

# Miernik wybranych parametrów środowiskowych



W artykule przedstawiono autorski projekt urządzenia elektronicznego będącego przyrządem służącym do pomiarów parametrów środowiskowych: wartości temperatury, ciśnienia, wilgotności i stężenia dwutlenku węgla. Ponadto, przyrząd może działać autonomicznie, bez konieczności wcześniejszego konfigurowania i kalibrowania, ma kompaktowe gabaryty i cechuje się niskim zużyciem energii.

**Rekomendacje:** miernik może przydać się do konstrukcji stacji klimatycznej lub jako miernik parametrów środowiska wewnątrz szklarni, palmiarni czy oranżerii i w innych miejscach wymagających monitorowania parametrów środowiska.

Środowisko geograficzne, które według definicji jest rozumiane jako całość elementów przyrodniczych oraz wytworów ludzkiej działalności, od zawsze stanowiło fundamentalną wartość dla ludzi i tworzonych przez nie społeczności. Ta istotność wynika z faktu, iż zdefiniowane w ten sposób środowisko otacza człowieka przez całe życie tworząc tym samym przestrzeń, w której żyje. Do podstawowych parametrów środowiska zaliczyć można temperaturę powietrza, wilgotność względną, ciśnienie atmosferyczne, natężenie światła, stężenie gazów w powietrzu, poziom opadów, żyzność gleby, zasoby mineralne i wiele innych. Czynniki te związane są ściśle z wieloma aspektami ludzkiej aktywności. Korzystne wartości parametrów środowiskowych decydują np. o komforcie życia i bezpieczeństwie w miejscu pracy, wpływają również na działalność ludzką o charakterze ekonomicznym. Dziedziny te pokazują, że istnieje wiele miejsc, gdzie zachodzi potrzeba nieustannego monitoro-



**AVT  
5328**

wania środowiska i w razie potrzeby wpływania na nie w celu uzyskania pożądanych wartości pewnych parametrów. Wśród licznych miejsc wymagających wykonywania pomiarów środowiskowych wymienić można np. domy mieszkalne, stanowiska pracy (szczególnie niebezpieczne miejsca pracy), klimatyzowane i wentylowane pomieszczenia, fermy hodowlane, szklarnie i inne obiekty z roślinami uprawnymi, magazyny żywności, serwerownie, laboratoria itp. Aby zautomatyzować pomiary stosuje się wyspecjalizowane przyrządy, które wyposażone są w czujniki różnych parametrów środowiska.

## Koncepcja budowy i dobór podzespołów

Zaprojektowanie i wykonanie urządzenia o opisanej funkcjonalności wymaga dobrania odpowiednich komponentów. Przede wszystkim czujników umożliwiających pomiar wybranych parametrów otoczenia. Kolejnym niezbędnym elementem jest wyświetlacz LCD, który pozwoli na wizualizację pomiarów. Następnym wymaganym podzespołem jest mikrokontroler, który będzie w stanie sterować całym urządzeniem realizując odbiór danych z czujników i przetwarzając je na pomiary, a następnie odpowiednio sterować wyświetlaczem LCD. Ostatnim potrzebnym w urządzeniu modułem jest blok zasilania. Schemat blokowy pokazujący wzajemne relacje między podzespołami składającymi się na urządzenie pokazano na **rysunku 1**.

W projekcie wykorzystano układ z rodziny EFM32 Gecko firmy Energy Micro. EFM32

**AVT-5328 w ofercie AVT:**  
AVT-5328A – płytka drukowana

### Podstawowe informacje:

- Zastosowanie mikrokontrolera EFM32G210F128 z rdzeniem ARM Cortex-M3.
- Pomiar parametrów środowiskowych: temperatury, wilgotności, ciśnienia atmosferycznego, zawartości dwutlenku węgla.
- Wizualizacja danych z czujników na wyświetlaczu LCD.
- Zasilanie ze źródła napięcia 6...9 V.

### Dodatkowe materiały na CD/FTP:

- <ftp://ep.com.pl>, user: 18453, pass: 5ey1854
- wzory płytek PCB
- karty katalogowe i noty aplikacyjne elementów oznaczonych w **Wykazie elementów** kolorem czerwonym

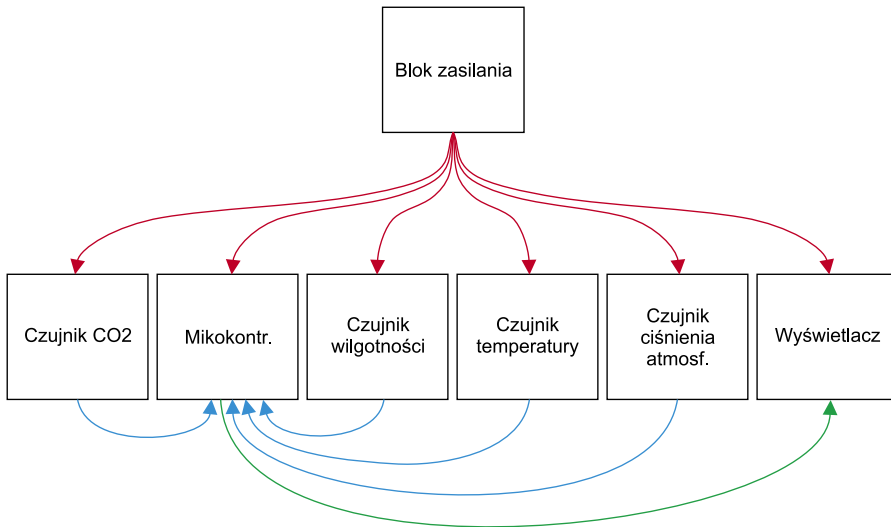
### Projekty pokrewne na CD/FTP:

- (wymienione artykuły są w całości dostępne na CD)
- AVT-5294 Stacja pogodowa z prognozowaniem pogody (EP 6/2011)

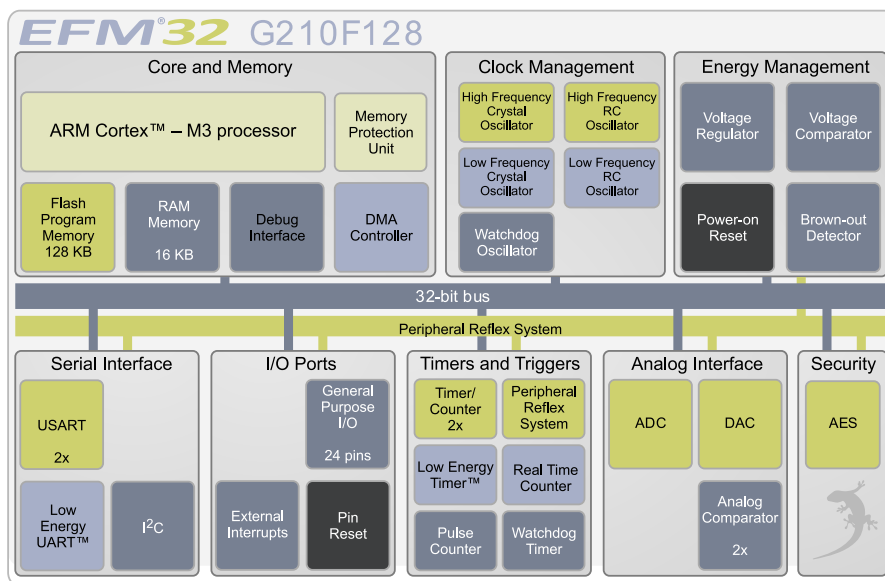
Autor dziękuje pani Annie Szewczyk za inspirację do stworzenia niniejszego projektu i artykułu.

Autor dziękuje firmie Energy Micro w osobie Pana Øyvind Grotmola za pomoc w realizacji projektu poprzez bezpłatne dostarczenie mikrokontrolerów EFM32 oraz zestawu EFM32 Gecko Starter Kit.

to rodzina mikrokontrolerów, której stworzeniu przyświecała idea zaprojektowania konstrukcji cechującej się przede wszystkim nowoczesną architekturą i dużą wydajnością, a jednocześnie rozbudowaną warstwą zarządzania energią. Uwzględniając te założenia producent zdecydował się zintegrować w strukturze swoich układów 32-bitowy rdzeń ARM Cortex-M3 (z możliwością taktowania do 32 MHz), pamięci Flash i SRAM, funkcjonalne zasoby cyfrowe i analogowe (w tym energooszczędny interfejs UART i Timer) oraz szereg funkcjonalności pozwa-



Rysunek 1. Schemat blokowy urządzenia



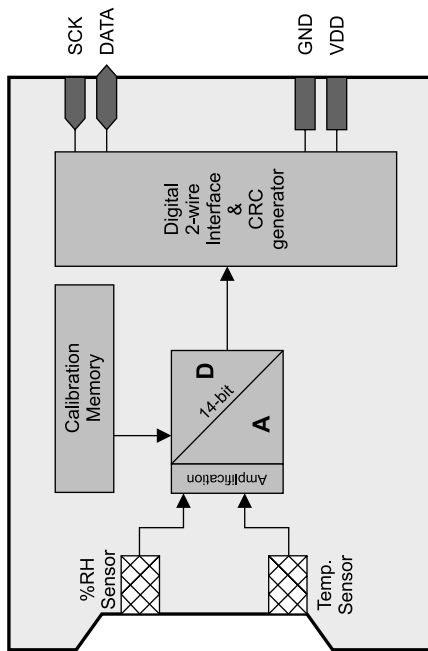
Rysunek 2. Schemat blokowy mikrokontrolera EFM32G210F128

Tabela 1. Najważniejsze dane techniczne mikrokontrolera EFM32G210F128	
Nazwa parametru	Wartość
Rdzeń	ARM Cortex-M3
Napięcie zasilania	1.8 V – 3.8 V
Częstotliwość taktowania	do 32 MHz
Wielkość pamięci FLASH	128 kB
Wielkość pamięci SRAM	16 kB
Liczba i rodzaj liczników	Dwa 16-bitowe z funkcją zliczania, porównania i PWM, jeden 16-bitowy o niskim poborze prądu, licznik impulsów, zegar RTC, Watchdog, licznik systemowy.
Peryferia analogowe	Przetwornik A/C (rozdzielczość 12 bit), przetwornik C/A (rozdzielczość 12 bit), komparator analogowy.
Interfejsy komunikacyjne	U(S)ART, SPI, I <sup>2</sup> C, UART o niskim poborze prądu (baudrate do 9600)
Interfejs programowania/debugowania	Serial Wire Debug, bootloader (UART)
Pozostałe zasoby	DMA, moduł szyfrujący AES
Liczba pinów I/O	24
Temperatura pracy	od -40 °C do 85 °C
Obudowa	QFN32
Ilość wyprowadzeń	32
Pobór prądu	Normalna praca (tryb EM0): 180 µA/MHz Tryb uśpienia EM1: 45 µA/MHz Tryb uśpienia EM2: 0,9 µA/MHz Tryb uśpienia EM3: 0,6 µA/MHz Tryb uśpienia EM4: 20 nA/MHz

lających obniżyć pobór prądu (np. technologię *Peripheral Reflex System* umożliwiającą komunikację międzymodułową bez użycia procesora). Szerzej na temat tej serii mikrokontrolerów można przeczytać na łamach EP 10/2010, EP 2...5/2011. W wymienionych wydaniach *Elektroniki Praktycznej* zamieszczono artykuły przedstawiające szczegółowo opis rodziny EFM32 (cechy charakterystyczne, odmiany układów, zestawy rozwojowe, dostępne oprogramowanie), wprowadzenie do opartego na platformie Eclipse zintegrowanego środowiska programistycznego (instalacja, konfiguracja, zakładanie nowych projektów, sposób debugowania kodu) oraz sposób testowania i optymalizacji zużycia energii tychże układów. W wykonanym projekcie urządzenia pomiarowego wykorzystano układ EFM32G210F128. Jego budowę wewnętrzną przedstawiono na **rysunku 2**, a wybrane parametry techniczne w **tabeli 1**.

Kolejnymi niezbędnymi elementami konstrukcyjnymi zaprojektowanego miernika są komponenty pomiarowe. Pierwszym wybranym na potrzeby projektu czujnikiem jest popularnie stosowany w różnych opisywanych na łamach EP projektach układ Sensirion SHT11, który realizuje pomiar dwóch parametrów: wilgotności i temperatury. Największymi zaletami produktu firmy Sensirion są: wysoka dokładność, małe gabaryty, mały pobór prądu oraz szeroki zakres napięcia zasilania. Czujnik składa się z dwóch pomiarowych struktur sensorowych (oddzielnie do pomiaru wilgotności i temperatury) przekształcających mierzoną wartość na analogowy sygnał elektryczny, wzmacniacza, przetwornika analogowo-cyfrowego, pamięci ze współczynnikami korekcyjnymi oraz modułu umożliwiającego cyfrową komunikację czujnika ze światem zewnętrznym. Schemat blokowy czujnika SHT11 pokazano na **rysunku 3**. Aplikacja czujnika jest nieskomplikowana i wymaga tylko jednego elementu zewnętrznego – rezystora „podciągającego” na linii danych. Odczyt danych odbywa się za pomocą zaimplementowanego w nim protokołu transmisyjnego. Zewnętrzny układ sterujący przesyła do czujnika ramkę skła-

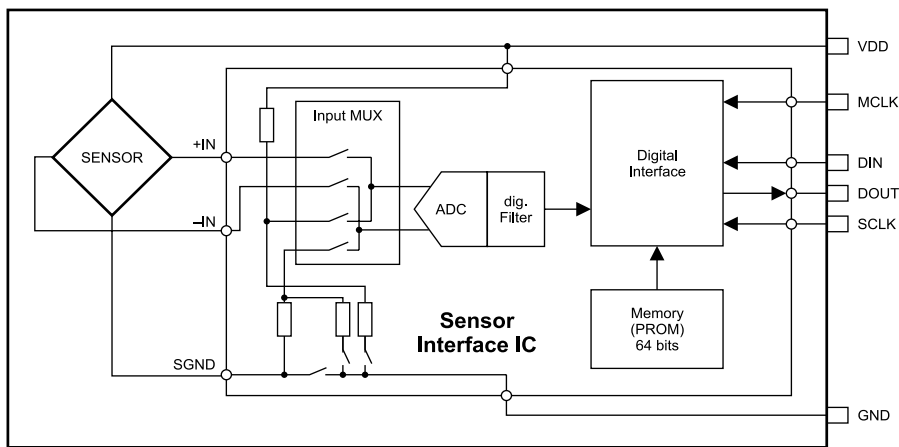
REKLAMA



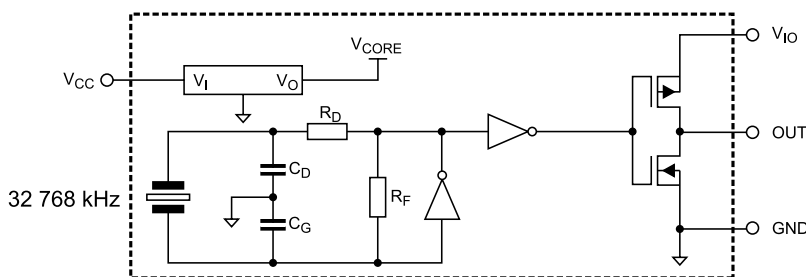
Rysunek 3. Schemat blokowy czujnika wilgotności i temperatury SHT11

dającą się z sekwencji inicjującej transmisję, adresu czujnika, rodzaju parametru do odczytu (wilgotność lub temperatura). Czujnik potwierdza poprawne przyjęcie danych i przystępuje do realizacji żądanego pomiaru (czas trwania od 20 ms do 320 ms w zależności od ustawionej rozdzielczości). Po wykonaniu pomiaru dane w postaci cyfrowej są odsyłane do układu sterującego i przeliczane na wartość temperatury lub wilgotności. Ponadto, opcjonalnie może być transmitowana suma kontrolna umożliwiająca weryfikację poprawności danych odebranych.

Jako czujnik ciśnienia atmosferycznego zastosowano układ MS5534A firmy Inter-



Rysunek 4. Schemat blokowy czujnika ciśnienia atmosferycznego MS5534A

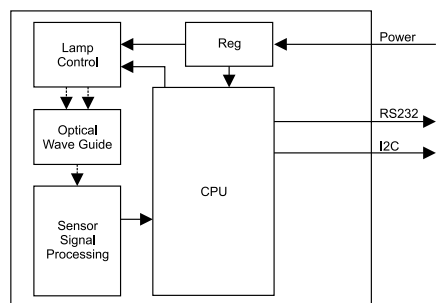


Rysunek 5. Schemat blokowy generatora SG-3032JC

sema (rysunek 4). Ma on podobne zalety funkcjonalne, co układ SHT11: duża dokładność, kompaktowe wymiary i niskie zapotrzebowanie na energię. W strukturze czujnika można wyodrębnić następujące bloki funkcjonalne: piezorezystancyjną strukturę sensorową z wyjściem analogowym, przetwornik A/C, pamięć z zapisanymi współczynnikami korekcyjnymi oraz moduł komunikacyjny. Czujnik MS5534A został wyposażony w interfejs SPI. Komunikacja z czujnikiem przebiega na drodze kierowanych do niego zapytań, w których znajdują się kody parametrów lub instrukcji (dane kalibracyjne, wartość temperatury, wartość ciśnienia, reset układu). Procedura pomiarowa wymaga w pierwszej kolejności odczytania z pamięci czujnika czterech 16-bitowych słów kalibracyjnych, które zamieniane są na współczynniki korekcyjne. Po tej czynności można przystąpić do odpytywania czujnika o mierzone przez niego wartości. Odebrane dane są przekształcane na wartość ciśnienia w jednostce hPa. Odbywa się to za pomocą odpowiedniego wzoru uwzględniającego wcześniej odczytane współczynniki korekcyjne.

Aplikacja czujnika wymaga doprowadzenia do niego sygnału zegarowego o częstotliwości 32768 Hz i jest to linia niezależna od wyprowadzenia przeznaczonego dla sygnału zegarowego interfejsu komunikacyjnego SPI. Do generowania tego sygnału zastosowano układ SG-3032JC (rysunek 5).

Ostatnią z wartości przewidzianych do mierzenia przez urządzenie jest stężenie dwutlenku węgla. Do pomiaru tego parametru wykorzystano czujnik S-100 firmy ELT.



Rysunek 6. Schemat blokowy czujnika CO2 S-100

Czujnik ten w porównaniu z innymi spotykanymi na rynku produktami wyróżnia się długą żywotnością, krótkim czasem rozruchu, dobrą selektywnością mierzonego gazu i brakiem podatności na inne związki chemiczne. Zalety te są wynikiem wykorzystania do konstrukcji czujnika obecnie najbardziej zaawansowanej technologii pomiaru stężenia gazów zwanej w skrócie NDIR (*Nondispersive Infrared*), która korzysta ze zjawiska absorpcji przez gazy promieniowania z zakresu podczerwieni. Technika pomiarowa polega na emitowaniu przez źródło światła fal o długości 4,3 μm, które docierają do detektora przez tubę zawierającą powietrze z otoczenia zewnętrznego czujnika. W zależności od stężenia dwutlenku węgla w powietrzu proporcjonalna ilość wyemitowanych fal zostaje pochłonięta przez gaz. Pozostała ich część dociera do detektora. Cały proces analizy spektralnej powietrza i odczyt na jej podstawie stężenia dwutlenku węgla wymaga wyspecjalizowanej aparatury pomiarowej.

Wykaz elementów

Rezystory:

- R1: 10 kΩ (SMD, 0805)
- R2, R3, R4: 100 Ω (SMD, 0805)

Kondensatory:

- C1: 10 μF/20 V (tantalowy „C”)
- C2: 22 μF/16 V (tantalowy „C”)
- C3: 4.7 μF/10 V (tantalowy „A”)
- C4, C6...C15, C20: 1 μF (SMD, 0805)
- C5: 10 μF/10V (tantalowy „A”)
- C16, C18, C21: 100 nF (SMD, 0805)
- C17, C19: 10nF (SMD, 0805)
- C22, C23: 12pF (SMD, 0805)

Półprzewodniki:

- IC1: EFM32G210F128
- IC2: LM1117-5.0
- IC3: MCP1825-3.3
- IC4: EA DOGM132
- IC5: ELT S-100
- IC6: Sensirion SHT11
- IC7: Intersema MS5534A
- IC8: SG-3032JC

- LED1, LED2: dioda LED (0805)

- B1: mostek prostowniczy B805 (80 V, 1 A)

Inne:

- SV1: listwa kołkowa 1×3 (raster 2.54 mm)
- Q1: kwarc 8.00 MHz (SMD)
- INPUT CON: złącze zasilające

Na CD: karty katalogowe i noty aplikacyjne elementów oznaczonych w wykazie elementów kolorem czerwonym



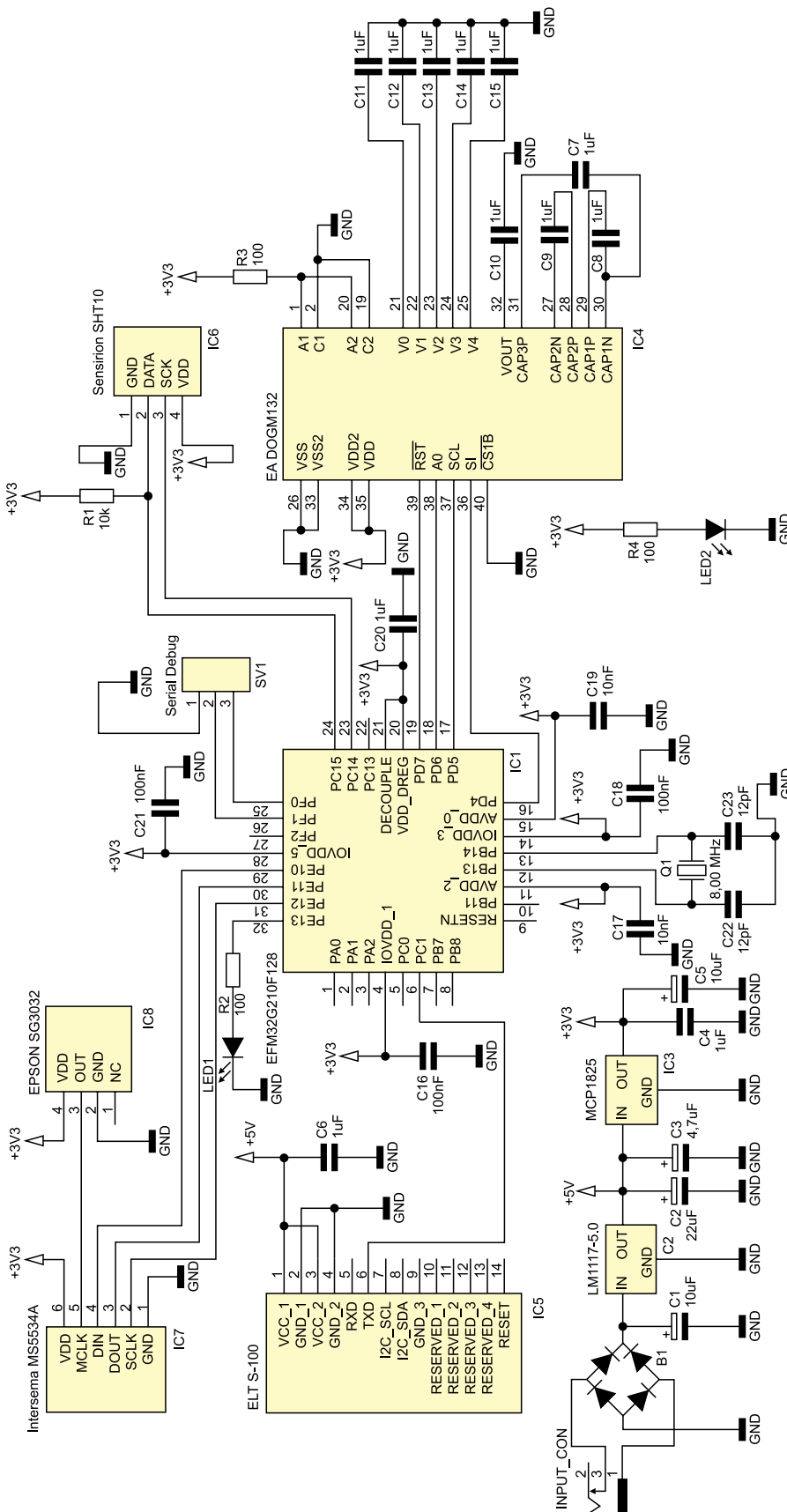
Z tego względu czujnik S-100 został wyposażony w służącą do tych zadań optykę, mechanikę i elektronikę (rysunek 6). Dzięki temu odczyt pomiarów z czujnika sprowadza się do odbioru co 3 sekundy za pomocą interfejsu UART sekwencji znaków ASCII, która

zawiera poziom dwutlenku węgla w powietrzu w jednostce PPM (*Parts Per Million*). Alternatywną metodą odczytu pomiarów jest interfejs zgodny z protokołem I<sup>2</sup>C.

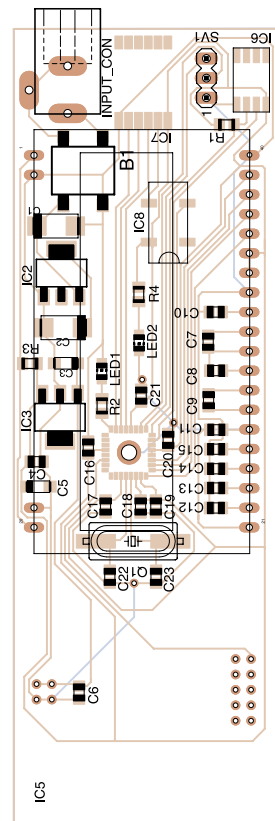
Kolejnym ważnym elementem zaprojektowanego urządzenia jest wyświetlacz

LCD, do którego funkcji należy wizualizacja pomiarów z czujników. Wybrano moduł EA DOGM132 firmy Electronic Assembly dobrze pasujący do założeń ze względu na mały pobór prądu, niewielkie wymiary i odpowiednią dla potrzeb rozdzielczość. Jego dodatkowymi atutami są łatwość montażu wynikająca z wyprowadzenia styków w formie przewlekanej.

Ostatnią, nieomówioną dotąd częścią składową urządzenia jest blok zasilania. Należy go zaprojektować wzięwszy pod uwagę wymagania prądowe i napięciowe zasilanych układów. Uwzględniając te warunki okazuje się, że prawie wszystkie zasilane elementy zostały dostosowane do napięcia zasilania 3,3 V, a ich sumaryczny pobór prądu to rząd pojedynczych miliamper. Wyjątkiem jest czujnik dwutlenku węgla, który wymaga



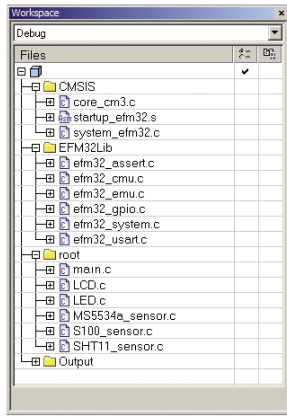
Rysunek 7. Schemat elektryczny miernika parametrów środowiskowych



Rysunek 8. Schemat montażowy miernika parametrów środowiskowych

REKLAMA





**Rysunek 9. Struktura plików programu dla urządzenia miernika (środowisko IAR EWARM 6.0)**

zasilania napięciem 5 V i pobiera 20 mA oraz 320 mA co 3 sekundy (przez 10 ms). Analizując wymagania zasilania zdecydowano się na zastosowanie dwóch regulatorów napięcia – układu National Semiconductor ze stabilizowanym wyjściem 5 V oraz Microchip MCP1825-3.3 ze stabilizowanym wyjściem 3,3 V. Stabilizatory te zapewniając wydajność prądową o wartości, odpowiednio, do 800 mA i 500 mA. Oba układy są regulatorami typu LDO.

**Budowa urządzenia**

Schemat ideowy miernika pokazano na **rysunku 9**. W jego centralnym miejscu umieszczono mikrokontroler EFM32G210F128 (IC1). Do każdego wyprowadzenia zasilania dołączono kondensator ceramiczny filtrujący zakłócenia oraz pełniący rolę źródła zasilania w krótkich, zgodnych z taktmem zegarowym momentach, gdy układ wymaga większej ilości energii. Są to kondensatory o pojemności 10 nF (C17, C19), 100 nF (C16, C18, C21) oraz 1 µF (C20), które dołączono do linii IOVDD, AVDD oraz DECOUPLE. Dodatkową linią wymagającą połączenia z napięciem zasilania jest wyprowadzenie VDD\_DREG.

Poprzez wyprowadzenia PB13 i PB14 do mikrokontrolera dołączono kwarc Q1, który pełni dla niego rolę źródła sygnału zegarowego. Układy z rodziny EFM32 mogą być taktowane z częstotliwością do 32 MHz. W wykonanym urządzeniu zastosowano kwarc 8 MHz. Obwód generatora sygnału zegarowego oprócz kwarcu składa się z dwóch kondensatorów 12 pF (C22, C23).

Do programowania mikrokontrolera wykorzystano interfejs SWD (Serial Wire Debug). Umożliwia on zarówno wgrywanie kodu, jak również jego pełne debugowanie (praca krokowa, podgląd wartości zmiennych, rejestrów itp.). Interfejs Serial Wire Debug zbudowany jest z zaledwie dwóch linii sygnałowych: SWDIO (dwukierunkowa linia danych) i SWCLK (linia zegarowa). W układzie EFM32G210F128 odpowiadają

im wyprowadzenia PF1 i PF0. Trzecim wymaganiem połączeniem jest masa. Linie interfejsu Serial Debug wyprowadzone zostały na złączu SV1.

Do układu EFM32G210F128 dołączono szereg komponentów peryferyjnych. Pierwszym z nich jest wyświetlacz LCD EA DOGM132. Jest on połączony z mikrokontrolerem za pomocą czterech linii sygnałowych SCL, SI, RST i A0 wyprowadzonymi PD5, PD4, PD7 i PD6. Pierwsze dwie linie są jednokierunkowym interfejsem komunikacyjnym kompatybilnym ze standardem SPI (jedna linia zegarowa i jedna linia danych do zapisu do wyświetlacza). Wyprowadzenie RST służy do resetowania wyświetlacza, natomiast linia A0 odpowiada za sposób interpretacji odbieranych przez wyświetlacz danych (instrukcje lub kody znaków do wyświetlenia). Do wyświetlacza podłączonych zostało osiem kondensatorów 1 µF (C8...C15). Ich obecność jest niezbędna dla umożliwienia wytworzenia przez wyświetlacz napięcia wyższego niż napięcie zasilania w celu uzyskania odpowiedniego kontrastu wyświetlanych znaków.

Kolejnym jest czujnik wilgotności i temperatury Sensirion SHT11. Komunikacja z czujnikiem odbywa się za pośrednictwem dwóch linii SCK i DATA, które tworzą szeregowy interfejs komunikacyjny. Linia danych wymaga dołączenia rezystora podciągającego do potencjału zasilania, czego efektem jest wysoki stan logiczny na linii danych w stanie spoczynkowym. Linie czujnika połączone są z portami mikrokontrolera PC14 i PC15.

Drugim elementem pomiarowym urządzenia jest czujnik ciśnienia atmosferycznego Intersema MS5534A. Jest on wyposażony w klasyczny interfejs SPI, którego linie SCLK, DOUT i DIN zostały połączone z wyprowadzeniami mikrokontrolera PE13, PE12 i PE11. Czujnik wymaga dodatkowo zewnętrznego sygnału zegarowego o częstotliwości 32.768 kHz, dlatego do tego celu wykorzystano generator EPSON SG-3032JC, którego wyprowadzenie z sygnałem wyjściowym zostało dołączone do odpowiedniego wejścia czujnika o nazwie MCLK.

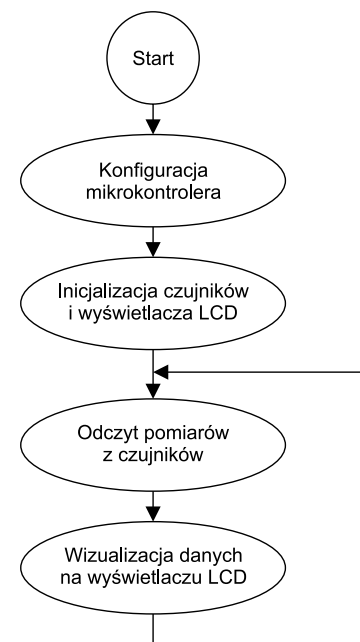
Kolejnym układem pomiarowym urządzenia jest czujnik dwutlenku węgla ELT S-100. Został on dołączony do mikrokontrolera interfejsem UART. Za realizację transmisji w tym standardzie odpowiedzialna jest linia RXD, która łączy się z układem EFM32G210F128 wyprowadzeniem PC1. Ze względu na impulsowy pobór prądu przez czujnik S-100 blisko jego wyprowadzenia zasilania umieszczono kondensator 1 µF (C6).

Ostatnim obwodem schematu jest blok zasilania, do którego napięcie doprowadzane jest poprzez gniazdo INPUT\_CON. Na wejściu bloku zasilania zastosowano mostek prostowniczy, który zapewnia poprawne

spolaryzowanie wejściowego napięcia zasilającego. Za mostkiem znajduje się regulator napięcia LM1117, który stabilizuje napięcie na poziomie 5,0 V. Jest ono niezbędne do zasilenia czujnika dwutlenku węgla. Na wejściu układu umieszczono kondensator tantalowy o wartości 10 µF (C1), a na wyjściu 22 µF (C2). Wszystkie pozostałe komponenty urządzenia wymagają napięcia zasilania o wartości 3,3 V. Do jego uzyskania służy drugi układ stabilizatora MCP1825.

W celach sygnalizacyjnych zastosowano dwie diody LED (LED1, LED2). Świecenie diody LED2 sygnalizuje obecność napięcia 3,3 V. Dioda LED1 jest dołączona do mikrokontrolera i można jej użyć w aplikacji.

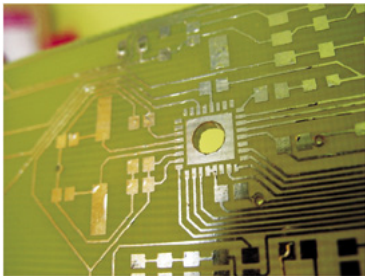
Schemat montażowy miernika pokazano na **rysunku 10**. W środkowej części płytki umieszczono mikrokontroler. W najbliższym otoczeniu mikrokontrolera znalazły się kondensatory ceramiczne 10 nF, 100 nF i 1 µF, których obecność jest rekomendowana przez producenta układu. W dalszej odległości znajduje się rezonator kwarcowy i diody LED. W górnej części płytki umiejscowiono obwód zasilania złożony z gniazda zasilania, mostka prostowniczego, dwóch regulatorów napięcia i towarzyszących im kondensatorów tantalowych. Ponad wymienionymi elementami umieszczony został wyświetlacz LCD, którego wyprowadzenia znajdują się w dolnej i górnej części płytki. Blisko dolnych wyprowadzeń umiejscowiono kondensatory wymagane do poprawnej pracy wyświetlacza. Po obu bokach płytki znajdują się czujniki – z lewej strony czujnik dwutlenku węgla, natomiast z prawej strony czujnik ciśnienia oraz wilgotności i temperatury. Między czujnikami po prawej stronie znajduje się 3-pinowa listwa do programowania i debugowania mikrokontrolera. Takie



**Rysunek 10. Schemat blokowy oprogramowania**



Rysunek 11. Dane na wyświetlaczu



Fotografia 12. Widok miejsca umieszczenia na płycie mikrokontrolera z otworem na wyprowadzenie masy

ułożenie elementów wydaje się optymalne, gdyż umożliwia osiągnięcie możliwie małych gabarytów urządzenia przy jednoczesnym wyeksponowaniu czujników na skraju płytki, co umożliwia swobodny dostęp powietrza do nich.

### Zasada działania urządzenia

Oprogramowanie do mikrokontrolera EFM32G210F128 zostało wykonane w środowisku programistycznym *IAR Embedded Workbench for ARM. 6.0 (Kickstart Edition)*. Kod źródłowy napisano w języku C na pod-

stawie przeznaczonego do obsługi peryferi mikrokontrolera zestawu bibliotek producenta o nazwie *EFM32 Library* oraz bibliotek *CMSIS*, które tworzą uniwersalny i niezależny od platformy sprzętowej interfejs programistyczny dla rdzeni ARM Cortex-M. Dodatkowo, utworzono biblioteki dla wykorzystanych układów: czujników (pliki *MS5534A\_sensor.c*, *MS5534A\_sensor.h*, *S100\_sensor.c*, *S100\_sensor.h*, *SHT11\_sensor.c*, *SHT11\_sensor.h*), diod LED (*LED.c*, *LED.h*) oraz wyświetlacza LCD (*LCD.c*, *LCD.h*). Strukturę plików programu pokazano na **rysunku 11**. Ogólny algorytm działania urządzenia przedstawiono w postaci schematu blokowego na **rysunku 12**, a główną pętlę programu na **listingu 1**.

Po włączeniu urządzenia mikrokontroler przystępuje do czynności konfiguracyjnych. Na początku wykonana zostaje funkcja *CHIP\_Init()*, której wywołanie jako pierwszej w programie sugeruje producent. Funkcja ta odpowiada za ustawienie wartości rejestrów domyślnymi, fabrycznymi wartościami. Następnie mikrokontroler dokonuje ustawień przygotowujących układ do pracy

z czujnikami i diodą LED. Proces ten odbywa się w funkcjach *LED\_Init()*, *SHT11\_Init()*, *MS5534a\_Init()*, *S100\_Init()*. Na tym etapie konfigurowane są porty, piny, peryferia (SPI, UART) i ich zegary. Kolejną czynnością jest odczytanie z czujnika ciśnienia współczynników kalibracyjnych, które są indywidualne dla każdego czujnika danego modelu i są wymagane w procesie przetwarzania odczytanego z czujnika zmierzonego pomiaru na wartość ciśnienia. Odczyt współczynników realizuje funkcja *MS5534a\_Read\_Calibration\_Coefficients()*. W kolejnym kroku zostaje zainicjowany wyświetlacz LCD. Mikrokontroler przesyła do niego sekwencję danych ustawiających poziom kontrastu i orientację tekstu, a następnie włącza opcję pokazywania tekstu i przesyła dane do wyświetlenia. Na górnych dwóch liniach wyświetlacza wyświetlona zostaje nazwa urządzenia, natomiast na dwóch dolnych pokazane zostają pierwsze litery nazw mierzonych wartości środowiskowych. Ta czynność kończy etap przygotowania urządzenia do pracy. Od tego momentu rozpoczyna się cykliczne odpytywanie czujników o mierzone przez nie wartości. Odczyt z czujników realizują funkcje *S100\_Read\_CO2()*, *SHT11\_Read\_Temperature()*, *SHT11\_Read\_Humidity()*, *MS5534a\_Read\_Temperature()*, *MS5534a\_Read\_Pressure()*. Wykonywanie kolejnych serii pomiarowych sygnalizowane jest rytmicznym zapalaniem i gaśnięciem diody LED (funkcje *LED\_On()* i *LED\_Off()*). Funkcje odczytujące dane z czujników po przetworzeniu informacji zwracają wartości pomiarów. Są one zapisywane w zmiennych *Carbon\_dioxide*, *Temperature*, *Humidity*, *Pressure*. Następnie wartości te są wizualizowane w przeznaczonych im miejscach na wyświetlaczu. Odbywa się to za pomocą funkcji *Put\_On\_LCD\_Sensor\_Values()*. Pomiar są wyświetlane na dwóch dolnych liniach wyświetlacza LCD. Na wyższej z nich po lewej stronie znajduje się pole temperatury. Rozpoczynają je symbolizujące ten parametr znaki „T:”, po których znajduje się liczba będąca wartością temperatury z dokładnością do setnych części stopni Celsjusza. Dalej na prawo umieszczone są symbole wilgotności

### Listing 1. Program główny miernika parametrów środowiskowych

```
#include <stdint.h>
#include <stdbool.h>
#include "efm32.h"
#include "efm32_chip.h"
#include "efm32_emu.h"
#include "efm32_cm_u.h"
#include "efm32_gpio.h"
#include "efm32_usart.h"
#include "LCD.h"
#include "SHT11_sensor.h"
#include "MS5534a_sensor.h"
#include "LED.h"
#include "S100_sensor.h"

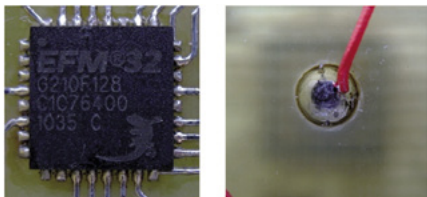
float Temperature, Humidity;
unsigned int Pressure, Carbon_dioxide;

int main(void)
{
    CHIP_Init();

    LED_Init();
    SHT11_Init();
    MS5534a_Init();
    S100_Init();
    MS5534a_Read_Calibration_Coefficients();
    LCD_Initialize();

    while(1)
    {
        LED_On();
        Carbon_dioxide = S100_Read_CO2();
        Temperature = SHT11_Read_Temperature();
        Humidity = SHT11_Read_Humidity();
        MS5534a_Read_Temperature();
        Pressure = MS5534a_Read_Pressure();
        LED_Off();
        Put_On_LCD_Sensor_Values(Temperature, Humidity, Pressure, Carbon_dioxide);
    }
}
```

REKLAMA



**Fotografia 13. Sposób montażu mikrokontrolera (po lewej stronie górna strona płytki, po prawej dolna strona płytki)**

(„W:”), za którymi z tą samą dokładnością wyświetlana jest wartość wilgotności, której jednostką są procenty. W analogiczny sposób

przedstawione są pomiary w najniższej linii wyświetlacza. Z lewej strony wyświetlona jest wartość ciśnienia (jednostka: hektopaskale) poprzedzona znakami „C:”, natomiast z prawej strony widnieją symbole „D:” i wyświetlana jest wartość stężenia dwutlenku węgla w powietrzu (jednostka PPM). Widok wyświetlacza LCD z wykonanymi pomiarami przedstawiono na **fotografii 13**.

Kod realizujący funkcjonalność urządzenia został w całości napisany przy użyciu wysokopoziomowych bibliotek przygotowanych przez firmę Energy Micro, dlatego warto przedstawić przykład ich wykorzystania. Na **listin-**

**gu 2** pokazano utworzoną na potrzeby urządzenia bibliotekę LED.c do sterowania diodą LED, która dołączona jest do wyprowadzenia 13 portu E. Biblioteka ta korzysta z funkcji znajdujących się w pliku efm32\_gpio.c. Jest to prosty przykład wykorzystania wyprowadzenia mikrokontrolera jako linii ogólnego przeznaczenia do ustalania poziomu logicznego.

Jak widać na zamieszczonym przykładzie czynność ta rozpoczyna się od włączenia sygnału zegarowego dla portów mikrokontrolera. Proces ten wykonywany jest za pomocą funkcji `CMU_ClockEnable(cmuClock_GPIO, true)`. Następnie dokonywany jest wybór konkretnego portu oraz pinu, dla których wybierany jest tryb działania. Realizuje to funkcja `GPIO_PinModeSet(gpioPortE, 13, gpioModePushPull, 0)`. Po wywołaniu tych dwóch funkcji można przystąpić do ustalania poziomu logicznego „1” lub „0”, za co odpowiadają funkcje `GPIO_PinOutSet(gpioPortE, 13)` i `GPIO_PinOutSet(gpioPortE, 13)`.

Kolejnym przykładem na użycie bibliotek producenta może być wykorzystanie interfejsu komunikacyjnego UART do odczytu danych z czujnika dwutlenku węgla ELT S-100. Kod źródłowy przedstawiono na **listingu 3**.

Podobnie jak dla pierwszego przykładu konfigurację należy rozpocząć od włączenia sygnału zegarowego dla UART oraz ustawienia odpowiedniego trybu działania wyprowadzeń mikrokontrolera (wyprowadzenie 1, port C). Zadania te realizują odpowiednio funkcja `CMU_ClockEnable(cmuClock_USART1, true)` i `GPIO_PinModeSet(gpioPortC, 1, gpioModeInput, 0)`. W celu ustawienia parametrów transmisji należy zdefiniować dwie struktury języka C o następujących typach: `USART_TypeDef`, `USART_InitAsync_TypeDef`. Po wypełnieniu pól struktur (`ROUTE`, `enable`, `refFreq`, `baudrate`, `oversampling`, `databits`, `parity`, `stopbits`) jest wykonywana funkcja `USART_InitAsync(usart, &init)`, której argumentami są stworzone struktury. Dzięki temu aktualna konfiguracja trafia do odpowiednich rejestrów mikrokontrolera. Odbiór danych odbywa się za pomocą funkcji `USART_Rx(USART1)`, która każdorazowo odbiera 1 bajt danych i zwraca go do przypisanej jej zmiennej.

## Montaż

Montaż należy bezwzględnie rozpocząć od przylutowania mikrokontrolera. Wybrany model dostarczany jest przez producenta w obudowie QFN, a więc wyprowadzenia układu są „ścięte” równo z krawędzią obudowy, przez co niezwykle ważne jest ich bardzo precyzyjne umieszczenie dokładnie na przewidzianych im padach płytki. Miejsca styku wyprowadzeń i padów znajdują się pod obudową układu, dlatego proces lutowania najprościej wykonać poprzez naniesienie pasty lutowniczej na powierzchnię płytki i mikrokontrolera, a następnie użycie lutownicy typu hot-air.

### Listing 2. Biblioteka do sterowania diodą LED (plik LED.c) dołączoną do wyprowadzenia 13 portu E

```
#include "efm32.h"
#include "efm32_cmu.h"
#include "efm32_gpio.h"
#include "LED.h"

void LED_Init(void)
{
    CMU_ClockEnable(cmuClock_GPIO, true);
    GPIO_PinModeSet(gpioPortE, 13, gpioModePushPull, 0);
}

void LED_On(void)
{
    GPIO_PinOutSet(gpioPortE, 13);
}

void LED_Off(void)
{
    GPIO_PinOutClear(gpioPortE, 13);
}
```

### Listing 3. Odczyt danych z czujnika CO2

```
#include "efm32.h"
#include "efm32_cmu.h"
#include "efm32_gpio.h"
#include "efm32_usart.h"
#include "S100.h"

unsigned int CO2 = 0;
unsigned char CO2_table[20] = {0};

void S100_Init(void)
{
    CMU_ClockEnable(cmuClock_USART1, true);

    GPIO_PinModeSet(gpioPortC, 1, gpioModeInput, 0);

    USART_TypeDef *usart = USART1;
    USART_InitAsync_TypeDef init;

    usart->ROUTE = USART_ROUTE_RXPEN;

    init.enable = usartEnable;
    init.refFreq = 0;
    init.baudrate = 38400;
    init.oversampling = usartOVS16;
    init.databits = usartDatabits8;
    init.parity = usartNoParity;
    init.stopbits = usartStopbits1;

    USART_InitAsync(usart, &init);
}

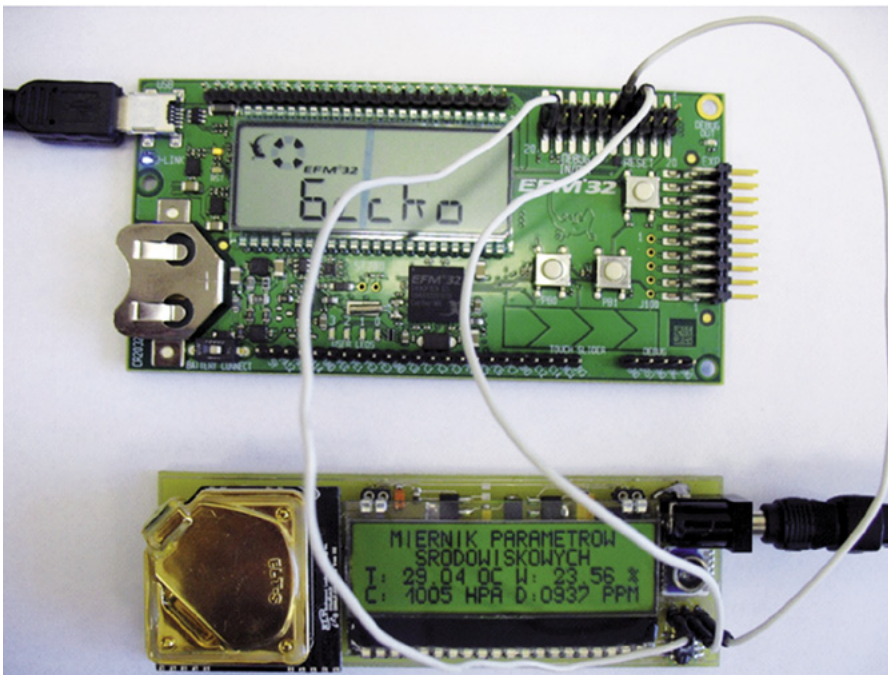
unsigned int S100_Read_CO2(void)
{
    unsigned char iteration = 0;
    do
    {
        CO2_table[iteration] = USART_Rx(USART1);
        iteration++;
    }
    while (CO2_table[iteration - 1] != 10);

    CO2_table[iteration-4] = CO2_table[iteration-4]-48;
    CO2_table[iteration-5] = CO2_table[iteration-5]-48;
    CO2_table[iteration-6] = CO2_table[iteration-6]-48;
    CO2_table[iteration-7] = CO2_table[iteration-7]-48;

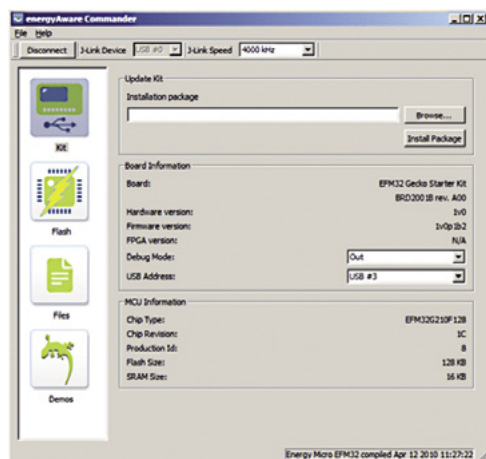
    CO2 = 1000*CO2_table[iteration-7] + 100*CO2_table[iteration-6] + 10*CO2_
table[iteration-5] + CO2_table[iteration-4];

    iteration = 0;
    return CO2;
}
```





Rysunek 14. Podłączenie zestawu EFM32 Gecko Starter Kit w konfiguracji programatora zewnętrznego



Rysunek 15. Sposób wykorzystania zintegrowanego w zestawie EFM32 Gecko Starter Kit programatora J-Link do programowania i debugowania układów zewnętrznych

Pewną trudnością może być połączenie masy układu ze względu na fakt, iż jest ona umiejscowiona centralnie pod układem. Może sobie poradzić wierząc otwór w płytce pośrodku miejsca przewidzianego na mikrokontroler (fotografia 14) i wykorzystanie wykonanego otworu do dogodnego wyprowadzenia masy przez wlutowanie w otwór przewodu masowego od spodu płytki (fotografia 15).

Po przylutowaniu mikrokontrolera można przystąpić do montażu kolejnych podzespołów. Należy zacząć od najmniejszych komponentów, a więc elementów SMD – kondensatorów ceramicznych i tantalowych, diod LED oraz rezystorów. Wykorzystując wyświetlacz w konfiguracji bez podświetlenia należy pominąć montaż rezystora R3. Następnie można przystąpić do przytwierdzenia większych elementów powierzchniowych – stabilizatorów napięcia, mostka prostowniczego, kwarcu, generatora zegarowego oraz czujników ciśnienia,

wilgotności i temperatury. Ostatnim etapem jest montaż elementów przewlekanych – gniazda zasilającego, gniazda do programowania, czujnika dwutlenku węgla, złącza do wyświetlacza oraz samego wyświetlacza.

Poprawność montażu można sprawdzić dołączając napięcie zasilania, czego efektem powinno być zaświecenie się diody LED2.

Jako programator i debugger dla wykonanego urządzenia wykorzystano zestaw EFM32 Gecko Starter Kit. Zestaw ten jest wyposażony w zintegrowany na płytce programator Segger J-Link, który może służyć zarówno do programowania układu EFM32G890F128 znajdującego się na płytce zestawu, jak również układów z rodziny EFM32 w zewnętrznych aplikacjach. Sposób połączenia wykonanego urządzenia z zestawem EFM32 Gecko Starter Kit w konfiguracji programatora z interfejsem SWD przedstawiono na fotografii 16. Po zestawieniu połączenia sprzętowego można przystąpić do kon-

figuracji oprogramowania. Firma Energy Micro w tym celu przygotowała specjalną aplikację o nazwie *energyAware Commander*. Za jej pomocą można wybrać, czy program zostanie wgrany do mikrokontrolera zestawu EFM32 Gecko Starter Kit, czy do aplikacji zewnętrznej. Konfigurację do programowania układu zewnętrznego ustawia się poprzez zmianę w zakładce *Kit* opcji *Debug Mode* na wartość *Out*. Po ustawieniu tej opcji wykryty zostanie nasz układ zewnętrzny – EFM32G210F128 (rysunek 17). Następnie można przejść do zakładki *Flash*, gdzie wybrany zostaje plik wyników (wsad do procesora), którym programowany jest mikrokontroler. Po wciśnięciu przycisku *Flash EFM32* pojawi się pasek stanu procesu programowania. Po jego całkowitym wypełnieniu czynność programowania jest zakończona, a urządzenie rozpoczyna pracę.

## Podsumowanie

W artykule przedstawiono autorski projekt urządzenia pomiarowego wybranych parametrów środowiskowych: temperatury otoczenia, wilgotności, ciśnienia atmosferycznego oraz stężenia dwutlenku węgla. Po wykonanych testach użytkowania urządzenia można stwierdzić, że działa ono poprawnie i prawidłowo spełnia przewidziane mu funkcje. Uzyskana funkcjonalność urządzenia sprawia, iż można je wykorzystać w różnych miejscach wymagających monitorowania stanu mierzonych parametrów środowiskowych. Dodatkowo, istnieje możliwość dalszego rozwoju urządzenia, co zwiększy zakres jego potencjalnych zastosowań. Po dokonaniu niewielkich zmian w oprogramowaniu i konstrukcji fizycznej urządzenia może ono pełnić funkcję np. sterownika w systemie klimatyzacji dokonując nie tylko pomiarów parametrów otoczenia, ale również sterując urządzeniami odpowiedzialnymi za ich zmianę.

Szymon Panecki, EP

## Bibliografia:

- [www.co2sensor.co.kr](http://www.co2sensor.co.kr) S-100 datasheet
- [www.energymicro.com](http://www.energymicro.com), EFM32 hardware design considerations rev. 1.32, listopad 2010
- [www.energymicro.com](http://www.energymicro.com), EFM32G reference manual rev. 1.10, kwiecień 2011
- [www.energymicro.com](http://www.energymicro.com), EFM32G210 datasheet rev. 1.2, grudzień 2010
- [www.epsontotocom.co.jp/english](http://www.epsontotocom.co.jp/english) SG-3032 crystal oscillator datasheet
- [www.intersema.com](http://www.intersema.com), MS5534A datasheet, czerwiec 2002
- [www.lcd-module.de](http://www.lcd-module.de), EA DOGM132-5 series datasheet, 2009
- [www.microchip.com](http://www.microchip.com), MCP1825/MCP1825S datasheet, 2008
- [www.national.com](http://www.national.com), LM1117/LM1117I datasheet, kwiecień 2006
- [www.sensirion.com](http://www.sensirion.com), SHT1x datasheet v.4.3, maj 2010