

Obsługa scalonego barometru MPL115A2 w języku BASCOM

Współczesne układy scalone to już nie tylko struktury krzemowe, w których zaimplementowano aktywne i pasywne elementy elektroniczne. Z użyciem technologii MEMS są tworzone miniaturowe elementy mechaniczne, dzięki którym stała się możliwa budowa scalonych przetworników wielkości nieelektrycznych. Szybki rozwój tej technologii zrewolucjonizował sposób pomiaru wielu nieelektrycznych wielkości fizycznych, co umożliwiło implementację tego typu pomiarów na platformach sprzętowych, dla których były one dotychczas niedostępne. Dobrym przykładem tego postępu jest scalony, cyfrowy układ do pomiaru bezwzględnego ciśnienia atmosferycznego typu MPL115A2 produkowany przez firmę Freescale, który jest kompletnym systemem pomiarowym pracującym z zastosowaniem technologii MEMS.

Z definicji, ciśnienie atmosferyczne jest stosunkiem wartości siły, z którą naciska słup powietrza atmosferycznego do pola powierzchni, na jaką ten słup naciska. Wynika z tego, że dla przykładu ciśnienie atmosferyczne w górach jest niższe a na nizinach wyższe, ponieważ słup powietrza ma w tych rejonach różne wysokości. Zależność ta ma w przybliżeniu charakter wykładniczy, co pokazano na **rysunku 1**.

Wartość ciśnienia atmosferycznego dla standardowych warunków pogodowych dla wysokości h (n.p.m.) możemy, zatem wyznaczyć z uproszczonej zależności opisanej wzorem:

$$P_h = P_0 \cdot e^{\frac{-h}{7990}}$$

gdzie:

P_h – ciśnienie na wysokości h n.p.m. dla standardowych warunków pogodowych

P_0 – ciśnienie na poziomie morza równe 1013,25 hPa

h – wysokość n.p.m.

Ciśnienie atmosferyczne ulega zmianom, które to są zależne od zmiany warunków pogodowych w związku z czym na podstawie zmian ciśnienia na pewnej wysokości n.p.m. możemy z dużym prawdopodobieństwem określić tendencję zmian pogody. Dla przykładu, w umiarkowanych szerokościach geograficznych powolne obniżanie się ciśnienia oznacza zbliżanie się niżu, czyli pogorszenia pogody (wystąpienie chmur, opadów, w lecie – ochłodzenia, w zimie – ocieplenia), zaś wzrost ciśnienia łączy się z poprawą pogody (ustą-

pieniem mgły/zachmurzenia, osłabieniem wiatru itp.). Pomimo tych, wydawałoby się prostych zależności, przewidywanie zmian pogody jest procesem bardzo skomplikowanym (zwłaszcza przygotowywanie prognoz długookresowych) i obciążonym pewnym błędem, jednak dla prostych aplikacji możemy posilkować się algorytmami, z których korzysta większość elektronicznych stacji pogodowych, opartych o śledzenie zmian ciśnienia atmosferycznego. Warto podkreślić istotną różnicę pomiędzy wartością bezwzględnego ciśnienia atmosferycznego dla danej miejscowości (ciśnienia tam panującego) a wartością ciśnienia podawanego w prognozach pogody. To ostatnie jest ciśnieniem, które wystąpiłoby danego dnia w danej miejscowości, gdyby znajdowała się ona na poziomie morza, czyli de facto jest to bezwzględne ciśnienie atmosferyczne dla tego rejonu przeliczone dla poziomu morza. Tego typu konwersja ułatwia zorientowanie się, co do warunków pogodowych dla różnych wysokości n.p.m.

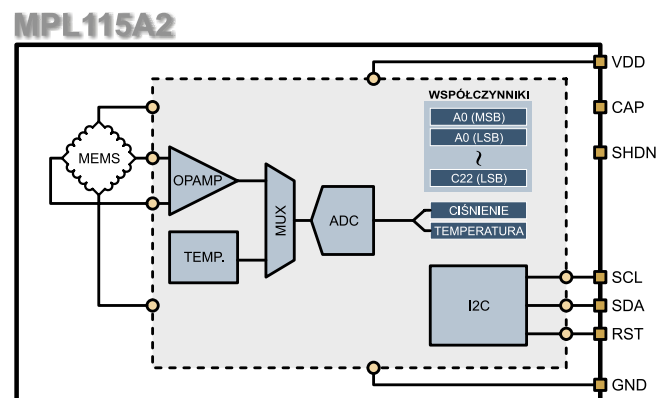
Obsługa czujnika

Po krótkim wstępie przejdźmy do czujnika MPL115A2. Jak wspomniano wcześniej, jest on scalonym, zaawansowanym systemem pomiaru bezwzględnego ciśnienia atmosferycznego charakteryzującym się następującymi, wybranymi cechami funkcjonalnymi:

- zintegrowany, 10-bitowy czujnik ciśnienia i temperatury,
- kompletny zestaw współczynników korekcyjnych zapisany w nieulotnej pamięci EEPROM czujnika a służący wyznaczeniu skom-



Rysunek 1. Wykres zależności bezwzględnego ciśnienia powietrza od wysokości n.p.m.



Rysunek 2. Uproszczony, blokowy schemat funkcjonalny czujnika MPL115A2

pensowanego temperaturowo bezwzględnego ciśnienia atmosferycznego,

- kalibracja na etapie produkcji,
- zakres pomiarowy 500...1150 hPa,
- dokładność 10 hPa,
- szeroki zakres napięć zasilających 2.375...5.5V,
- zintegrowany interfejs komunikacyjny I2C (SPI dla wersji A1),
- małogabarytowa obudowa LGA.

Uproszczony, blokowy schemat funkcjonalny czujnika MPL115A2 pokazano na **rysunku 2**, natomiast rozkład i opis wyprowadzeń zamieszczono w **tabeli 1**.

Układ MPL115A2 wyposażono w interfejs I²C. Przykładowy schemat nieskomplikowanego systemu zbudowanego z użyciem mikrokontrolera ATmega8 pokazano na **rysunku 3**.

Wykorzystano sprzętowy interfejs I²C wbudowany w strukturę mikrokontrolera. W przypadku mikroprocesorów bez wbudowanego interfejsu I²C można go zaimplementować programowo. Język Bascom umożliwia łatwą obsługę zarówno sprzętowego, jak i emulowanego interfejsu I²C. My jednak skupimy się na sprzętowej realizacji obsługi tego protokołu i objaśnieniu metody przeliczenia danych zwracanych przez czujnik.

Pomiar ciśnienia atmosferycznego przy wykorzystaniu układu MPL115A2 polega w tym przypadku na zainicjowaniu procesu pomiarowego (zarówno ciśnienia jak i temperatury) za pomocą specjalnych komend wysłanych do czujnika przy użyciu magistrali I²C (w naszym przypadku interfejsu TWI mikrokontrolera ATmega8), a następnie na odczycie zmierzonych, „surowych”, 10-bitowych wartości (z wewnętrzznego przetwornika A/D czujnika). Na podstawie odczytanych wartości zmierzonych ciśnienia, temperatury (niezbędnej dla celów kompensacji) oraz właściwych konkretnemu czujnikowi współczynników korekcyjnych, jesteśmy w stanie wyznaczyć bezwzględne ciśnienie atmosferyczne. Do tego celu musimy jednak posiłkować się zależnością dostarczoną przez producenta układu a wynikającą z charakterystyki pomiarowej jak i zastosowanego sposobu pomiaru, którą przedstawia poniższy wzór:

$$P_{comp} = A0 + (B1 + C11 \cdot P_{adc} + C12 \cdot T_{adc}) \cdot P_{adc} + (B2 + C22 \cdot T_{adc}) \cdot T_{adc}$$

gdzie:

P_{comp} – skompensowane, **bezwzględne** ciśnienie atmosferyczne,

P_{adc} – 10-bitowa wartość ciśnienia odczytana z rejestru ciśnienia czujnika MPL115A2,

T_{adc} – 10-bitowa wartość temperatury odczytana z rejestru temperatury czujnika MPL115A2,

A0 – współczynnik korekcyjny offsetu członu pomiarowego czujnika ciśnienia,

B1 – współczynnik korekcyjny czułości członu pomiarowego czujnika ciśnienia,

C11 – współczynnik korekcyjny nieliniowości członu pomiarowego czujnika ciśnienia (drugiego rzędu),

C12 – współczynnik korekcyjny czułości członu pomiarowego czujnika temperatury,

B2 – współczynnik korekcyjny offsetu członu pomiarowego czujnika temperatury (pierwszego rzędu),

C22 – współczynnik korekcyjny offsetu członu pomiarowego czujnika temperatury (drugiego rzędu).

Co ważne, wartości wszystkich, wyszczególnionych powyżej współczynników korekcyjnych charakterystyczne są dla każdego egzempla-

PIN	Symbol	Opis
1	VDD	Napięcie zasilania (2.375 ÷ 5.5V)
2	CAP	Kondensator filtrujący wewnętrznego regulatora napięcia
3	GND	Masa zasilania
4	SHDN	Wejście aktywujące tryb uśpienia (gdy połączone z GND), w którym to czujnik pobiera prąd rzędu 1 µA
5	RST	Wejście dezaktywacji interfejsu I ² C (gdy połączone z GND)
6	NC	Nie używane
7	SDA	Wejście/wyjście danych interfejsu I ² C
8	SCL	Sygnal zegarowy interfejsu I ² C

rza czujnika MPL115A2 i zapisywane są w nieulotnej pamięci czujnika przez producenta w czasie kalibracji układu. Odczyt tych wartości jest możliwy za pomocą interfejsu I²C i odpowiednim rozkazom sterującym. Sekwencje sygnałów magistrali I²C dla procedur inicjacji procesu pomiarowego, odczytu wartości ciśnienia oraz temperatury i współczynników korekcyjnych pokazano na **rysunkach 4... 6**.

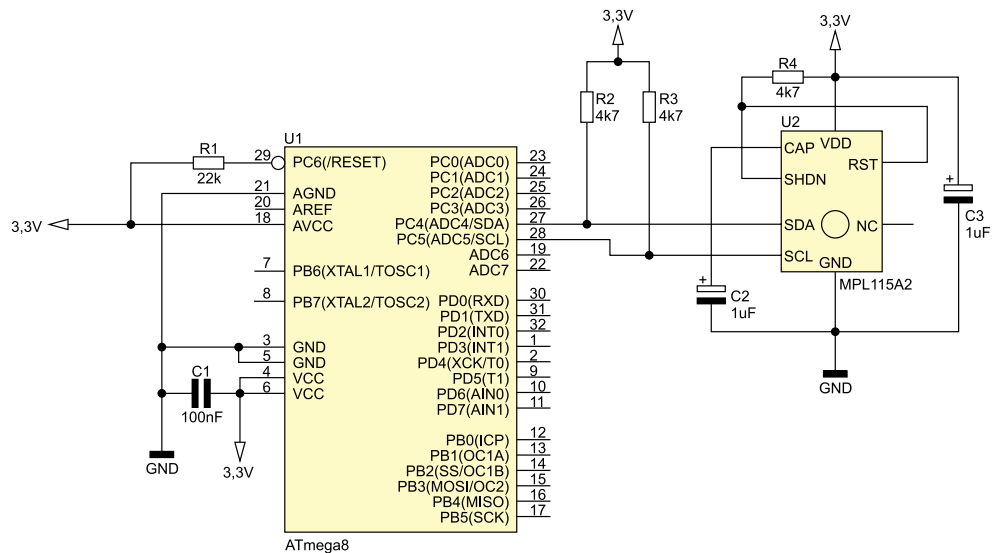
Aby jednak rozpocząć pracę z czujnikiem MPL115A2 należy uruchomić magistralę I²C, co w pierwszym kroku sprowadza się w zasadzie do określenia częstotliwości jej pracy za pomocą następującego wpisu do rejestru konfiguracyjnego prędkości transmisji:

```
TWBR = 32 'Uruchomienie i konfiguracja interfejsu TWI. Częstotliwość magistrali ustawiamy na 200kHz (dla fosc=8MHz)
```

Warto także od razu przygotować sobie procedury umożliwiające obustronną komunikację, których deklaracje w naszym przypadku wyglądają następująco:

```
Declare Sub Twi_start 'Procedura ta powoduje wygenerowanie sygnału Start na magistrali I2C
Declare Sub Twi_write_byte(byval Byte_to_send As Byte) 'Procedura wysyła 1 bajt danych
Declare Sub Twi_read_word(word_to_read As Word , Byval Acknowledgement As Byte) 'Procedura ta odczytuje 2 kolejne bajty danych przesłane magistralą I2C a wynik umieszcza w zmiennej word_to_read. Dodatkowo umożliwia określenie czy odbiornik ma wystawiać sygnał potwierdzenia czy też nie, czyli określa czy oczekiwane są kolejne dane
```

Ponadto, dla potrzeb procedury Twi_read_word i dla zwiększenia czytelności w programie obsługi zdefiniowano ponadto dwie stałe określające potrzebę wystawienia sygnału potwierdzenia po stronie odbiornika:



Rysunek 3. Sposób dołączenia układu MPL115A2 do systemu mikroprocesorowego

```
Const Ack_ = 1
Const Nack_ = 0
```

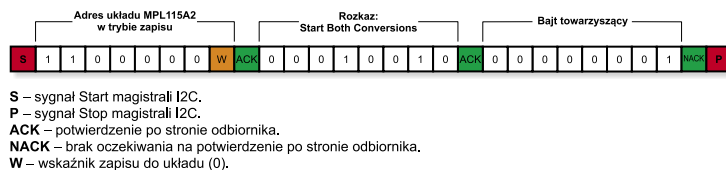
Przykłady procedur wykorzystujących opisane wyżej nastawy zamieszczono na listingach 1...3.

Dysponując podstawowymi funkcjami umożliwiającymi komunikację z układem MPL115A2 za pośrednictwem interfejsu TWI mikrokontrolera ATmega8 możemy przejść do zagadnień szczegółowych. W tabeli 2 zestawiono listę dostępnych rozkazów sterujących dla układu MPL115A2 wraz z opisem ich znaczenia, natomiast w tabeli 3 zamieszczono dokładną specyfikację wspomnianych współczynników korekcyjnych, jeśli chodzi o sposób ich zapisu. Jako uzupełnienie informacji o współczynnikach korekcyjnych w tabeli 4 przedstawiono mapę pamięci układu MPL115A2 w tym zakresie.

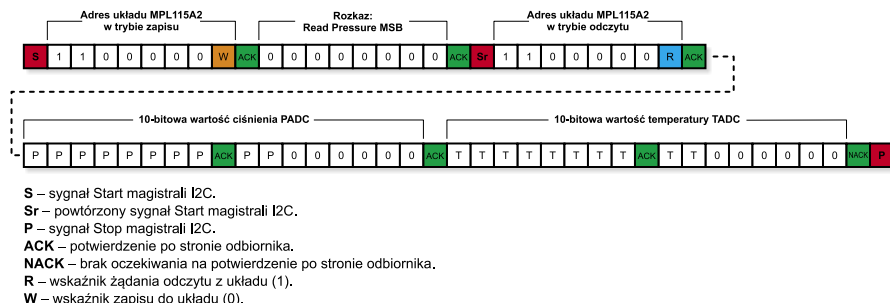
Jak wspomniano wcześniej, obliczenie końcowej, skompensowanej wartości bezwzględnego ciśnienia atmosferycznego wymaga znajomości wartości wielu współczynników korekcyjnych charakterystycznych dla każdego egzemplarza czujnika a zapisanych w jego pamięci w związku z czym potrzebna jest stosowna procedura, która wykona to zadanie. Odpowiednią procedurę pokazano na listingu 4.

Jak widać, ciało procedury Read_and_calculate_coefficients korzysta z innej procedury Extract_coefficients, która to służy obliczeniu rzeczywistej i ułamkowej części współczynników korekcyjnych z surowych wartości odczytanych z pamięci układu mając na uwadze ich specyfikację (dokładność części całkowitej i ułamkowej). Wygląd procedury pokazano na listingu 5 (aby zapewnić wysoką czytelność poszczególnych kroków procedura ta nie została zoptymalizowana).

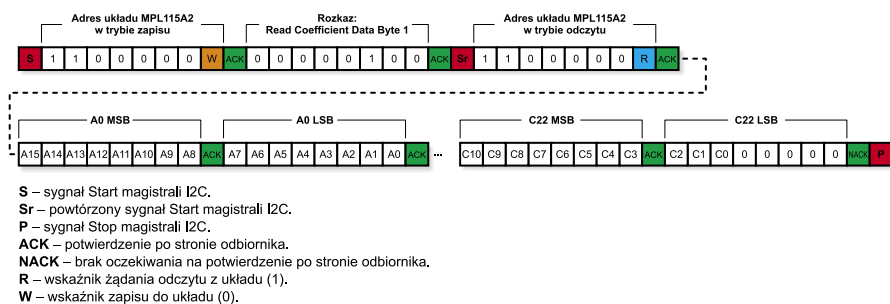
Mając obliczone niezbędne współczynniki korekcyjne możemy przejść do obliczenia mierzonego ciśnienia atmosferycznego. Zanim jednak tego dokonamy niezbędne jest odczytanie „surowych” wartości pomiaro-



Rysunek 4. Sekwencja sygnałów magistrali I²C dla procedury inicjalizacji procesu pomiarowego czujnika MPL115A2



Rysunek 5. Sekwencja sygnałów magistrali I²C dla procedury odczytu wartości ciśnienia P_{ADC} i temperatury T_{ADC} czujnika MPL115A2



Rysunek 6. Sekwencja sygnałów magistrali I²C dla procedury odczytu współczynników korekcyjnych czujnika MPL115A2

wych za co odpowiada procedura przedstawiona na listingu 6.

Mając odczytane surowe dane pomiarowe możemy przystąpić do wyznaczenia końcowego, skompensowanego ciśnienia atmosferycznego. Niestety, wymaga to szeregu działań z użyciem obliczeń zmiennoprzecinkowych co zajęłoby sporo zasobów pamięci Flash. Zadanie to można jednak z powodzeniem znacznie uprościć! Po pierwsze,

można, nie popełniając wielkiego błędu, pominąć we wzorze na ciśnienie wartości współczynników C11 i C22 nadając im wartości zerowe (co z resztą zastrzeżają producenci układów). Po drugie, tak otrzymany wzór można rozbić na część całkowitą i ułamkową wykonując jedynie działania na liczbach całkowitych. W ten sposób nieznacznie zmniejszymy (jeśli w ogóle) dokładność pomiaru,

Listing 1. Wysłanie sekwencji Start I²C

```
Sub Twi_start
  Twcr = &B10100100 'Wysyłamy sygnał Start: ustawiony TWINT, TWSTA, TWEN
  Bitwait Twcr.twint , Set 'Czekamy, aż interfejs TWI wykona Start i zgłosi ten fakt ustawiając flagę TWINT
End Sub
```

Listing 2. Wysłanie bajtu danych za pomocą I²C

```
Sub Twi_write_byte(byval Byte to_send As Byte)
  Twdr = Byte to_send 'Do rejestru danych interfejsu TWI wczytujemy bajt przeznaczony do wysłania
  Twcr = &B10000100 'Inicjujemy proces wysłania bajta danych: Ustawiony TWINT, skasowany TWSTA i ustawiony TWEN
  Bitwait Twcr.twint , Set 'Czekamy, aż interfejs TWI dokona transmisji i zgłosi ten fakt ustawiając flagę TWINT
End Sub
```

Listing 3. Odczyt bajtu danych za pomocą I²C

```
Sub Twi_read_word(word_to_read As Word , Byval Acknowledgement As Byte)
  Local Half As Byte
  Twcr = &B11000100 'Inicjujemy odczyt pierwszej, starszej połówki słowa Word
  Bitwait Twcr.twint , Set 'Czekamy na otrzymanie bajta danych sygnalizowane ustawieniem flagi TWINT
  Word to_read = Twdr
  Shift Word to_read , Left , 8 'Przesuwamy pierwszy bajt na pozycję MSB słowa Word
  'W zależności czy nasz odbiornik (Master Receiver) ma potwierdzać odczyt bajta danych ustawiamy odpowiednią wartość
  rejestru kontroli interfejsu TWI
  If Acknowledgement = Ack Then Twcr = &B11000100 Else Twcr = &B10000100
  Bitwait Twcr.twint , Set 'Czekamy na otrzymanie bajta danych sygnalizowane ustawieniem flagi TWINT
  Half = Twdr
  Word to_read = Word to_read Or Half 'Zapisujemy ten bajt na pozycję LSB słowa Word
End Sub
```

Tabela 2. Lista rozkazów sterujących wraz z opisem ich znaczenia

Nazwa rozkazu	Adres	Opis
Read Pressure MSB	0x00	Inicjuje odczyt starszego bajtu rejestru ciśnienia
Read Pressure LSB	0x01	Inicjuje odczyt młodszego bajtu rejestru ciśnienia (2 bity znaczące: B7:B6)
Read Temperature MSB	0x02	Inicjuje odczyt starszego bajtu rejestru temperatury
Read Temperature LSB	0x03	Inicjuje odczyt młodszego bajtu rejestru temperatury (2 bity znaczące: B7:B6)
Read Coefficient data byte 1	0x04	Inicjuje odczyt starszego bajtu współczynnika korekcyjnego A0
Start Pressure Conversion	0x10	Inicjuje proces pomiaru ciśnienia (wybiera wewnętrzny czujnik ciśnienia → inicjuje i przeprowadza konwersję A/D → aktualizuje rejestry ciśnienia)
Start Temperature Conversion	0x11	Inicjuje proces pomiaru temperatury (wybiera wewnętrzny czujnik temperatury → inicjuje i przeprowadza konwersję A/D → aktualizuje rejestry temperatury)
Start both Conversions	0x12	Inicjuje proces pomiaru ciśnienia i temperatury

lecz biorąc pod uwagę, iż standardowo jest ona na poziomie 10 hPa, nie ma to wielkiego znaczenia. Kierując się tymi założeniami wspomniany wcześniej wzór przekształcamy do postaci:

$$P_{comp} = A0 + (B1 + C12 \cdot T_{adc}) \cdot P_{adc} + B2 \cdot T_{adc} = A0 + B1 \cdot P_{adc} + C12 \cdot T_{adc} \cdot P_{adc} + B2 \cdot T_{adc}$$

Teraz „rozbijamy” nasz wzór na część całkowitą (index „i” współczynników) i część ułamkową (index „f” współczynników) otrzymując następujące działania:

$$P_{comp}^i = A0 + (B1^i \cdot P_{ADC}) + (C12^i \cdot T_{ADC} \cdot P_{ADC}) + (B2^i \cdot T_{ADC})$$

Tabela 3. Lista współczynników korekcyjnych (dla współczynników o długości mniejszej niż 16 bitów występuje wyrównanie do lewej z zerami nieznaczącymi w części LSB). Dodatkowo, producent zastrzega, że wartość współczynników C11 i C22 może wynosić 0.

Współczynnik	A0	B1	B2	C12	C11	C22
Całkowita liczba bitów	16	16	16	14	11	11
Bit znaku	1	1	1	1	1	1
Liczba bitów części całkowitej	12	2	1	0	0	0
Liczba bitów części ułamkowej	3	13	14	13	10	10
Liczba dodatkowych zer po przecinku				9	11	15

$$P_{comp}^f = \frac{B1^f \cdot P_{ADC}}{B1_divider} + \frac{C12^f \cdot T_{ADC} \cdot P_{ADC}}{C12_divider} + \frac{B2^f \cdot T_{ADC}}{B2_divider}$$

gdzie:

B1_divider, C12_divider i B2_divider to dzielniki wynikające ze specyfikacji części ułamkowej poszczególnych, branych pod uwagę, współczynników korekcyjnych a samo dzielenie jest dzieleniem liczb całkowitych. Część ułamkową współczynnika A0 pomija-

Listing 4. Odczyt współczynników korekcyjnych

```
Sub Read_and_calculate_coefficients
Local Raw_coefficient As Word
Call Twi_start 'Wysyłamy sygnał Start
Call Twi_write_byte(write_addr) 'Wysyłamy adres układu MPL115A2 w trybie zapisu
Call Twi_write_byte(&H04) 'Wysyłamy rozkaz: Read Coefficient data byte #1
Call Twi_start 'Wysyłamy sygnał ReStart
Call Twi_write_byte(read_addr) 'Wysyłamy adres układu MPL115A2 w trybie odczytu inicjując tym samym tryb Master
Receiver dla naszego interfejsu TWI
Call Twi_read_word(raw_coefficient, Ack_) 'Inicjujemy odbiór A0
A0i = Raw_coefficient
Shift A0i, Right, 3 'Pomijamy część ułamkową współczynnika A0 jako mało znaczącą dla obliczenia wartości ciśnienia
Call Twi_read_word(raw_coefficient, Ack_) 'Inicjujemy odbiór B1
Call Extract_coefficients(raw_coefficient, 16, 13, B1i, B1f) 'Obliczamy część całkowitą i ułamkową B1
Call Twi_read_word(raw_coefficient, Ack_) 'Inicjujemy odbiór B2
Call Extract_coefficients(raw_coefficient, 16, 14, B2i, B2f) 'Obliczamy część całkowitą i ułamkową B2
Call Twi_read_word(raw_coefficient, Nack_) 'Inicjujemy odbiór C12 (bez ACK)
Call Extract_coefficients(raw_coefficient, 14, 13, C12i, C12f) 'Obliczamy część całkowitą i ułamkową C12
Twr = &B100T0100 'Inicjujemy wysłanie sygnału Stop: ustawiony TWINT, TWSTO i TWEN
End Sub
```

Listing 5. Obliczenie rzeczywistej i ułamkowej części współczynników korekcyjnych

```
Sub Extract_coefficients(raw_data As Word, Byval Total_bits As Byte, Byval Fraction_bits As Byte, Integer_part As Integer, Fractional_part As Integer)
Local Sign As Byte 'Wskaźnik znaku zmiennej
'Sprawdzamy czy nie mamy doczynienia ze współczynnikiem C12, dla którego trzeba dokonać odpowiednich przesunięć, jako że współczynnik ten składa się jedynie z 14 bitów
If Total_bits <> 16 Then
Shift Raw_data, Right, 2
Raw_data.15 = Raw_data.13 'Przepisanie bitu znaku, gdyż C12 może mieć wartości ujemne
Raw_data.13 = 0
End If
'Ustalamy znak współczynnika by dokonać konwersji z kodu U2 do NKB dla przypadku, gdy współczynnik < 0
Sign = Raw_data.15
If Sign <> 0 Then
Raw_data = Not Raw_data
Raw_data = Raw_data Or &H0001
End If
'Teraz wyznaczamy część całkowitą i część ułamkową zmiennej wyjściowej
Integer_part = Raw_data
Shift Integer_part, Right, Fraction_bits
Fraction_bits = 16 - Fraction_bits
Fractional_part = Raw_data
Shift Fractional_part, Left, Fraction_bits
Shift Fractional_part, Right, Fraction_bits
If Sign <> 0 Then
'Aktualizujemy wyjściowe zmienne przechowujące wartość całkowitą i ułamkową o znak
Integer_part = 0 - Integer_part
Fractional_part = 0 - Fractional_part
End If
End Sub
```

Listing 6. Odczyt wyników pomiaru

```
Sub Read_adc_values
  Call Twi_start           'Wysyłamy sygnał Start
  Call Twi_write_byte(write_addr) 'Wysyłamy adres układu MPL115A2 w trybie zapisu
  Call Twi_write_byte(&H12) 'Wysyłamy rozkaz: Start Both Conversion
  Call Twi_write_byte(&H01) 'Wysyłania tego bajta danych wymaga dokumentacja (nie tłumaczac dlaczego)
  Twcr = &B10010100 'Inicjujemy wysłanie sygnału Stop: ustawiony TWINT, TWSTO i TWEN
  Waitms 7 'Czekamy aż dokona się konwersja
  Call Twi_start           'Wysyłamy sygnał Start
  Call Twi_write_byte(write_addr) 'Wysyłamy adres układu MPL115A2 w trybie zapisu
  Call Twi_write_byte(&H00) 'Wysyłamy rozkaz: Read Pressure Hi Byte
  Call Twi_start           'Wysyłamy sygnał ReStart
  Call Twi_write_byte(read_addr) 'Wysyłamy adres układu MPL115A2 w trybie odczytu inicjując tym samym tryb Master
Receiver dla naszego interfejsu TWI
  Call Twi_read_word(p_adc , Ack) 'Inicjujemy odbiór Pressure
  Shift P_adc , Right , 6 'Tutaj mamy już 10-bitowa wartość pomiarową (surową) ciśnienia
  Call Twi_read_word(t_adc , Nack) 'Inicjujemy odbiór Temperature (bez ACK)
  Shift T_adc , Right , 6 'Tutaj mamy już 10-bitowa wartość pomiarową (surową) temperatury
  Twcr = &B10010100 'Inicjujemy wysłanie sygnału Stop: ustawiony TWINT, TWSTO i TWEN
End Sub
```

Listing 7. Wyznaczenie bezwzględnego ciśnienia atmosferycznego

```
Sub Calculate_pressure
  'Obliczamy wartość ciśnienia biorąc pod uwagę wyłącznie części całkowite współczynników korekcyjnych
  Long1 = B1i * P_adc
  Long2 = C12i * T_adc
  Long2 = Long2 * P_adc
  Long3 = B2i * T_adc
  Result = Long1 + A0i
  Result = Result + Long2
  Result = Result + Long3 'Tutaj mamy sumę składników części całkowitej
  'Obliczamy części ułamkowe współczynników korekcyjnych korzystając z odpowiednich dzielników
  Long1 = B1f * P_adc
  Long1 = Long1 \ B1_divider
  Long2 = C12f * T_adc
  Long2 = Long2 * P_adc
  Long2 = Long2 \ C12_divider
  Long3 = B2f * T_adc
  Long3 = Long3 \ B2_divider
  'Obliczamy wartość końcową dodając do sumy części całkowitej poszczególne składniki części ułamkowych
  Result = Result + Long1
  Result = Result + Long2
  Result = Result + Long3
  'Na koniec skalujemy otrzymaną wartość do zakresu 500...1150
  Result = Result * 650
  Result = Result \ 1023
  Result = Result + 500
  Pressure = Result 'Wynikowe, bezwzględne ciśnienie atmosferyczne
End Sub
```

Listing 8. Zmienne używane w programie

```
Dim P_adc As Word , T_adc As Word 'Wartości ciśnienia i temperatury odczytane z rejestrów układu MPL115A2
Dim A0i As Integer , B1i As Integer , B2i As Integer , C12i As Integer 'Części całkowite współczynników korekcyjnych
Dim B1f As Integer , B2f As Integer , C12f As Integer 'Części ułamkowe współczynników korekcyjnych
Dim Pressure As Word 'Obliczone, bezwzględne ciśnienie atmosferyczne
Dim Long1 As Long , Long2 As Long , Long3 As Long , Result As Long 'Pomocnicze zmienne obliczeniowe
```

Tabela 4. Mapa pamięci układu MPL115A2 w zakresie współczynników korekcyjnych.

Adres rejestru	Nazwa współczynnika
0x04	A0 (starszy bajt)
0x05	A0 (młodszy bajt)
0x06	B1 (starszy bajt)
0x07	B1 (młodszy bajt)
0x08	B2 (starszy bajt)
0x09	B2 (młodszy bajt)
0x0A	C12 (starszy bajt)
0x0B	C12 (młodszy bajt)
0x0C	C11 (starszy bajt)
0x0D	C11 (młodszy bajt)
0x0E	C22 (starszy bajt)
0x0F	C22 (młodszy bajt)

my, gdyż nie ma ona znaczenia, jeśli chodzi o końcowy wynik obliczeń. Wspomniane dzielniki wynoszą odpowiednio:

```
Const B1_divider = 8192
Const B2_divider = 16384
Const C12_divider = 4194304
```

W związku z powyższym, skompensowane, bezwzględne ciśnienie atmosferyczne obliczamy jak niżej:

$$P_{comp} = P_{comp}^i + P_{comp}^f$$

Rozwinięciem przedstawionych powyżej założeń w zapisie języka Bascom Basic jest procedura obliczająca wspomniane, bezwzględne ciśnienie atmosferyczne o poniższej treści (jednocześnie zapewnia ona minimalizację wygenerowanego kodu obsługi, choć korzysta z nadmiarowej liczby zmiennych) pokazana na **listingu 7**.

Na **listingu 8** przedstawiono specyfikację zmiennych używanych w programie obsługi urządzenia, które nie zostały wymienione wcześniej.

Robert Wołgajew, EP

REKLAMA

AVT5094 Beprzewodowy regulator temperatury

CENA: 160,00 PLN AVT5094 B Zestaw do samodzielnego montażu

CENA: 8,00 PLN AVT5094 F Panel czołowy

CENA: 190,00 PLN AVT5094 C Zestaw zamontowany i uruchomiony

www.sklep.avt.pl