



multiTID

– wielofunkcyjny, samochodowy komputer pokładowy (2)

Po lekturze pierwszej części artykułu wiemy już wszystko w zakresie obsługi wyświetlacza pokładowego oraz emulowania jego funkcjonalności, więc pora na zaprezentowanie szczegółów budowy komputera samochodowego multiTID.

Schemat ideowy komputera samochodowego multiTID pokazano na **rysunku 7**. Został on zbudowany z zastosowaniem mikrokontrolera Atmega644-PA, zegara czasu rzeczywistego z podtrzymaniem zasilania MCP7940N-I/SN oraz wyświetlacza TFT pełniącego rolę graficznego interfejsu użytkownika.

Mikrokontroler, jak to zwykle bywa, stanowi „serce”. Jest taktowany za pomocą rezonatora kwarcowego 12,288 MHz dla zapewnienia dużej dokładności pomiaru czasu oraz prędkości ładowania obrazków stanowiących elementy graficznego interfejsu użytkownika. Oczywiście, lepiej byłoby zastosować maksymalną, dostępną częstotliwość taktowania mikrokontrolera tj. 20 MHz, jednak przy napięciu zasilania 3,3 V jest to wartość progowa, dla której deklarowana jest poprawna praca mikrokontrolera.

W celu realizacji założonej funkcjonalności urządzenia, w programie obsługi niniejszego sterownika, wykorzystano dwa sprzętowe układy czasowo-licznikowe znajdujące się „na pokładzie” mikrokontrolera oraz jedno przerwanie zewnętrzne. Sposób konfiguracji wspomnianych peryferiów oraz realizowaną przez nie funkcjonalność pokazano w **tabeli 6**. W aplikacji użyto także ostatniego dostępnego układu

czasowo-licznikowego Timer2. Skonfigurowano go do pracy w trybie CTC, w którym generuje cykliczne przerwania (co 10 ms) wywołując obsługę klawiatury (przyciski MENU/<, OK/>). Dzięki zastosowaniu tego typu rozwiązania, program obsługi aplikacji urządzenia nie używa opóźnień, co zapewnia jego bezproblemową pracę oraz możliwość detekcji czasu naciśnięcia przycisku (krótki/długi/przytrzymanie itd.), dzięki czemu udało się poprawić ergonomię.

Jak opisano w tab. 6, multiTID dokonuje w czasie każdej sekundy pomiaru sumarycznej liczby impulsów doprowadzanych na wejście T0 mikrokontrolera z przetwornika drogi pojazdu oraz pomiaru sumarycznego czasu wtrysków, których to sygnał jest doprowadzony na wejście ICP1 licznika Timer1. W celu realizacji drugiej z funkcjonalności zaprojektowano dwa kompletne, bezpieczne układy wejściowe formujące sygnały wtryskiwaczy paliwa (oddzielnie dla Pb i LPG) dla potrzeb wejściowych obwodów mikrokontrolera. Wybór wtryskiwacza paliwa, z którego impulsy podawane są na wejście ICP mikrokontrolera, jest realizowany poprzez scalony przełącznik NC7SB3157 (U5), dla którego sygnałem sterującym jest poziom logiczny na wyjściu PD7 mikrokontrolera. Jest on zmieniany w procedurze obsługi przerwania PCINT1, które

jest wywoływane zmianą poziomu na wprowadzeniu PB6 na skutek załączenia zaworu na reduktorze instalacji LPG (czyli zmiany paliwa zasilającego silnik na LPG, w tym „start” wtryskiwaczy gazu). To rozwiązanie pozwala na efektywną realizację funkcji automatycznej detekcji rodzaju paliwa i reakcję po stronie programu obsługi.

Na **listingu 8** pokazano wzory zaczerpnięte bezpośrednio z programu obsługi aplikacji służące do obliczenia wszystkich parametrów rzeczywistych komputera pokładowego.

Kilka słów komentarza wymaga mechanizm akumulowania przejechanej drogi i zużytego paliwa. Komputer multiTID oblicza i wyświetla średnie wartości zużycia paliwa i prędkości, objętość paliwa pozostającego w baku pojazdu oraz pokonany dystans. Jak łatwo się domyślić, w celu wyznaczenia wspomnianych wartości jest niezbędna znajomość całkowitego zużycia paliwa i dystansu od momentu wyzerowania liczników, a co za tym idzie – niezbędny staje się mechanizm akumulowania mierzonych wartości, niezależnie dla każdego rodzaju paliwa.

Ustawienia Fuse-bitów:

```
CKSEL3...0: 1111
SUT1...0: 11
CKDIV8: 1
CKOUT: 1
JTAGEN: 1
EESAVE: 1
BODLEVEL2: 110
```

Wykaz elementów:**Rezystory:** (SMD 0805)

R1, R3: 10 kΩ
 R2: 6,34 kΩ/1%
 R4, R5, R7, R9, R11, R12, R14...R16, R18: 3 kΩ
 R6, R10, R13, R19: 1 kΩ
 R8, R17: 8,2 kΩ
 R20, R21: 10 Ω

Kondensatory: (SMD 0805)

C1, C2, C8, C9: 22 pF
 C4: 470 nF
 C3, C5, C10, C13, C14: 100 nF
 C6, C7, C11: 10 μF/16 V (SMD „A”)
 C12: 1 μF

Półprzewodniki:

U1, U3 – MCP1703-3302E/DB (obudowa SOT223)
 U2 – MCP7940N-I/SN (obudowa SO08)
 U4 – ATMEGA644PA (obudowa TQFP44)
 U5 – NC7SB3157 (obudowa SC-70-6)
 U6, U7 – LTV817S-SMD (obudowa SO04)
 T1, T2 – BC817 (obudowa SOT23)
 D1, D2 – 1N4148 (obudowa MINIMELF)

Inne:

TFT – wyświetlacz graficzny Multi-Inno Technology MI0420CT-5 (kontroler FT1509, taśma ZIF 45 pin)
 L1 – dławik 10μH (obudowa SMD 0805)
 Q1 – rezonator kwarcowy 12,288 MHz niski
 Q2 – rezonator kwarcowy, zegarkowy 32768 Hz
 MENU/<, OK/> – microswitch THT wysokość 6 mm
 LPG – gniazdo IDC męskie typu P1303 (PIN3, raster 2,54mm)
 PB – gniazdo IDC męskie typu P1302 (PIN2, raster 2,54mm)
 TID_CON – gniazdo wyświetlacza Opel TID lub gniazdo IDC do druku (PIN 2×6, raster 2,54mm)

Mikrokontroler Atmega644-PA ma nielotną pamięć EEPROM, jednak zapewnia ona ograniczoną liczbę gwarantowanych cykli zapisu. Dlatego posłużono się pamięcią RAM dostępną w układzie zegara czasu rzeczywistego MCP7940N, której zawartość jest podtrzymywana nawet po wyłączeniu zasilania za pomocą baterii BAT. Dla wygody wszystkie dane obliczeniowe uformowano w strukturę zaopatrzoną w sumę kontrolną CRC8.

Kolejne słowo komentarza należy się układowi odpowiedzialnemu za pomiar temperatury zewnętrznej. Samochody marki Opel mają rezystancyjny czujnik temperatury zewnętrznej montowany w okolicach przedniego zderzaka. Jest to termistor o ujemnym współczynniku temperaturowym (NTC), którego zaciski dostępne są w złączu wyświetlacza TID. Skoro tak, to sposób pomiaru temperatury nasuwa się sam. Dysponując przetwornikiem A/C wystarczy zastosować dzielnik rezystancyjny, którego jednym z elementów jest termistor w pojeździe, a drugim dokładny rezystor o odpowiedniej rezystancji. Dzielnik rezystancyjny, o którym mowa zbudowano z termistora dostępnego na zaciskach NTC/NTC wyświetlacza TID oraz rezystora R2.

Wszystkie wejścia PORTA mikrokontrolera, do których w wewnętrznej strukturze układu są dołączone wejścia przetwornika A/C użyto do sterowania pracą wyświetlacza TFT. Jak w takiej sytuacji zmierzyć napięcie występujące na termistorze? Zastosowano tu pewną „sztuczkę”. Napięciem wejściowym (mierzonym) przetwornika A/C jest specjalne, wysokostabilne, wewnętrzne

źródło napięcia odniesienia VBG (bandgap), a napięcie z dzielnika NTC/R2 jest napięciem referencyjnym przetwornika A/C. W ten sposób odwrócimy „normalny” sposób pomiaru napięcia przez przetwornik A/C. Nie mając de facto do dyspozycji żadnego z wejściowych portów mikrokontrolera, do których dołączono wejścia przetwornika A/C, możemy dokonywać pomiaru napięcia korzystając głównie z faktu, iż przetwornik może mierzyć napięcie VBG.

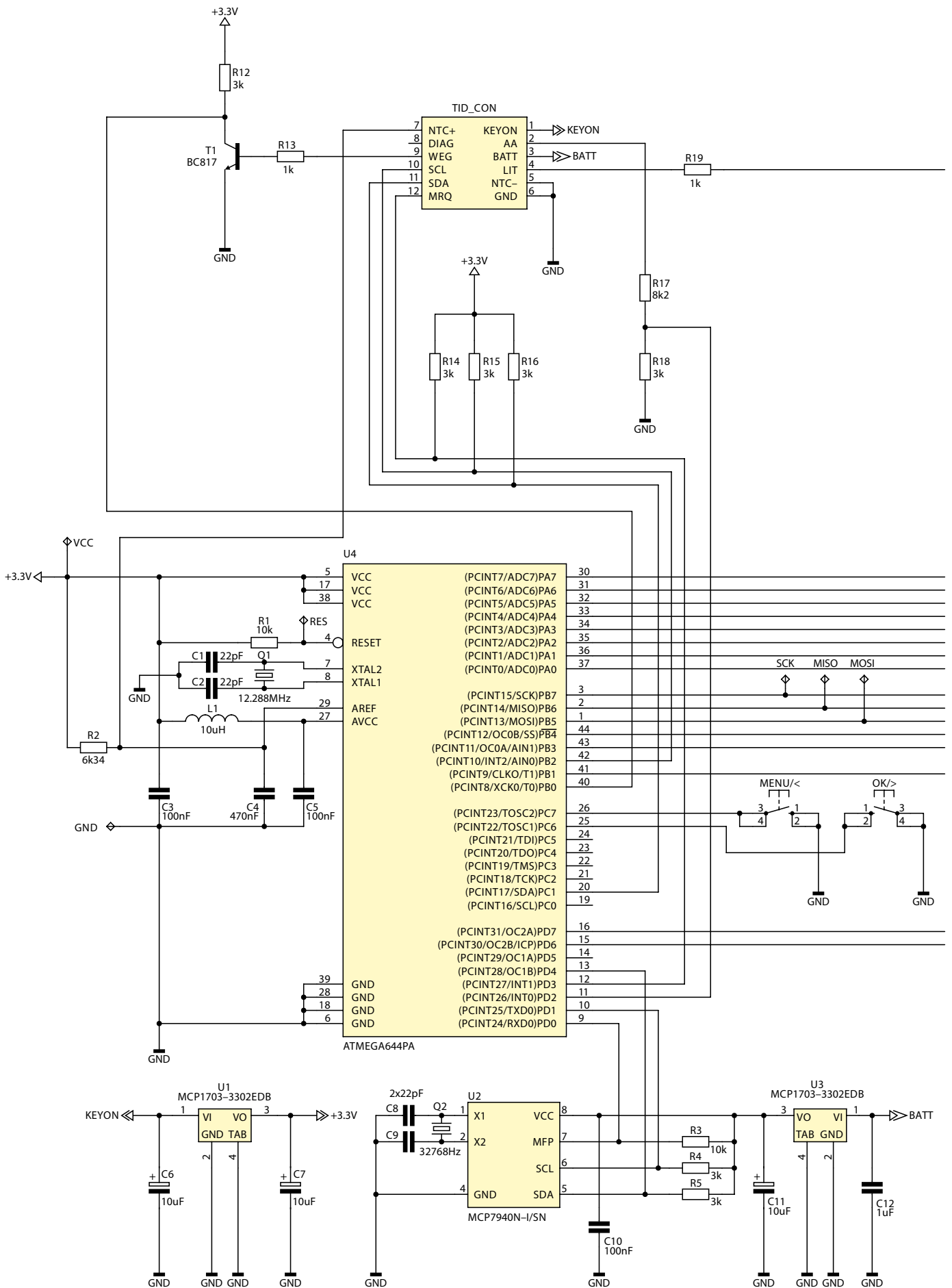
Napięcie wyjściowe dzielnika NTC/R2 będące zarazem napięciem referencyjnym (odniesienia) przetwornika A/C wyraża się wzorem $U_{WY} = U_{REF} = R_{NTC} / (R_{NTC} + R_2) \times 3,3 \text{ V}$. Z kolei, wartość zmierzona przez przetwornik A/C wyraża się wzorem $ADC = (U_{IN} \times 1024) / U_{REF}$, gdzie $U_{IN} = VBG = 1,1 \text{ V}$. Za pomocą obu wzorów otrzymujemy wyrażenie pozwalające wyznaczyć wartość mierzoną przez przetwornik A/C – $ADC = (1,1 \times 1024 \times (R_{NTC} + R_2)) / (3,3 \times R_{NTC})$. Tego sposobu możemy użyć prawie zawsze, gdy brakuje wejść, aby skoryzować z wbudowanego przetwornika A/C. By obliczyć wartość temperatury zewnętrznej, program obsługi aplikacji korzysta z predefiniowanej i zapisanej w pamięci Flash tablicy stałych odwzorowujących charakterystykę zastosowanego termistora NTC. Już zupełnie na koniec dodam, iż zdecydowaną większość pamięci programu aplikacji zajęły wzorce obrazków wyświetlanych w ramach graficznego interfejsu użytkownika (niektóre z nich, mimo kompresji, zajmują po 5 kB) oraz wzorce czcionek.

Robert Wołgajew, EP

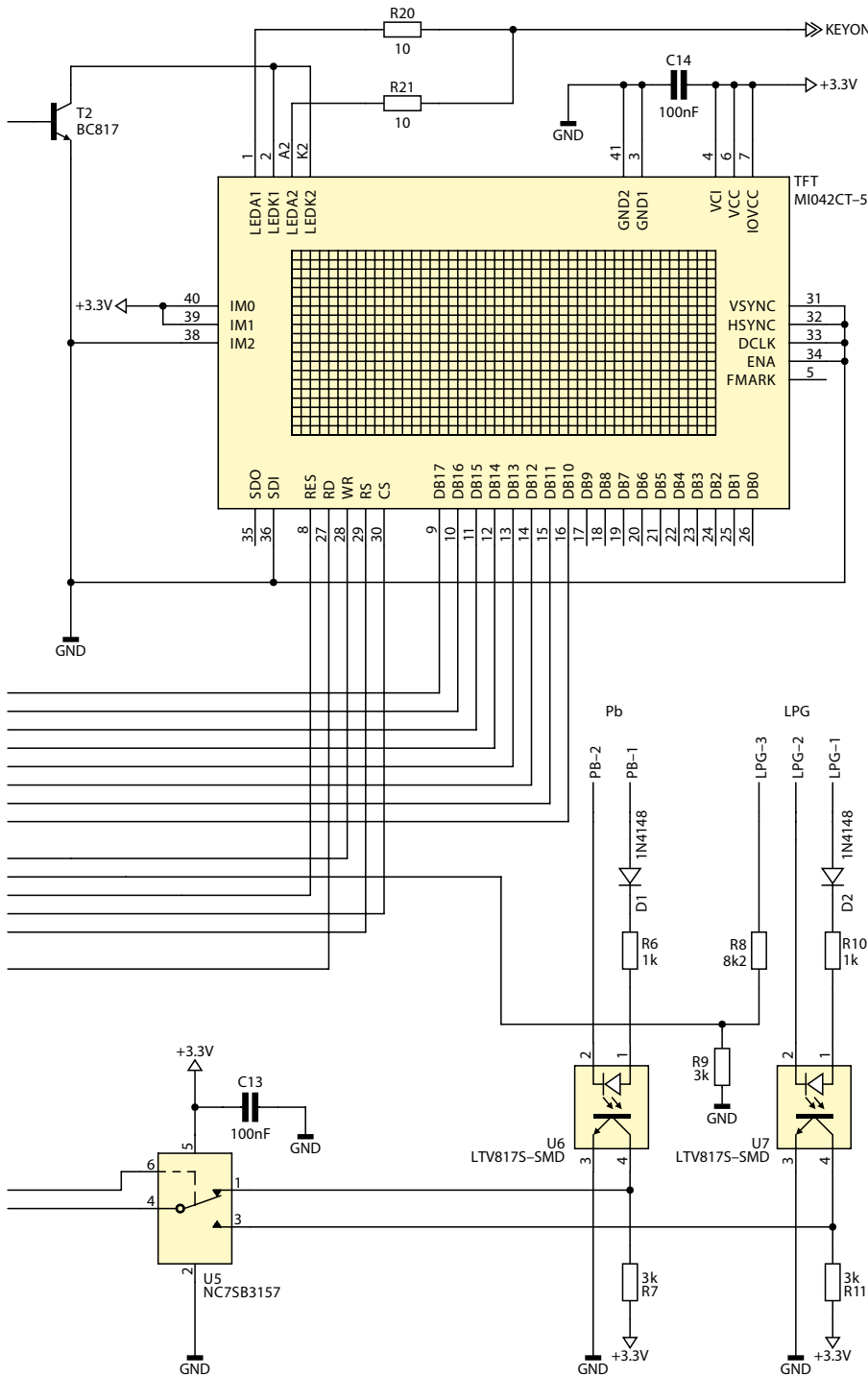
Listing 8. Wzory używane do obliczania wyświetlanych parametrów na ekranie

```
//
//Obliczamy spalone w czasie ostatniej sekundy paliwo [ul] - jeśli tylko nie jesteśmy w trybie kalibracji
if(Config.Calibration[Fuel] == 0) spentFuelPerls = ((10UL*Config.Cylinders*injectionTime*Config.CcPerMin[Fuel])/2880UL); else spentFuelPerls = 0;
//Akumulujemy wartości zużytego paliwa [ul] oraz przejechanego dystansu [m] niezbędne do obliczenia wartości średnich
Accu.spentFuel[Fuel] += spentFuelPerls;
Accu.Distance[Fuel] += ((10UL*WEGpulses*Config.Wheel) / (1000UL*Config.PulsPerRot));
//Obliczamy również ilość paliwa pozostającego w baku pojazdu [ul] pod warunkiem, że jeszcze jakieś pozostało
if(Accu.remainingFuel[Fuel] >= spentFuelPerls) Accu.remainingFuel[Fuel] -= spentFuelPerls;
//Dla potrzeb procedury kalibracji akumulujemy sumaryczny czas wtrysków, jeśli tylko ta procedura jest aktywna
if(Config.Calibration[Fuel]) Accu.injectionTime += injectionTime;
//Obliczamy wartości chwilowe dostępne z pomiarów wykonywanych co 1 sekundę
Speed = ((360UL*WEGpulses*Config.Wheel) / (1000UL*Config.PulsPerRot)); //Prędkosc chwilowa [km/h]
if(Speed > Accu.Vmax) Accu.Vmax = Speed; //Zapamiętujemy wartość maksymalnej prędkości
//Zużycie chwilowe dla wszystkich wtrysków: w [l/h *10] gdy Speed<=5 i w [l/100km *10] gdy Speed>5
if(Speed<=5) Consum = ((5UL*Config.Cylinders*injectionTime*Config.CcPerMin[Fuel]) / 400000UL);
else Consum = ((5UL*Config.Cylinders*injectionTime*Config.CcPerMin[Fuel]*Config.PulsPerRot) / (144UL*WEGpulses*Config.Wheel));
if(Consum > 999) Consum = 999; //Zabezpieczenie przed przekroczeniem zakresu
//Na podstawie akumulat. wartości zużytego paliwa i przejech. dystansu obliczamy wartości średnie prędkości oraz zużycia paliwa
SpeedAvg = ((360UL*Accu.Distance[Fuel]) / (100UL*Accu.Measurements[Fuel])); // [km/h]
if(Accu.Distance[Fuel]>999) ConsumAvg = (Accu.spentFuel[Fuel]/Accu.Distance[Fuel]); else ConsumAvg = 0; //l/100km *10
//Obliczamy dostępny zasięg pojazdu na pozostałym w baku paliwie [km] = pozostałe paliwo/średnie zużycie paliwa
if(Accu.spentFuel[Fuel]) availableDistance = (Accu.remainingFuel[Fuel]*Accu.Distance[Fuel]) / (1000UL*Accu.spentFuel[Fuel]); else availableDistance = 0;
if(availableDistance > 999) availableDistance = 999; //Zabezpieczenie przed przekroczeniem zakresu

/*
- InjectionTime - sumaryczny czas wtrysku zliczony w czasie 1s [ms*48]
- WEGpulses - liczba impulsów z przetwornika drogi zliczona w czasie 1 sekundy
- Fuel - bieżący rodzaj paliwa - zmienna ustawiana w procedurze przerwania PCINT1
- spentFuelPerls - bieżące paliwo spalane w czasie ostatniej sekundy [ul]
- Accu.Measurements - akumulator liczby interwałów pomiarowych [s] - oddzielnie dla każdego paliwa
- Accu.spentFuel - akumulator ilości spalonego paliwa [ul] - oddzielnie dla każdego paliwa
- Accu.remainingFuel - akumulator ilości paliwa pozostającego w baku [ul] - oddzielnie dla każdego paliwa
- Accu.Distance - akumulator przejechanego dystansu [m] - oddzielnie dla każdego paliwa
- Consum - chwilowe zużycie paliwa [l*10/h], dla prędkości<5 km/h lub [l*10/100km], dla prędkości>5 km/h
- ConsumAvg - średnie zużycie paliwa [l*10/100km]
- Speed - prędkość chwilowa [km/h]
- SpeedAvg - prędkość średnia [km/h]
- availableDistance - orientacyjny, dostępny dystans na paliwie pozostającym w baku [km] - oddzielnie dla każdego paliwa
- Config.CcPerMin - stała wtryskiwacza [ml/min] - oddzielnie dla każdego paliwa
- Config.PulsPerRot - stała przetwornika drogi [imp/obr]
- Config.Cylinders - liczba wtryskiwaczy paliwa
- Config.Wheel - obwód opony [cm]
*/
```



Rysunek 7. Schemat ideowy komputera samochodowego multiTID



DODATKOWE MATERIAŁY NA FTP:

<ftp://ep.com.pl>

USER: 33948, PASS: 5gckdmq

W ofercie AVT*

AVT-5562

Podstawowe informacje:

- Napięcie zasilania: 8...15 V DC.
- Maksymalny prąd obciążenia (z napięciem +12 V): 10 mA.
- Prąd podtrzymania zegara RTC (z napięciem BATT): 1 mA.
- Maksymalny prąd podświetlenia (z napięciem ILL+): 75 mA.
- Dokładność pomiaru temperatury: 1°C.
- Zakres pomiarowy temperatury zewnętrznej: -30...35°C.
- Zakres pomiarowy prędkości pojazdu: 0...255 km/godz.
- Zakres pomiarowy chwilowego zużycia paliwa: 0...99,9 l/100 km.
- Zakres pomiarowy średniego zużycia paliwa: 0...25,5 l/100 km.
- Zakres pomiarowy paliwa dostępnego w baku: 0...99,9 l.
- Zakres pomiarowy przejechanej odległości: 0...9999 km.
- Zakres pomiarowy dystansu do przejechania na dostępnym paliwie: 0...999 km.
- Zakresy regulacji parametrów konfiguracyjnych:
 - Stała wtryskiwaczy: 1...999 ml/min.
 - Stała przetwornika drogi: 1...99 impulsów/obrót.
 - Obwód opony: 50...255 cm.
 - Liczba cylindrów: 2...8.
 - Pojemności baków: 25...99 l.
- Przesunięcie belek informacyjnych: 0:9 pikseli.

Projekty pokrewne na FTP:

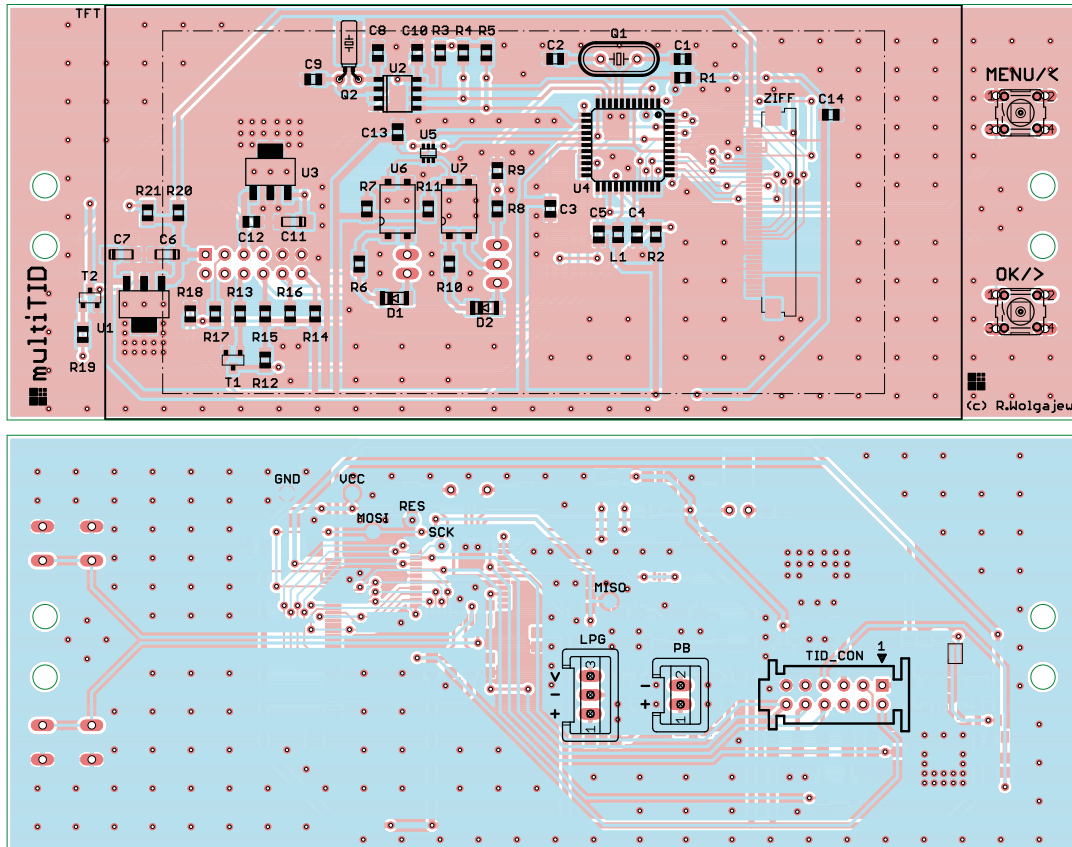
(wymienione artykuły są w całości dostępne na FTP)

- AVT-5545 Komputer samochodowy Mee MK II (2.0) (EP 7-8-9/2016)
- AVT-5495 Uniwersalny komputer samochodowy Mee (EP 3/2015)
- AVT-3095 Komputer samochodowy (EdW 4-5/2014)
- AVT-5405 TripCo – komputer samochodowy (EP 7/2013)
- AVT-5395 TIDex – komputer dla samochodów z silnikiem Diesla (EP 5/2013)

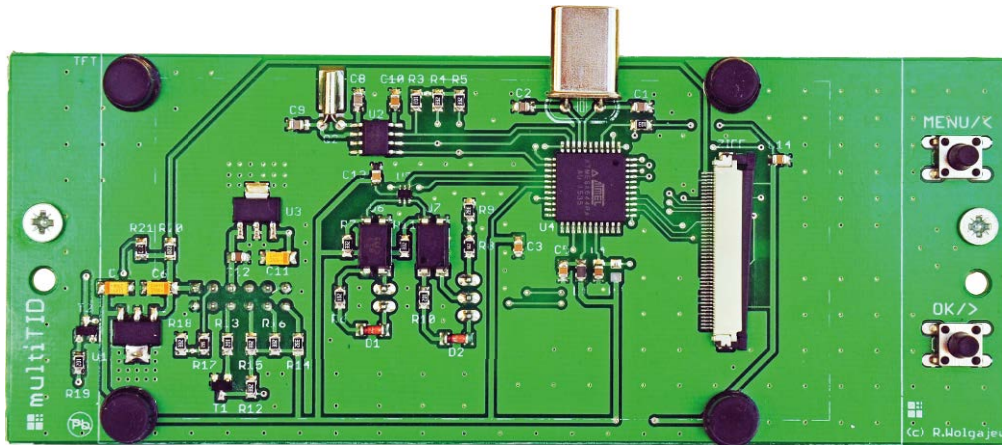
*** Uwaga:**
 Zestawy AVT mogą występować w następujących wersjach:
AVT xxxx UK to zaprogramowany układ. Tylko i wyłącznie. Bez elementów dodatkowych.
AVT xxxx A płytka drukowana PCB (lub płytki drukowane, jeśli w opisie wyraźnie zaznaczono), bez elementów dodatkowych.
AVT xxxx A1 płytka drukowana i 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 nic innego jak zmontowany zestaw B, czyli elementy wylutowane 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 ściągnąć, 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>

Tabela 6. Konfiguracja peryferiów Atmega644-PA dla potrzeb pomiarów

Nazwa	Sposób konfiguracji	Funkcjonalność
Timer0	Licznik impulsów zewnętrznych podawanych na wejście T0 – reakcja na zbocze opadające.	Liczy impulsy przetwornika drogi WEG zapewniając pomiar przejechanego dystansu.
Timer1	Licznik taktowany wewnętrznym przebiegiem o częstotliwości 48 kHz (prescaler=256, 48 taktów na 1 ms). Wejście Input/Capture dołączone do układu formującego przebieg z wtryskiwacza paliwa.	Mierzy sumaryczny czas wtrysku w czasie 1 s. zapewniając pomiar zużytego paliwa.
Przerwanie INT0	Wyzwalane zboczem opadającym na wejściu INT0, do którego doprowadzono wyjście częstotliwości wzorcowej zegara RTC.	Służy do odmierzania czas pomiaru równego 1 s.



Rysunek 8 Schemat montażowy komputera samochodowego multiTID



Fotografia 9 Wygląd obwodu drukowanego zmontowanego sterownika multiTID od strony elementów (bez wyświetlacza TFT)

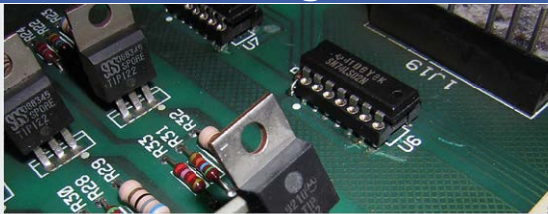
REKLAMA

Elektronika Praktyczna na facebook



Elektronika Praktyczna

- Strona główna
- Informacje
- Zdjęcia
- Osoby, które to lubią
- Filmy
- Posty
- [Utwórz stronę](#)



Lubisz to? Wiadomości Więcej

Status Zdjęcia / film

Napisz coś na tej stronie...

Elektronika Praktyczna
27 lipiec o 16:51 · [uk](#)

Kolejna konsolidacja - Analog Devices kupił Innear Technology. Obie firmy mają ogromny dorobek i znaczenie dla przemyślu elektronicznego.
<http://www.bloomberg.com/> <http://analog-devices.ead.in-advanced.com/>

Magazyn

Szukaj postów na tej stronie

3508 użytkowników lub to
Szymon Narcański i 8 innych znajomych

Zaproś znajomych do polubienia tej strony

INFORMACJE

- Sławomir Kasper lubi post użytkownika Bożena. Knecht na swojej osi czasu.
- Marek Witold Słonecznik Grabski lubi: Eastwest Rockers.
- Anna Bnej i Jerzy Chmielewski są teraz znajomymi.
- Seba Veł Zónek zareagował na post.
- Kasia Minczał lubi post użytkownika Agnieszka Fugęska.
- Maja Sorochnę skomentowała post użytkownika Marek Witold Słonecznik Grabski.
- Włodzisław Szwed lubi post użytkownika Marynka Wojnacka RP.
- Maciek Zieliński lubi post użytkownika Paweł Piłarczyk.
- Marcin Bednarek lubi zdjęcie użytkownika Teraz44.
- Adam Dębowski został oznaczony na zdjęciu użytkownika Jan.