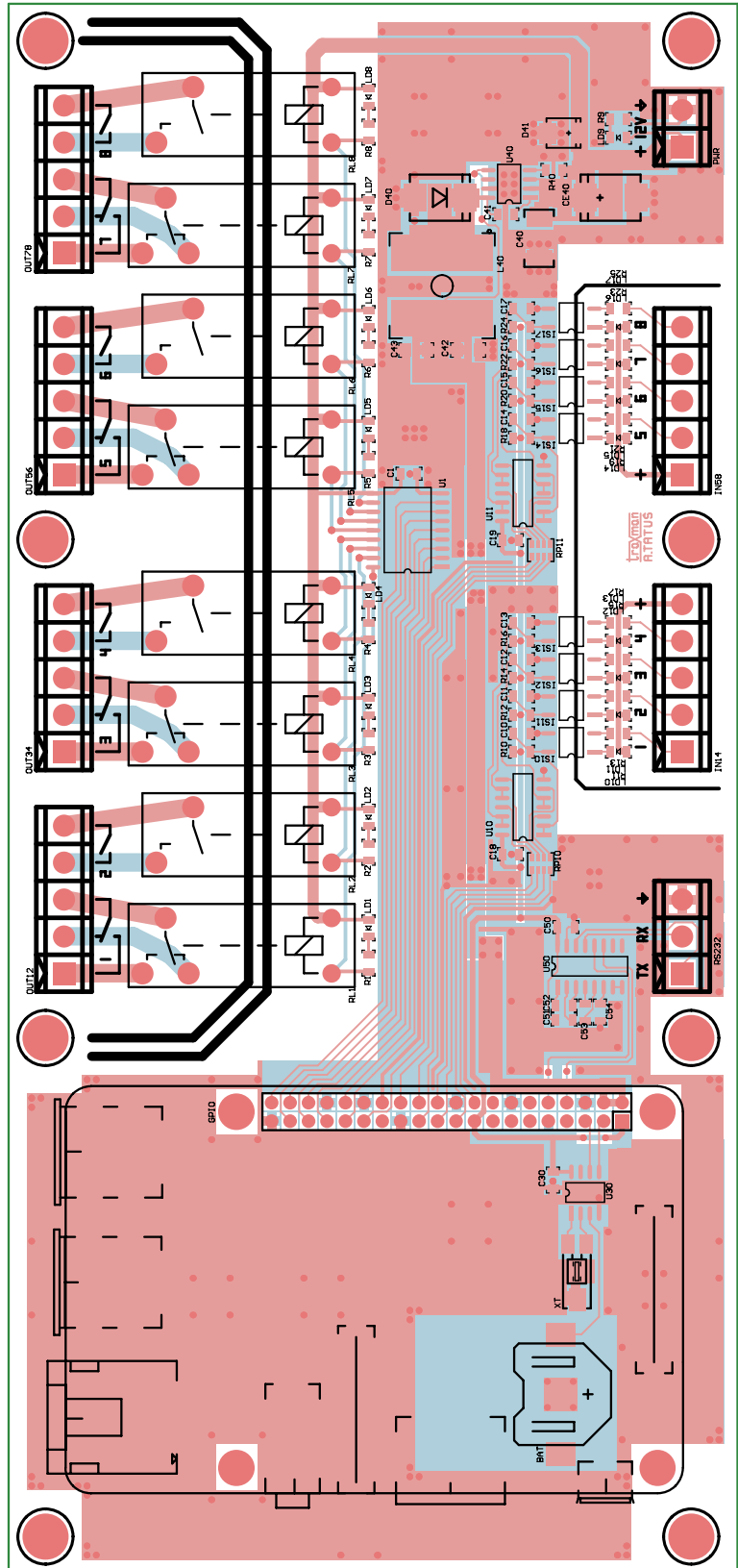
po uruchomieniu Pi był aktualizowany z RTC automatycznie musimy w pliku `sudo nano /etc/modules` dodać linię `rtc-ds1307`, a w pliku `sudo nano /etc/rc.local` dodać linie:

```
echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-1/new_device
sudo hwclock -s
```

Należy je umieścić przed poleceniem `exit 0`. Przy kolejnym uruchomieniu PI, czas

zostanie pobrany z RTC bez synchronizacji z zegarem sieciowym.

W korzystaniu z komunikacji RS232 musimy uzyskać dostęp do portu szeregowego, który jest domyślnie zablokowany przez terminal SSH. W tym celu należy edytować plik `cmdline.txt` `$ sudo nano /boot/cmdline.txt` i usunąć wpisy dotyczące konsoli `console=ttyAMA0,115200`



Rysunek 7. Schemat montażowy płytki automatyki dla Raspberry Pi

Wydanie specjalne

„Raspberry Pi”

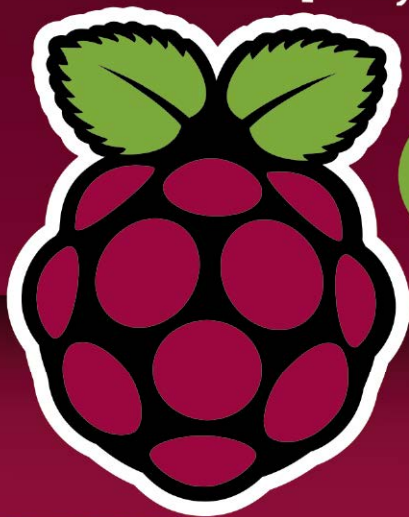
to polski przekład światowego bestsellera na temat słynnego minikomputera

WYDANIE SPECJALNE „MŁODEGO TECHNIKA” NR 1/2015

Raspberry Pi

Ależ to bardzo proste!

Jak w pełni wykorzystać możliwości minikomputera Raspberry Pi



196 pomysłów i porad

KOMPENDIUM DLA NIEELEKTRONIKÓW

- ROZPOCZĘCIE PRACY
- PODSTAWOWE UMIEJĘTNOŚCI
- PROGRAMOWANIE
- PROJEKTY

To kompendium wiedzy o konfiguracji

i sposobach programowania tego uniwersalnego urządzenia oraz prawie dwieście pomysłów i sztuczek aplikacyjnych

Nie będziesz rozczarowany!

Nie musisz być elektronikiem, aby zaprzęgnąć Raspberry Pi do wykonywania niezliczonych rodzajów funkcji i aplikacji

Z tym poradnikiem możesz to osiągnąć!

Szukaj na

www.UlubionyKiosk.pl

(przesyłka GRATIS)

Raspberry Pi B+ J8 Header

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1, I2C)	DC Power 5v	04
05	GPIO03 (SCL1, I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO18	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

Rev. 1.1
18/07/2014

<http://www.element14.com>

Rysunek 8. Zaktualizowana lista GPIO

`kgdboc=ttyAMA0,115200` oraz edytować plik `initab` \$ `sudo nano /etc/initab` komentując # w nim linię `T0:23respawn:/sbin/getty -L ttyAMA0 115200 vt100`. Po wprowadzeniu zmian, należy zrestartować Raspberry.

Dla sprawdzenia działania modułu można wykorzystać WebIOPI. W tym celu należy pobrać oprogramowanie spod adresu <http://goo.gl/CTVEnh> i zainstalować je:

```
$ tar xvzf WebIOPi-x.y.z.tar.gz
$ cd WebIOPi-x.y.z
$ sudo ./setup.sh
```

Aktualna wersja 0.7.1 nie ma jeszcze obsługi rozszerzonego GPIO (plus), należy więc wykonać pewne modyfikacje plików źródłowych. W pliku `./python/utills/version.py` należy edytować linię 11 i dopisać do niej wartości wyróżnione pogrubieniem:

```
_MAPPING[2] = [„V33”, „V50”, 2, „V50”, 3, „GND”, 4,
14, „GND”, 15, 17, 18, 27, „GND”, 22, 23, „V33”, 24,
10, „GND”, 9, 25, 11, 8, „GND”, 7, „DNC”, „DNC”, 5,
„GND”, 6, 12, 13, „GND”, 19, 16, 26, 20, „GND”, 21]
```

Należy również edytować plik `./htdocs/webiopi.js`:

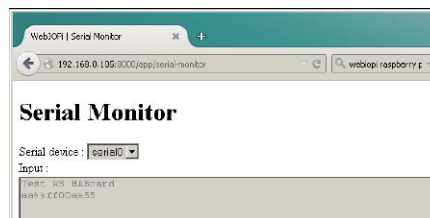
Odszukać linię 76: `this.PINS = Array(27)`; zamienić ją na `this.PINS = Array(41)`.

Linię 560 `for (var pin=1; pin<=26; pin++)` zamienić na `for (var pin=1; pin<=40; pin++)`.

Plik należy skompilować zmianą poleceniem `sudo ./setup.sh skip-apt`. Po kompilacji należy skonfigurować Webiopi do obsługi portu szeregowego. W tym celu w pliku `/etc/webiopi/config` w sekcji DEVICE należy dodać wpis `serial = Serial device:ttyAMA0 baudrate:9600` lub usunąć komentarz # jeżeli wpis istnieje, zapisać zmiany w pliku i uruchomić serwer Webiopi `sudo /etc/init.d/webiopi start`. Po przejściu do przeglądarki internetowej pod adres <http://localhost:8000/> (użytkownik: **webiopi**, hasło: **raspberrypi**) w linku GPIO-header powinna być widoczna zaktualizowana lista GPIO (**rysunek 8**) oraz po wybraniu linku Serial Monitor, możliwe jest sprawdzenie działania portu szeregowego **rysunek 9**.

Elastyczność Webiopi umożliwia wykorzystanie go w prostych aplikacjach domowej automatyki poprzez modyfikację skryptów w dołączonych przykładach, sposoby wykorzystania opisane są na stronie Wiki projektu oraz na forum.

Adam Tatuś, EP



Rysunek 9. Aktywny port szeregowy

REKLAMA