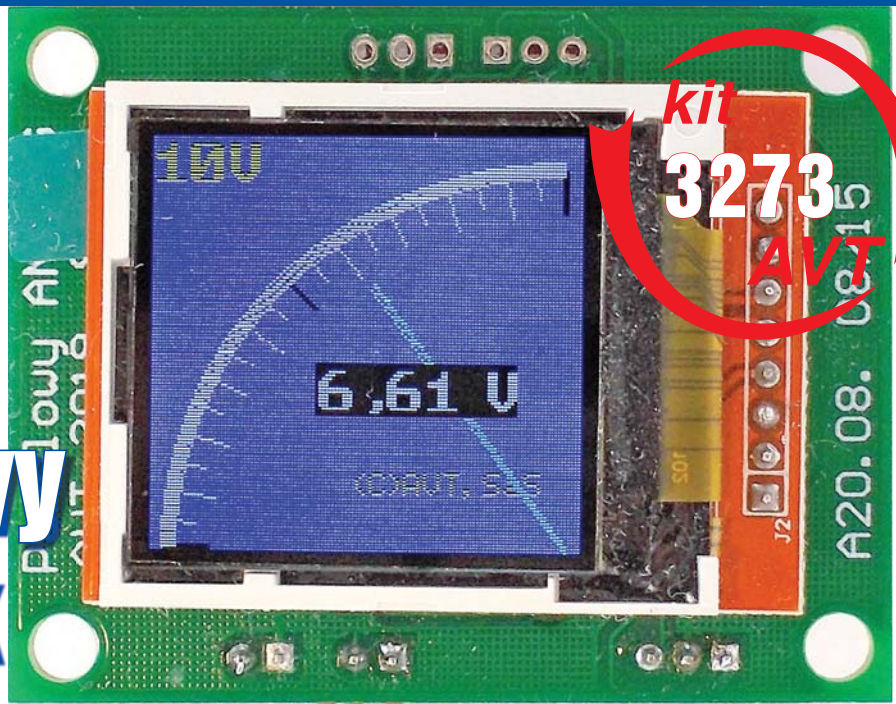




PAW - Pseudo- analogowy wskaźnik



Technika cyfrowa zdominowała wszystkie dziedziny elektroniki. Ma to zalety ale i wady. Wadą mierników z cyfrowym wyświetlaczem jest problem z określeniem kierunku zmian wielkości mierzonej, gdy ta zmienia się stosunkowo szybko. Migające cyferki niewiele pokazują. Czy ktośkolwiek wyobraża sobie wskaźnikysterowania wzmacniacza audio na wyświetlaczu w postaci cyfr? Z pewnością nie.

Do czego to służy?

Problem pokazywania w czytelny sposób szybko zmieniających się pomiarów dostrzegli producenci multimetrów. W lepszych ich wykonaniach dodawany jest bargraf. Spotyka się też konstrukcje łączące miernik wskazówkowy z cyfrowym – **fotografia 1**. A gdyby tak na wyświetlaczu graficznym symulować miernik wskazówkowy? Widziałem już takie rozwiązanie, ale na wyświetlaczu monochromatycznym. Kolorowy wydał się ciekawszy. Powstała więc konstrukcja zaprezentowana w artykule. Efekt działania urządzenia można zaobserwować na filmach: <https://www.youtube.com/playlist?list=PLdtkbzWTUVMkv3nDujB1Cpzz0gawlG2V>.

Fot. 1

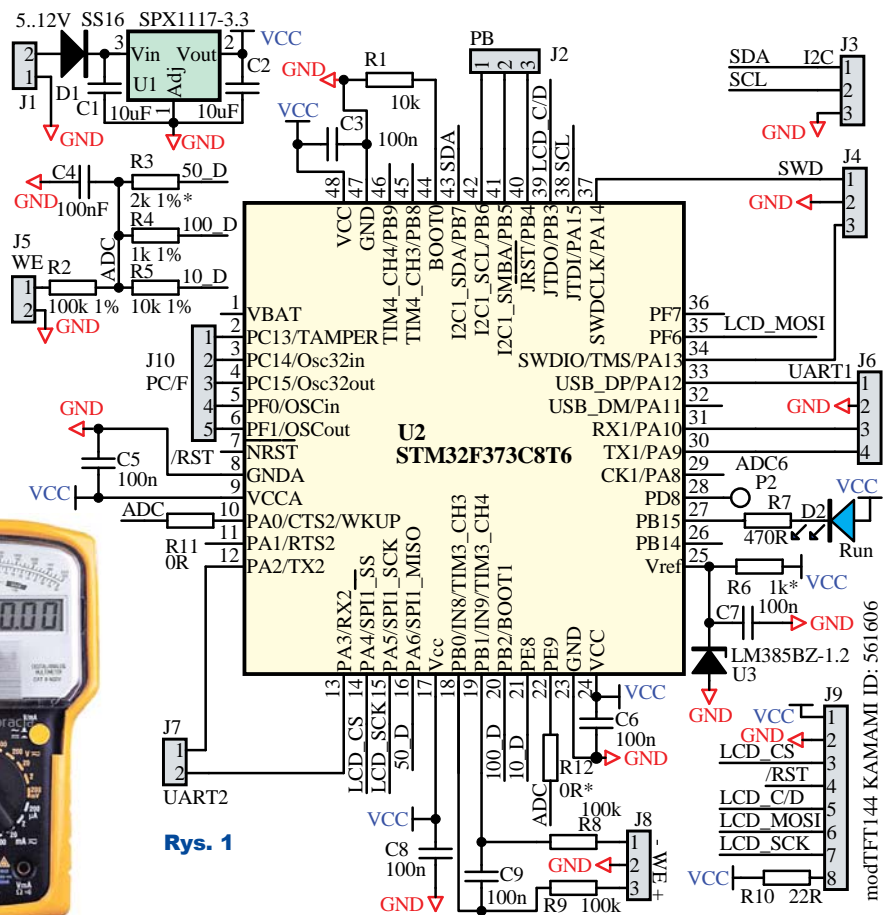


Jak to działa?

Schemat ideowy pokazany jest na **rysunku 1**. Układ zasilany jest napięciem z zakresu 5–12V stabilizowanym układem U1. Dioda

D1 zabezpiecza urządzenie przed skutkami błędnego przyłączenia zasilania. Mierzone napięcie jest doprowadzone do złącza J5. Rezystory R2 z R3, R4 lub R5 tworzą dzielnik wejściowy. Cztery zakresy pomiarowe uzyskano, przyłą-

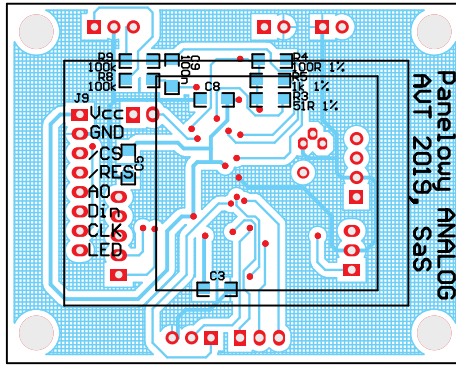
czając jeden z rezystorów R3, R4 lub R5 do masy lub żaden z nich. Wyprowadzenia mikrokontrolera przyłączone do rezystorów pracują w trybie Open Drain. Dzięki temu albo są połączone z masą (stan niski wyjścia), albo są



Rys. 1

modTFT144 KAMAMI ID: 561606

w stanie wysokiej impedancji (w zakresie napięć od -600mV do +5,5V). Kluczową rolę odgrywa mikrokontroler U2. Wyniki pomiarów prezentowane są na kolorowym wyświetlaczu THT o rozdzielczości 128x128 pikseli. W urządzeniu przewidziano możliwość wykorzystania zewnętrznego napięcia odniesienia U3 ale jego użycie ma sens tylko wtedy, gdy przeprowadzi się czasochłonną procedurę kalibracji przetwornika ADC. Złącze J6 służy do programowania mikrokontrolera. J2 i J7 mogą służyć do komunikacji miernika z innym mikrokontrolerem lub komputerem. Pozostałe złącza nie są używane, służyły do testów. Program dla mikrokontrolera, zarówno w postaci źródłowej, jak i pliki wynikowe (BIN, Intel-Hex S-recordy), jest umieszczony na Elportalu wśród materiałów dodatkowych do tego numeru EdW. Można, a wręcz trzeba dostosować go do swoich potrzeb. Zmiany mogą dotyczyć wskazywanych wielkości, liczby zakresów pomiarowych, kolorów. Możliwe jest, jak na filmie <https://www.youtube.com/watch?v=AgA4Gt2vriA&list=PLdtkbzWTUVMkv3nDujB1Cpzz0gawIG2V&index=3&t=0s>, uzyskanie kolorowej skali czy też zmiany koloru wskaźnika cyfrowego bądź wskaźniki, zależnie od wartości mierzonej wielkości <https://youtu.be/82TjEMmNTn4?list=PLdtkbzWTUVMkv3nDujB1Cpzz0gawIG2V>.



Rys. 2

Montaż i uruchomienie

Układ można zmontować na płytce drukowanej, której projekt pokazany jest na rysunku 2. Układ montujemy, zaczynając od elementów najmniejszych, a kończąc na największych.

Fotografia wstępna oraz **fotografie 2 i 3** pokazują model.

Osoby niedoświadczone powinny poprosić kogoś o pomoc w zaprogramowaniu procesora. Zmontowany układ nie wymaga uruchamiania, jeżeli ma służyć do pomiaru napięcia w zakresie 1...100V, a skala jest jednobarwna. Po włączeniu zasilania pokaże się ekran tytułowy widoczny na **fotografii 4**. W modelu pobór prądu wynosił około 50mA.

Miernik można skalibrować. Kalibracja nie jest bezwzględnie konieczna. Bez niej trzeba się jednak liczyć z większymi błędami pomiarowymi. W tym celu w kodzie programu, należy odnaleźć deklarację:

```
#define VREF_VCC 3.3
```

po czym wpisać faktyczne napięcie zasilające mikrokontroler. W prototypie było to:

```
#define VREF_VCC 3.2984
```

Po tym należy skompilować kod i wgrać do mikrokontrolera. Jak to zrobić, można przeczytać na końcu artykułu. Kolejne modyfikacje polegają na odnalezieniu:

```
#define KALIBRACJA_DZIELNIKA_0
```

przypisując jej wartość „1.0” („1.0” nie „1”):

```
#define KALIBRACJA_DZIELNIKA_0 1.0
```

Po kompilacji i uruchomieniu programu należy podać napięcie odpowiadające około 90% zakresu 0. W przypadku domyślnych ustawień będzie to 900mV. Po odczytaniu wartości wskazanej przez miernik należy przeprowadzić proste obliczenie:

$$\text{korekta} = U_{we} / U_{odczytane}$$

na przykład, odczytano napięcie 853mV, na wejście podano 910mV, wartość korekty wyniesie:

$$910 / 850 = 1,0668229772567$$

W kodzie źródłowym należy więc zmienić deklarację na:

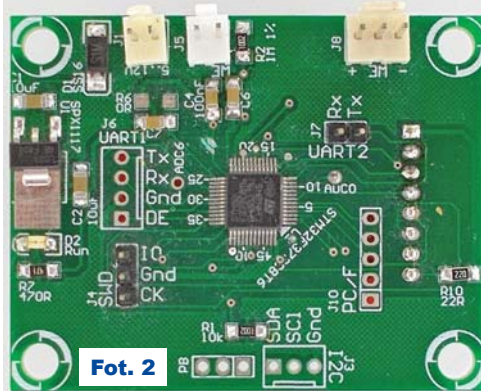
```
#define KALIBRACJA_DZIELNIKA_0 1.0668229772567
```

Tę samą procedurę należy przeprowadzić dla pozostałych używanych zakresów pomiarowych. W deklaracji istotne jest (typ float) siedem cyfr znaczących. Kalibracja nie rozwiąże problemu nieliniowości przetwornika ADC. Ze względu na to, że wskaźnik panelowy najczęściej pokazuje wielkości w wąskiego zakresu, jak na przykład napięcie sieciowe 230V, to aby uniknąć czasochłonnej kalibracji nieliniowości, której opis, ze względu na stopień skomplikowania pominię, wskaźnik można skalibrować dla środka zakresu mierzonej wielkości, w przypadku 230V będzie to właśnie to napięcie. Gdyby kalibracja dotyczyła wskazania temperatury z zakresu 15...35 stopni, kalibrację należy przeprowadzić dla 25 stopni = (15+35) / 2.

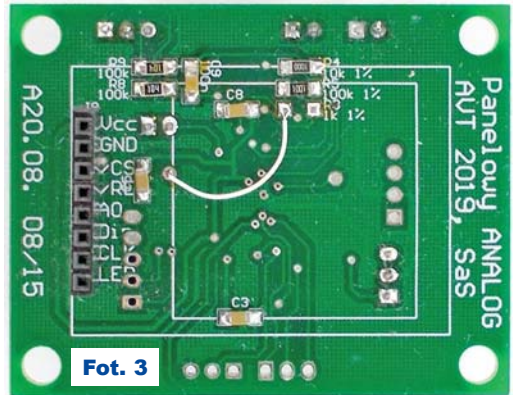
Możliwość zmian

Aby ułatwić wprowadzenie zmian, pokrótce opiszę najważniejsze stałe zdefiniowane w programie.

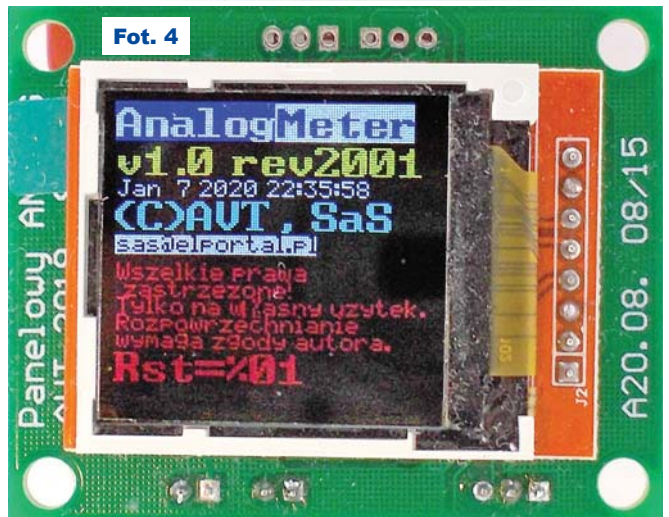
```
#define AUTO_ON //Wykomentowanie wyłączy automatykę oraz ustawi najwyższy zakres pomiarowy
```



Fot. 2



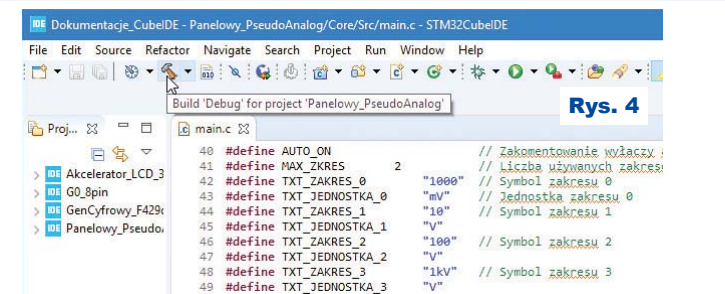
Fot. 3



Fot. 4



Rys. 5



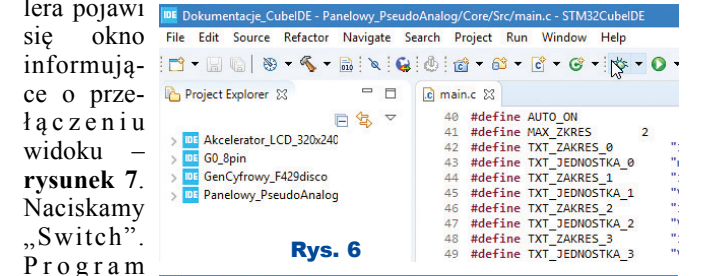
Rys. 4

```
t1 = readTimSys();
PrintWskaznik( wartosc );
t1 = readTimSys() - t1;
//----- END wskazówka

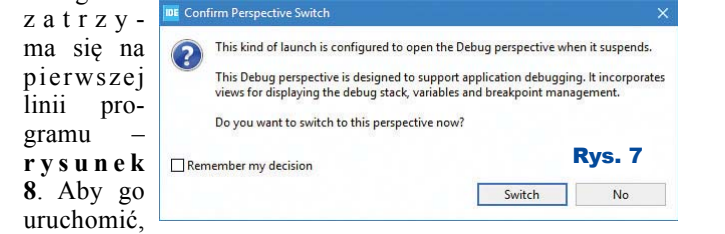
bezpośrednio przed:
//----- BEGIN wysłanie danych do wyświetlacza
t2 = readTimSys();
TFT_ILI9163C_Display(); // TFT_ILI9163C_DisplayDMA();
t2 = readTimSys() - t2;
//----- END wysłanie danych do wyświetlacza
```

Kompilacja i programowanie mikrokontrolera

Projekt wskaźnika zrealizowano z wykorzystaniem środowiska CubeIDE, które można pobrać ze strony producenta mikrokontrolerów STM32 <https://www.st.com/en/development-tools/stm32cubeide.html#tools-software>. Gdy środowisko jest zainstalowane i pobierze wymagane pliki z Internetu, można otworzyć projekt, klikając w ikonę projektu **rysunek 3** (**Uwaga!** Wszystkie rysunki – zrzuty są też umieszczone w Elportalu). Kod kompilujemy, naciskając symbol młotka – **rysunek 4**. Raport z kompilacji ukazuje się w lewym oknie na dole, po prawej znajdziemy informacje o zajętości pamięci mikrokontrolera – **rysunek 5**. Program wgrywamy do mikrokontrolera, naciskając symbol robaka – **rysunek 6**. Zanim to zrobimy, należy podłączyć programator do mikrokontrolera. Liniję SwdCK przyłączamy do wyprowadzenia 1 złącza J4, SwdIO do 3 J4, masę do 2 J4. Po wgraniu programu do mikrokontrolera pojawi się okno informujące o przełączeniu widoku – **rysunek 7**. Naciskamy „Switch”. Program z a t r z y m a się na pierwszej linii programu – **rysunek 8**. Aby go uruchomić,



Rys. 6



Rys. 7

rysunek 8. Aby go uruchomić,

```
#define MAX_ZKRES 2 //Liczba używanych zakresów pomiarowych
#define TXT_ZAKRES_0 "1000" //Symbol zakresu 0
#define TXT_JEDNOSTKA_0 "mV" //Jednostka zakresu 0
#define TXT_ZAKRES_1 "10" //Symbol zakresu 1
#define TXT_JEDNOSTKA_1 "V"
#define TXT_ZAKRES_2 "100" //Symbol zakresu 2
#define TXT_JEDNOSTKA_2 "V"
#define TXT_ZAKRES_3 "1kV" //Symbol zakresu 3
#define TXT_JEDNOSTKA_3 "V"
```

Nie powinno być wątpliwości, czemu służą definicje. Zwracam tylko uwagę, że tekst „TXT_ZAKRES” w połączeniu z „TXT_JEDNOSTKA” jest wyświetlany w lewym górnym rogu wyświetlacza, natomiast „TXT_JEDNOSTKA” za wynikiem pomiaru na wskaźniku cyfrowym.

Nazwa	Roz.	Wielkość
..settings	<DIR>	<DIR>
Core	<DIR>	<DIR>
Debug	<DIR>	<DIR>
Drivers	<DIR>	<DIR>
..cproject		28 027
..mxproject		8 985
..project		1 310
Panelowy_PseudoAnalog	ioc	9 644
Panelowy_PseudoAnalog Debug	launch	6 035
STM32F373C8TX_FLASH	ld	5 952

Rys. 3

„POMIAR_ADC” decyduje, czy wykorzystywany jest szybki przetwornik ADC 12-bit, czy wolny 16-bit. Aktualnie oprogramowanie obsługuje tylko przetwornik 12-bit. „VREF_VCC” to napięcie zasilania mikrokontrolera. W przypadku wykorzystania przetwornika 16-bit istotna jest wartość napięcia Vref dostępnego na wyprowadzeniu 25 mikrokontrolera.

```
#define D_WE_0 1
#define D_WE_1 11
#define D_WE_2 101
#define D_WE_3 1001
```

definiują wartość podziału dzielników napięcia zbudowanego z R2 oraz R3, R4, R5. Oprogramowanie wskaźnika dostępne w Elportalu używa trzech zakresów pomiarowych. Definicje:

```
#define KALIBRACJA_DZIELNIKA_0 1.0
#define KALIBRACJA_DZIELNIKA_1 1.0
#define KALIBRACJA_DZIELNIKA_2 1.0
#define KALIBRACJA_DZIELNIKA_3 1.0
```

były już opisane przy okazji kalibracji. Definicja włącza kolorową skalę

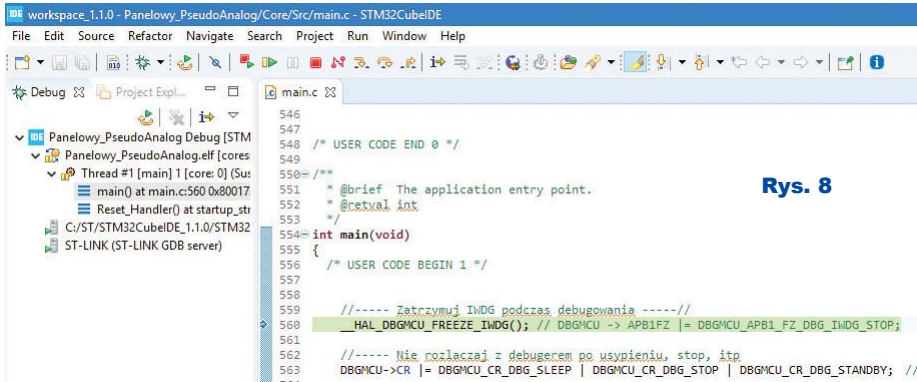
```
//#define SKALA_KOLOR // Definicja włącza kolorową skalę
#define KAT_SKALI_GORA 20.0
#define KAT_SKALI_DOL 30.0
#define KOLOR_SKALI_GORA ANSI_RED
#define KOLOR_SKALI_SRODEK ANSI_GREEN
#define KOLOR_SKALI_DOL ANSI_YELLOW
```

Powyższe włączają zakres i kolory skali, natomiast:

```
//#define DIGITAL_KOLOR // Definicja włącza kolorowy wskaźnik cyfrowy
#define WSKAZANIE_ZAKRES_KOLOR_Y 200
#define WSKAZANIE_ZAKRES_KOLOR_G 205
#define WSKAZANIE_ZAKRES_KOLOR_R 250
```

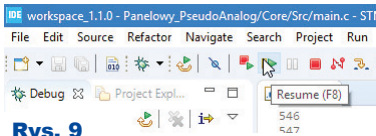
umożliwia zmianę koloru wskaźnika cyfrowego zależnie od mierzonej wartości. Na filmach widać, że wskazówka chowa się za wskaźnikiem cyfrowym. Nie jest to naturalne, ale taka jest „magia kina”. Dzięki temu nienaturalnemu zachowaniu wskaźnika i wskazówki jest on bardziej czytelny. Można to zmienić, przynosząc fragment:

```
//----- BEGIN wskazówka
```

Rys. 8

należy nacisnąć przycisk „PLAY” – **rysunek 9** lub klawisz F8. Instalację i obsługę CubeIDE opisałem bardzo



Rys. 9

pobieżnie. Wynika to z tego, że nie da się w kilku zdaniach opisać nawet podstawowej obsługi tak potężnego narzędzia. Na ten cel trzeba by przeznaczyć kilkanaście stron czasopisma. Jeśli Czytelnicy są zainteresowani takim artykułem, proszę o e-maile do redakcji.

SaS
sas@elportal.pl

Wykaz elementów

Rezystory 1206

R1	10kΩ
R2	100kΩ 1%
R3	2kΩ 1%*
R4	1kΩ 1%
R5	10kΩ 1%
R6	1kΩ*
R7	470Ω
R8,R9	100kΩ
R10	22Ω
R11,R12*	0Ω

Kondensatory 1206

C1,C2	10uF
C3,C4,C5,C6,C7,C8,C9	100nF
U1	SPX1117-3.3 SOT-223
U2	STM32F373C8T6
U3	LM385BZ-1.2G* TO-92B
D1	SS16
D2	Dioda LED Niebieska 1206
J1	NS25-W2K
J3	NS25-W3P
J4	NS25-W3P
J2,J5,J6,J7,J8,J10,P2*	do przyszłych zastosowań
J9	Wyświetlacz TFT 128x128 1,8” modTFT144 KAMAMI ID: 561606

Elementów oznaczonych gwiazdka nie montujemy.

Komplet podzespołów z płytką jest dostępny w Sklepie AVT jako zestaw AVT3273